

© 2012 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

In-Layer Multi-Buffer Framework for Rate-Controlled Scalable Video Coding

Sergio Sanz-Rodríguez, *Member, IEEE*, Fernando Díaz-de-María, *Member, IEEE*



Abstract—Temporal scalability is supported in scalable video coding (SVC) by means of hierarchical prediction structures, where the higher layers can be ignored for frame rate reduction. Nevertheless, this kind of scalability is not totally exploited by the rate control (RC) algorithms since the hypothetical reference decoder (HRD) requirement is only satisfied for the highest frame rate sub-stream of every dependency (spatial or coarse grain scalability) layer. In this paper we propose a novel RC approach that aims to deliver several HRD-compliant temporal resolutions within a particular dependency layer. Instead of using the common SVC encoder configuration consisting of a dependency layer per each temporal resolution, a compact configuration that does not require additional dependency layers for providing different HRD-compliant temporal resolutions is proposed. Specifically, the proposed framework for rate-controlled SVC uses a set of virtual buffers within a dependency layer so that their levels can be simultaneously controlled for overflow and underflow prevention while minimizing the reconstructed video distortion of the corresponding sub-streams. This in-layer multi-buffer approach has been built on top of a baseline H.264/SVC RC algorithm for variable bit rate applications. The experimental results show that our proposal achieves a good performance in terms of mean quality, quality consistency, and buffer control using a reduced number of layers.

Index Terms—H.264/advanced video coding (AVC), H.264/SVC, hypothetical reference decoder (HRD), rate control, scalable video coding (SVC), variable bit rate (VBR).

I. INTRODUCTION

DURING the last years video applications have grown in popularity because of the increasing advances on network infrastructure, data storage, and computational and memory capacity of multimedia devices. Within this technological framework, scalable video coding (SVC) provides an attractive solution for bit rate adaptation to certain application requirements, such as display resolutions and computational capabilities of target devices, or varying channel conditions. Specifically, SVC enables the extraction of either one or a subset of sub-streams from a high-quality bit stream, so that these simpler sub-streams, bearing lower spatio-temporal resolutions or reduced quality versions of the original sequence, can be decoded by a given target receiver. Furthermore, unequal error protection (UEP) or unequal erasure protection (UXP)

techniques [1] can be used to ensure an error free transmission of more important sub-streams, such as that associated with the lowest spatio-temporal resolution. UEP/UXP would be located on top of the already existing channel forward error correction. Several industries and application areas, from video conference or video surveillance [2] to Internet protocol television (IPTV) broadcast [3], have benefited from these SVC features for multimedia information delivery.

Scalable profiles have been developed for video coding standards prior to H.264/advanced video coding (AVC) [4], such as MPEG-2 [5], H.263 [6], and MPEG-4 Visual [7]. Nevertheless, most of these extensions have been rarely used in real applications. Several factors have caused that limited deployment: on the one hand, the unsuitability of traditional video transmission systems and the lack of an actual diversity of decoding devices; on the other hand, the loss in coding efficiency and the increase in decoding complexity when compared to non-scalable profiles [8]. Consequently, for earlier coding standards, alternative approaches such as *simulcasting* or *transcoding* have been preferred to scalable profiles. In contrast, nowadays, the transmission systems have evolved to properly manage this kind of traffic, and the diversity of devices has become an apparent reality. Furthermore, the recently standardized scalable extension of H.264/AVC, named H.264/SVC [8], [9], is able to provide both coding efficiency and decoding complexity more similar to those achieved using non-scalable coding.

As prior scalable standards, H.264/SVC supports *spatial*, *temporal*, and *quality* (or *signal-to-noise ratio*, SNR) scalability. For spatial scalability, a layered coding approach is used to encode different picture sizes of an input video sequence. The base layer provides an H.264/AVC compatible bit stream for the lowest spatial resolution, while larger picture sizes are encoded as enhancement layers. In addition, the redundancies between contiguous spatial layers can be exploited via inter-layer prediction tools in order to improve the coding efficiency.

Moreover, each spatial layer is capable of supporting temporal scalability by means of hierarchical prediction structures, which go from these very efficient ones using hierarchical bipredictive (B) pictures to those with zero structural delay. The pictures of the temporal base layer can only use previous pictures of the same layer as references. The pictures of a temporal enhancement layer can be bidirectionally predicted from pictures of a lower layer. The number of temporal layers in a spatial layer is determined by the group of pictures (GoP) size, defined in H.264/SVC as the distance between two consecutive intra (I) or predictive pictures, also named *key* pictures.

This work has been partially supported by the National Grant TEC2011-26807 of the Spanish Ministry of Science and Innovation.

The authors are with the Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Leganés, Madrid 28911 Spain (e-mail: sescalona, fdiaz @tsc.uc3m.es).

When SNR scalability is considered, different reconstruction quality levels with the same spatio-temporal resolution are provided. In particular, the H.264/SVC standard defines two types of SNR scalable coding: coarse grain scalability (CGS) and medium grain scalability (MGS). The first is a special case of spatial scalability with identical picture sizes. The second employs a multi-layer approach within a spatial layer in order to provide a finer bit rate granularity in the rate-distortion (R-D) space.

For a variety of video coding applications, the rate control (RC) algorithm is a key subsystem in both scalable (multi-layer) and non-scalable (single-layer) video coding systems. The RC algorithm works in two steps. First, a bit budget is allocated to a video segment such as GoP, picture, or macroblock according to the video content, the target bit rate, and the buffer constraints imposed by the hypothetical reference decoder (HRD) [10] (additionally, for digital storage, the bit allocation method must be aware of the maximum allowed storage capacity). Second, a quantization parameter (QP) value is assigned to the video segment in order to satisfy these buffer and/or budget constraints, while minimizing the reconstructed video distortion. In the case of SVC, it is also worth noting that the RC algorithm actually consists of a set of rate controllers, each one located at each dependency (spatial scalability or CGS) or MGS layer, to provide a set of HRD-compliant scalable sub-streams, each one for a certain target bit rate suitable for a target decoding terminal managing a particular spatio-temporal resolution or computational capability.

The RC problem has been extensively studied for both single-layer video coding and SVC. According to the target application, two kinds of RC methods have been proposed: constant bit rate (CBR) and variable bit rate (VBR) control algorithms. On the one hand, the CBR controllers, commonly used for real-time video conference, pursue a short-term target bit rate adjustment to guarantee a low buffer delay. On the other hand, the VBR controllers, typically used for video streaming or digital storage, manage a long-term target bit rate adaptation at the expense of a longer buffer delay to maintain a high visual quality consistency [11], [12].

Most of the CBR controllers have focused on modeling the discrete cosine transform (DCT) coefficients to provide analytical R-D functions for QP estimation. In single-layer video coding, several R-D functions have been proposed: logarithmic [13]–[15], linear [16], [17], quadratic [18]–[22] (in particular, Chen *et al.* [21] proposed separate R-D models for luminance and chrominance DCT coefficients, whilst Kwon *et al.* [22] proposed separate rate models for source and header bits), ρ -domain [23] and exponential [24], [25]. Although the RC algorithm is not a normative part of video coding standards, it usually forms part of their reference implementations, such as the Test Model Version 5 for MPEG-2 [16], the Verification Model Version 8 for MPEG-4 [18], the Test Model Near-Term 8 for H.263 [14], and the Joint Model for H.264/AVC [19]. Likewise, most of the CBR control algorithms proposed for SVC also rely on analytical R-D models for QP estimation; in particular: logarithmic [26], linear [27], quadratic [28], ρ -domain [29], [30], and exponential [31], [32] models have been proposed.

Regarding VBR controllers, several solutions for single-layer coding have been proposed for a variety of applications, such as video streaming and broadcast [33], [34], one-pass digital storage [35], [36], or two-pass digital storage [37], [38]. Other methods, such as those in [39] and [40], take advantage of networking infrastructures supporting VBR transport [12] to improve the visual quality while reducing the buffer delay. For SVC, a few approaches have been proposed for video streaming [41], [42], broadcast [43], as well as applications dealing with varying channel conditions [44]. From the R-D modeling point of view, while some of these methods rely on analytical R-D functions for QP estimation [33], [35], [37], [41], [44], others estimate a QP increment with respect to a reference QP value [34], [36], [39], [40], [43], to reduce the QP variation for the sake of visual quality consistency.

The bit allocation problem has also been studied for SVC. In particular, R-D models for optimal bit allocation among spatial, quality, and temporal layers have been proposed in [31] and [32]. Likewise, the optimal distribution of the total target bit rate among dependency layers for visual quality maximization has been addressed in [45]. It is also worth noting that quality scalability was specially investigated for MPEG-4 fine grain scalability (FGS) [41], [44] and H.264 MGS [31], [42], [46], [47].

Nevertheless, all these previous RC approaches for SVC only guarantee the HRD requirement for the highest temporal layer of each dependency layer. Therefore, temporal scalability is not fully exploited since, in order to deliver HRD-compliant sub-streams, it is necessary to increase the number of dependency layers. For instance, if a video transmission service offered the same quality of service (QoS) to two target decoders with identical spatial resolutions but different temporal resolutions, the SVC encoder would have to use two CGS layers, one per temporal layer. Although the two desired HRD-compliant sub-streams are provided, temporal scalability is underused since each one of the highest temporal layers actually also contains the lower frame rate. In summary, the common SVC encoder configuration for rate-controlled video may incur in redundant dependency layers, producing an unnecessary increase in bit rate and coding complexity.

In this paper we propose a novel RC approach for delivering HRD-compliant temporal resolutions within a particular dependency layer. Specifically, the proposed method uses a set of virtual buffers (one per HRD-compliant temporal resolution) within a dependency layer, so that the buffer levels can be simultaneously controlled for overflow and underflow prevention, while minimizing the reconstructed video distortion of the corresponding sub-streams. The proposed in-layer multi-buffer (IL-MB) approach has been built on top of a baseline RC algorithm described in [43], which relies on an effective radial basis function (RBF)-based model for QP estimation in VBR scenarios.

The paper is organized as follows. In Section II, an overview of the baseline RC algorithm for H.264/SVC is given. In Section III a detailed description of the proposed IL-MB VBR controller is provided. First, a general description of the proposed method is given. Second, the proposed VBR controller is described stage by stage, making special emphasis

TABLE I
SUMMARY OF NOTATION

D	Number of dependency layers
d	Dependency layer identifier
t	Temporal layer identifier
j	Current picture number
BD	Buffer size in seconds
nTF	Normalized target buffer fullness with respect to BD
H	Gaussian-type function
L	Number of Gaussian-type functions
$\mathbf{C}, \Sigma, \mathbf{w}, w_0$	Centers, widths, weights, and bias of the RBF network
For each layer d	
$T^{(d)}$	Number of temporal layers
$t_{max}^{(d)}$	Maximum temporal layer identifier
$t_{min}^{(d)}$	Minimum involved temporal layer identifier
$RC^{(d)}$	Rate control module
k	Temporal layer index that goes from $t_{min}^{(d)}$ to $t_{max}^{(d)}$
$R^{(d,k)}$	Target bit rate for the sub-stream k
$f_{out}^{(d,k)}$	Output frame rate of the sub-stream k
$QP^{(d)}$	QP value
$QP_{REF}^{(d)}$	Reference QP value
$\Delta QP^{(d)}$	QP increment
$\mathbf{QP}^{(d)}$	Set of previous QPs
$BS^{(d,k)}$	Buffer size in bits associated with the sub-stream k
$V^{(d,k)}$	Buffer fullness associated with the sub-stream k
$\mathbf{V}^{(d,k)}$	Set of involved buffers after encoding a picture at the layer t
$n\mathbf{V}^{(d)}$	Set of normalized versions of all the buffer fullness
$nV^{(d)}$	Normalized version of the buffer fullness
$G^{(d,t,k)}$	AU target bits at the layer t to meet $R^{(d,k)}$
$AU^{(d,t)}$	AU output bits of a picture at the layer t
$n\mathbf{AU}^{(d)}$	Set of normalized versions of the AU output bits
$nAU^{(d)}$	Normalized version of the AU output bits
$\overline{C}_{TEX}^{(d,t)}$	Average texture complexity of the layer t
$\overline{C}_{MOT}^{(d,t)}$	Average motion complexity of the layer t
$\mathbf{X}^{(d)}$	Input vector to the RBF network

on the *buffer modeling* stage, which is used to properly manage the set of virtual buffers. Section IV reports and discusses the experimental results. Finally, in Section V conclusions are drawn and future work is outlined.

II. BASELINE VBR CONTROLLER SUMMARY [43]

A. System Overview

In order to make the reading easier, the notation used along the paper has been summarized in Table I. In this way, the reader may turn to it when necessary and some superfluous definitions may be skipped in the text to make it more readable. The baseline RC scheme is illustrated in dark gray in Fig. 1 for an encoder consisting of two dependency layers. Let us denote as D the number of dependency layers, identified as $d = \{0, 1 \dots D - 1\}$, and let us denote as $T^{(d)}$ the number of temporal layers for a particular dependency layer, identified as $t = \{0, 1 \dots T^{(d)} - 1\}$. Alternatively, for the sake of notation consistency with the proposed method, we will also refer to the maximum temporal layer identifier $T^{(d)} - 1$ as $t_{max}^{(d)}$.

Each dependency layer d involves a rate controller $RC^{(d)}$ and a virtual buffer. The virtual buffer at the layer d receives the contributions of those layers with identifiers from $(0, 0)$ to $(d, t_{max}^{(d)})$ and simulates the encoder buffering process of the highest temporal resolution sub-stream. Thus, both the virtual buffer and the corresponding sub-stream will be identified as

$(d, t_{max}^{(d)})$ to indicate that the video packets with higher spatio-temporal identifiers will be discarded by the target decoder.

The generation of an HRD-compliant sub-stream depends on two fundamental parameters: the target bit rate $R^{(d,t_{max}^{(d)})}$ and the output frame rate $f_{out}^{(d,t_{max}^{(d)})}$. It should be noticed that $R^{(d,t_{max}^{(d)})}$ must be higher than those associated with lower layers, i.e., $R^{(d-x,t_{max}^{(d)}-y)} \leq R^{(d,t_{max}^{(d)})}$, with $x = 0, 1 \dots d$, and $y = 0, 1 \dots t_{max}^{(d)}$, since those lower layers form part of the sub-stream $(d, t_{max}^{(d)})$.

If a particular dependency layer contained additional $Q^{(d)}$ MGS refinements, denoted as $q = \{1 \dots Q^{(d)} - 1\}$ (it should be noticed that $q = 0$ refers to the quality base layer for a given dependency layer), a rate controller $RC^{(d,q)}$ and the corresponding virtual buffer would be located at each spatio-quality layer (d, q) . However, in order to make the notation easier, hereafter we will only consider spatial/CGS and temporal scalability.

In order to encode the j th picture with spatio-temporal identifier (d, t) , the $RC^{(d)}$ module should provide an appropriate $QP_j^{(d)}$ value, on a frame basis, so that the QP fluctuation is minimized (to improve visual quality consistency), while the buffer fullness $V^{(d,t_{max}^{(d)})}$ is maintained at secure levels. To this end, the $RC^{(d)}$ module operation leans on three input parameters:

- 1) The fullness $V^{(d,t_{max}^{(d)})}$ of the corresponding virtual buffer.
- 2) The amount of bits yield by the encoding of the spatial layers 0 to d for a given time instant. Henceforth, following the H.264/SVC nomenclature [8], we will refer to this amount of bits as *access unit* (AU) output bits $AU^{(d,t)}$.
- 3) The QP value used to encode the previous picture of the same dependency layer $QP_{j-1}^{(d)}$.

A proper QP increment $\Delta QP^{(d)}$ is estimated from the two firsts, and $QP_{j-1}^{(d)}$ is employed as a reference value to obtain the final quantization parameter as follows:

$$QP_j^{(d)} = QP_{j-1}^{(d)} + \Delta QP^{(d)}. \quad (1)$$

Furthermore, in the case of CGS scalability, the QP obtained is lower bounded by the QP of the reference layer, so that a higher quality for the enhancement layer is ensured:

$$QP_j^{(d)} = \min \left[QP_j^{(d-1)}, QP_j^{(d)} \right]. \quad (2)$$

The VBR control algorithm for a specific spatial or CGS layer, i.e., the algorithm that estimates an appropriate QP increment for the j th picture with identifier (d, t) is illustrated in Fig. 2. As shown in the figure, the RC module $RC^{(d)}$ is organized in two stages, named *parameter updating* and *RBF-based QP increment estimation*. These stages are briefly described below.

B. Parameter Updating

After encoding the $(j - 1)$ th picture with layer identifier (d, t') (t' is used instead of t because the previous picture can belong to a different temporal layer), two parameters required to estimate the QP increment are updated: 1) a normalized version of the buffer fullness, denoted as $nV^{(d)}$; and 2) a normalized version of the amount of bits generated by the

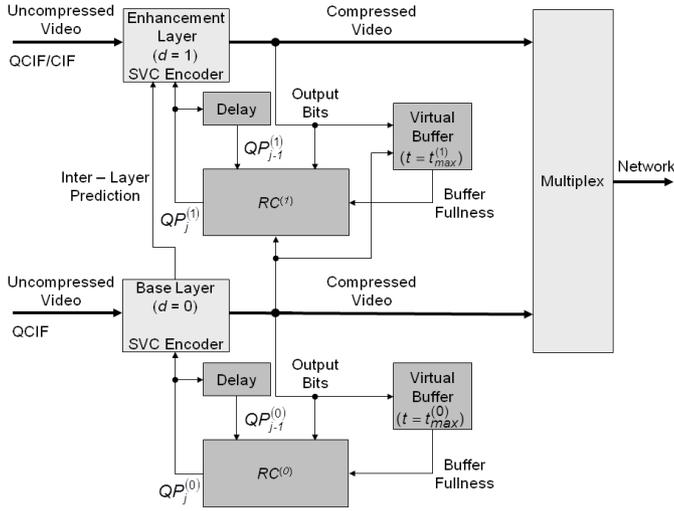


Fig. 1. Block diagram of the baseline H.264/SVC RC scheme for two dependency layers ($D=2$).

AU, denoted as $nAU^{(d)}$. These normalized versions of the buffer fullness and the AU output bits are defined as follows:

$$nV^{(d)} = \max \left[0, \min \left[\frac{V^{(d,t_{max}^{(d)})}}{BS^{(d,t_{max}^{(d)})}}, 1 \right] \right], \quad (3)$$

$$nAU^{(d)} = \max \left[\frac{1}{2}, \min \left[\frac{AU^{(d,t')}}{G^{(d,t',t_{max}^{(d)})}}, 2 \right] \right], \quad (4)$$

where $BS^{(d,t_{max}^{(d)})}$ denotes the buffer size in bits, $V^{(d,t_{max}^{(d)})}$ and $AU^{(d,t')}$ have already been defined, and $G^{(d,t',t_{max}^{(d)})}$ denotes the bit budget for the AU at the layer (d, t') in order for the sub-stream $(d, t_{max}^{(d)})$ to satisfy the target bit rate constraint $R^{(d,t_{max}^{(d)})}$. The updating equations for $V^{(d,t_{max}^{(d)})}$, $AU^{(d,t')}$, and $G^{(d,t',t_{max}^{(d)})}$ are briefly summarized next.

The buffer fullness $V^{(d,t_{max}^{(d)})}$ is updated as follows:

$$V_j^{(d,t_{max}^{(d)})} = V_{j-1}^{(d,t_{max}^{(d)})} + AU_{j-1}^{(d,t')} - \frac{R^{(d,t_{max}^{(d)})}}{f_{out}^{(d,t_{max}^{(d)})}}, \quad (5)$$

The amount AU output bits $AU^{(d,t')}$ is updated as:

$$AU_{j-1}^{(d,t')} = \sum_{m=0}^d \left(b_{j-1}^{(m,t')} + h_{j-1}^{(m,t')} \right), \quad (6)$$

where $b_{j-1}^{(m,t')}$ and $h_{j-1}^{(m,t')}$ are, respectively, the amount of texture bits and header plus motion data bits generated by the $(j-1)$ th picture with layer identifier (m, t') .

Finally, $G^{(d,t',t_{max}^{(d)})}$ is determined by the following model:

$$G^{(d,t',t_{max}^{(d)})} = G_{NOM}^{(d,t_{max}^{(d)})} + \Delta G_{TEX}^{(d,t',t_{max}^{(d)})} + \Delta G_{MOT}^{(d,t',t_{max}^{(d)})}, \quad (7)$$

where $G_{NOM}^{(d,t_{max}^{(d)})}$ is the nominal bit budget:

$$G_{NOM}^{(d,t_{max}^{(d)})} = \frac{R^{(d,t_{max}^{(d)})}}{f_{out}^{(d,t_{max}^{(d)})}}, \quad (8)$$

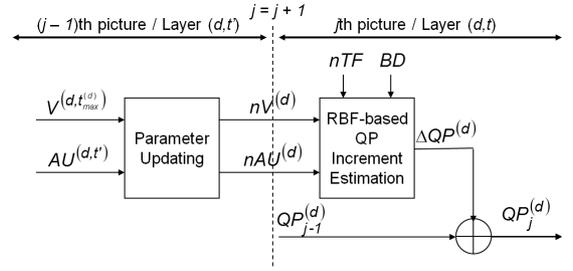


Fig. 2. Block diagram of the rate controller module $RC^{(d)}$ for a specific dependency layer d .

and $\Delta G_{TEX}^{(d,t',t_{max}^{(d)})}$ and $\Delta G_{MOT}^{(d,t',t_{max}^{(d)})}$ represent texture and motion bit increments, respectively, i.e.:

$$\Delta G_{TEX}^{(d,t',t_{max}^{(d)})} = \frac{R^{(d,t_{max}^{(d)})}}{f_{out}^{(d,t_{max}^{(d)})}} \left(\frac{\bar{C}_{TEX}^{(d,t')} \sum_{u=0}^{t_{max}^{(d)}} N^{(d,u)}}{\sum_{u=0}^{t_{max}^{(d)}} (\bar{C}_{TEX}^{(d,u)} N^{(d,u)})} - 1 \right), \quad (9)$$

$$\Delta G_{MOT}^{(d,t',t_{max}^{(d)})} = \bar{C}_{MOT}^{(d,t')} - \frac{\bar{C}_{TEX}^{(d,t')} \sum_{u=0}^{t_{max}^{(d)}} (\bar{C}_{MOT}^{(d,u)} N^{(d,u)})}{\sum_{u=0}^{t_{max}^{(d)}} (\bar{C}_{TEX}^{(d,u)} N^{(d,u)})}, \quad (10)$$

where $N^{(d,u)}$ is the total number of pictures per GoP with layer identifier (d, u) , and $\bar{C}_{TEX}^{(d,t')}$ and $\bar{C}_{MOT}^{(d,t')}$ denote, respectively, the average texture and motion complexities of the encoded pictures at the dependency layers 0 to d belonging to the temporal layer t' . The following updating equations for both complexity measurements are proposed:

$$\bar{C}_{TEX}^{(d,t')} = \alpha \sum_{m=0}^d \left(Q_{j-1}^{(m)} b_{j-1}^{(m,t')} \right) + (1 - \alpha) \bar{C}_{TEX}^{(d,t')}, \quad (11)$$

$$\bar{C}_{MOT}^{(d,t')} = \beta \sum_{m=0}^d h_{j-1}^{(m,t')} + (1 - \beta) \bar{C}_{MOT}^{(d,t')}, \quad (12)$$

where α and β are forgetting factors that are set to 0.5 in our experiments, and $Q_{j-1}^{(m)}$ is the quantization step value associated with $QP_{j-1}^{(m)}$.

C. RBF-based QP Increment Estimation

Before encoding the j th picture, the proper QP increment $\Delta QP^{(d)}$ with respect to $QP_{j-1}^{(d)}$ should be estimated from $nV^{(d)}$ in Eq. (3) and $nAU^{(d)}$ in Eq. (4). Furthermore, two additional constant parameters are considered as inputs to this process in order to provide a solution suitable for a variety of scenarios. The first, denoted as nTF , is the normalized target buffer fullness with respect to the buffer size; and the second, denoted as BD , is the maximum buffering delay (or buffer size in seconds), which is related to that measured in bits as

$$BS^{(d,t_{max}^{(d)})} = BD \times R^{(d,t_{max}^{(d)})}. \quad (13)$$

Thus, the proposed $\Delta QP^{(d)}$ estimation method operates on the following input vector:

$$\mathbf{X}^{(d)} = \left(nV^{(d)}, nAU^{(d)}, nTF, BD \right)^T, \quad (14)$$

implicitly assuming that all the virtual buffers share the same nTF and BD values. Since the input parameters nTF and BD are set before starting the encoding process, the proposed $\Delta QP^{(d)}$ prediction function can be seen as a surface whose shape depends on these constants.

An RBF network is used to estimate $\Delta QP^{(d)}$ from the input vector $\mathbf{X}^{(d)}$ for any dependency layer. This RBF-based estimation obeys:

$$\Delta QP^{(d)} = \text{round} \left[w_0 + \sum_{i=1}^L w_i H_i(\mathbf{X}^{(d)}) \right], \quad (15)$$

where L is the number of basis functions $\{H_i(\mathbf{X}^{(d)})\}_{i=1,\dots,L}$, w_i the output weights, and w_0 the bias. The output of the RBF network is then converted into an integer, given the discrete nature of the QP in H.264/SVC. The basis functions are Gaussian-type functions with centers \mathbf{C}_i and widths Σ , that is:

$$H_i(\mathbf{X}^{(d)}) = \exp \left(- \sum_{j=1}^4 \frac{(X_j^{(d)} - C_{ij})^2}{\Sigma_j^2} \right). \quad (16)$$

The training of the RBF network relies on a data set containing pairs *input vector-desired output*, which have to be previously generated. Once these training data were generated, it was observed that their distributions for key (K) and non-K (NK) pictures were different enough to justify the design of two RBF networks, one for K pictures and the other for NK pictures. Furthermore, some validation experiments were performed to properly dimension the RBF networks whose results led to 7 Gaussian functions in both cases.

Finally, since some unnecessary fluctuations of the QP value at NK pictures were observed in cases of stationary video complexity when the buffer level approached the target buffer fullness, a simple post-processing stage of the output of the NK-picture RBF network was proposed, that obeys:

$$\Delta QP^{(d)} = \begin{cases} -1 & \text{if } \Delta QP^{(d)} = -2 \\ 0 & \text{if } \Delta QP^{(d)} = -1 \\ 0 & \text{if } \Delta QP^{(d)} = 1 \\ 1 & \text{if } \Delta QP^{(d)} = 2. \end{cases} \quad (17)$$

In doing so, the number of short-term QP fluctuations happening in stationary complexity situations was minimized without decreasing the performance in time-varying situations.

III. IN-LAYER MULTI-BUFFER VBR CONTROLLER

A. System Overview

The proposed VBR control scheme is illustrated in Fig. 3. For clarity reasons, only the dependency base layer ($d = 0$) of the SVC encoder is shown. The blocks depicted in dark gray are the extensions required by the baseline VBR controller shown in Fig. 1 to become an IL-MB rate controller.

Each dependency layer d involves an RC module $RC^{(d)}$ and a set of virtual buffers. Each of these virtual buffers simulates the encoder buffering process of the sub-stream corresponding to certain temporal resolution. In order to properly formulate the IL-MB rate controller, a parameter $t_{min}^{(d)}$ is introduced that

indicates which of those temporal resolution sub-streams from $(d, 0)$ to $(d, t_{max}^{(d)})$ should comply with the HRD constraints. Specifically, when $t_{min}^{(d)} = t_{max}^{(d)}$, the proposed IL-MB RC scheme becomes the baseline algorithm.

To make the explanation of the IL-MB framework easier, let us follow the example illustrated in Fig. 3. In particular, the input video is a quarter common intermediate format (QCIF) sequence at 25 Hz using a GoP size of 8 pictures, so that encoded video from QCIF@3.125 Hz to QCIF@25 Hz can be provided. Setting $t_{min}^{(0)} = 1$ means that the three higher temporal resolution sub-streams $(0, 1)$, $(0, 2)$, and $(0, 3)$ should be HRD-compliant and, consequently, their corresponding virtual buffers should be controlled for proper video content delivery. For the lowest temporal resolution sub-stream, however, the HRD compliance would not be guaranteed.

Following with the example, when the j th picture with layer identifier $(0, 2)$ (see Fig. 3) is going to be encoded, the goal of the the RC module $RC^{(0)}$ is to provide an appropriate $QP_j^{(0)}$ value, so that the set of virtual buffers involved are maintained at secure levels. Specifically, the set of virtual buffers involved in the encoding of the j th picture with layer identifier $(0, 2)$ are:

$$\mathbf{V}^{(0,2)} = \left\{ V^{(0,k)} \right\}_{k=\max[t_{min}^{(0)}, 2] \dots t_{max}^{(0)}},$$

where $V^{(0,k)}$ denotes the buffer fullness associated with the sub-stream $(0, k)$, with $k = \max[t_{min}^{(0)}, 2] \dots t_{max}^{(0)}$. It should be noticed that, since $t_{min}^{(0)} = 1$, the lowest k value is 2 and, therefore, the two higher virtual buffers are updated. However, if the picture belonged to a temporal layer lower than or equal to $t_{min}^{(0)}$, the three virtual buffers would be updated. From now on, we will refer to the virtual buffers to be updated at the time instant j as *involved buffers*.

It is also worth mentioning that all the involved buffers must be taken into account to estimate the current QP value, since a proper behavior is not guaranteed in all of them otherwise. Thus, the method for properly controlling any set $\mathbf{V}^{(d,t)}$ becomes the main focus of the proposed IL-MB VBR controller.

The rate controller $RC^{(d)}$, similarly to what was described for the baseline RC approach, obtains a reference QP, $QP_{REF}^{(d)}$, estimates a $\Delta QP^{(d)}$ value, and finally computes the desired $QP_j^{(d)}$ as follows:

$$QP_j^{(d)} = QP_{REF}^{(d)} + \Delta QP^{(d)}, \quad (18)$$

The reference QP is computed from those QPs used for encoding the last pictures belonging to the sub-streams $(d, t_{min}^{(d)})$ to $(d, t_{max}^{(d)})$ (see Subsection III-B2 for details). This set of previous QPs, defined as

$$\mathbf{QP}^{(d)} = \left\{ QP^{(d,k)} \right\}_{k=t_{min}^{(d)} \dots t_{max}^{(d)}},$$

is updated on a frame basis according to the involved buffers at the time instant j , as described in Algorithm 1. It should be noticed that the storage of this set of QPs requires a memory block (see Fig. 3) that was not necessary in the baseline approach (see Fig. 1), where there was just a delay line to make previous QP value available.

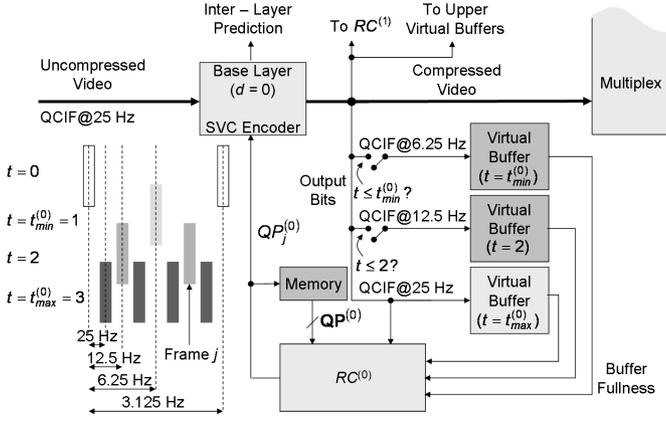


Fig. 3. Block diagram of the proposed H.264/SVC RC scheme for IL-MB control. Only the spatial base layer is depicted for the sake of clarity.

Algorithm 1 $QP^{(d)}$ updating procedure

1. **for** $k = \max[t_{min}^{(d)}, t]$ to $t_{max}^{(d)}$ **do** {involved buffers}
2. $QP^{(d,k)} \leftarrow QP_j^{(d)}$
3. **end for**

The QP increment is selected to provide a slow QP variation so that the visual quality consistency is improved. Similarly to what was described for the baseline VBR control algorithm, the following input parameters are required to compute $\Delta QP^{(d)}$:

- 1) The current fullness of the virtual buffers $(d, t_{min}^{(d)})$ to $(d, t_{max}^{(d)})$.
- 2) The amount $AU^{(d,t)}$ of AU output bits.

In the following subsection, a detailed description of the RC module for IL-MB control at a specific dependency layer is given.

B. The Rate Controller Module $RC^{(d)}$

The MB-based RC module $RC^{(d)}$ is illustrated in Fig. 4. The estimation of $QP_j^{(d)}$ is performed in three stages, namely: *parameter updating*, *buffer modeling* and *RBF-based QP increment estimation*, which are described in more detail through the next subsections.

1) *Parameter Updating*: After encoding the $(j-1)$ th picture with layer identifier (d, t') , two parameter sets, required to estimate the QP increment, should be updated: 1) the normalized versions of the buffer levels $(d, t_{min}^{(d)})$ to $(d, t_{max}^{(d)})$, denoted as $nV^{(d)}$; and 2) the normalized versions of $AU^{(d,t')}$ for the sub-streams $(d, t_{min}^{(d)})$ to $(d, t_{max}^{(d)})$, denoted as $nAU^{(d)}$. These parameter sets are defined as follows:

$$nV^{(d)} = \left\{ \frac{V^{(d,k)}}{BS^{(d,k)}} \right\}_{k=t_{min}^{(d)} \dots t_{max}^{(d)}},$$

$$nAU^{(d)} = \left\{ \frac{AU^{(d,t')}}{G^{(d,t',k)}} \right\}_{k=t_{min}^{(d)} \dots t_{max}^{(d)}},$$

where $BS^{(d,k)}$ is the buffer size in bits for the sub-stream (d, k) , which is computed from the buffer size in seconds,

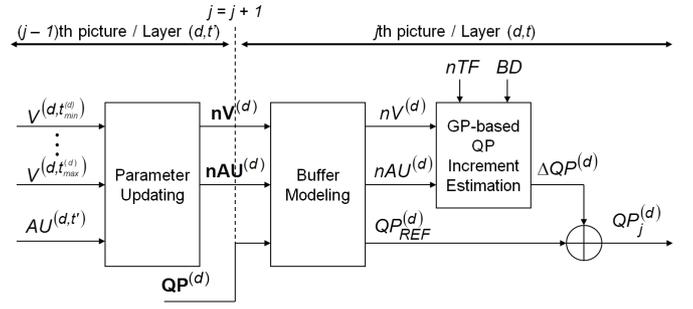


Fig. 4. Block diagram of the MB-based rate controller module $RC^{(d)}$ for a specific dependency layer d .

BD , and the target bit rate, $R^{(d,k)}$ (see Eq. (13)); and $G^{(d,t',k)}$ is the AU target bits at the layer (d, t') to satisfy $R^{(d,k)}$.

These updating equations require the previous update of the involved buffers $V^{(d,t')}$ and the estimation of the set of AU target bits $\{G^{(d,t',k)}\}$. In turn, the update of the involved buffers requires to obtain the AU output bits $AU^{(d,t')}$, and the estimation of the set of AU target bits requires the previous update of average texture and motion complexities for each temporal layer u from 0 to $t_{max}^{(d)}$, $\overline{C}_{TEX}^{(d,u)}$ and $\overline{C}_{MOT}^{(d,u)}$, respectively.

The virtual buffer levels, the AU output bits, the AU target bits, as well as the average texture and motion complexities are updated as in Subsection II-B, but replacing $t_{max}^{(d)}$ by the index k , which takes values from $t_{min}^{(d)}$ to $t_{max}^{(d)}$. Algorithm 2 summarizes the complete updating procedure for $nV^{(d)}$ and $nAU^{(d)}$.

Algorithm 2 $nV^{(d)}$ and $nAU^{(d)}$ updating procedure

1. Compute $AU^{(d,t')}$ (6)
2. Update $\overline{C}_{TEX}^{(d,t')}$ (11)
3. Update $\overline{C}_{MOT}^{(d,t')}$ (12)
4. **for** $k = \max[t_{min}^{(d)}, t']$ to $t_{max}^{(d)}$ **do** {involved buffers}
5. Update $V^{(d,k)}$ (5)
6. Compute $G^{(d,t',k)}$ (7), (8), (9), (10)
7. $nV^{(d,k)} \leftarrow \max \left[0, \min \left[\frac{V^{(d,k)}}{BS^{(d,k)}}, 1 \right] \right]$
8. $nAU^{(d,k)} \leftarrow \max \left[\frac{1}{2}, \min \left[\frac{AU^{(d,t')}}{G^{(d,t',k)}}, 2 \right] \right]$
9. **end for**

2) *Buffer Modeling*: In this stage three parameters required to estimate the QP value are computed. These parameters are representative values of the sets $QP^{(d)}$ (Algorithm 1), $nV^{(d)}$, and $nAU^{(d)}$ (Algorithm 2), which are denoted as $QP_{REF}^{(d)}$, $nV^{(d)}$, and $nAU^{(d)}$, respectively. The first parameter is used as reference QP in Eq. (18), while the last two are required for $\Delta QP^{(d)}$ estimation.

The buffer modeling algorithm suggested for estimating the aforementioned values is made up of several decision rules that are described next. If none of the involved buffer levels is close to overflow or underflow, then $nV^{(d)}$, $nAU^{(d)}$ and $QP_{REF}^{(d)}$ are computed as the arithmetic average of the parameters corresponding to the involved temporal resolutions. Otherwise, only the parameters coming from that temporal resolution showing

the most critical buffer fullness is considered. Nevertheless, given that more than one involved buffer fullness could be considered as critical at a certain time instant, the following precedence rules have been established (relying on certain observations about the time evolution of the virtual buffers for a variety of video sequences):

- 1) Since the overflow risk is more likely than the underflow risk, especially when encoding I pictures, the overflow risk is given precedence in each involved buffer.
- 2) Since the buffer of the lowest temporal resolution usually exhibits the largest fluctuations and, therefore, the highest overflow and underflow risks (since its buffer size in bits is the smallest for a given pre-established BD value), the involved buffer levels are given precedence according to their temporal layer identifier.

The pseudocode given in Algorithm 3 summarizes the proposed buffer modeling process.

Algorithm 3 $nV^{(d)}$, $nAU^{(d)}$ and $QP_{REF}^{(d)}$ updating procedure

```

1.  $nV^{(d)} = nAU^{(d)} = QP_{REF}^{(d)} = 0$ 
2.  $NumOfInvolvedBuffers = t_{max}^{(d)} - \max[t_{min}^{(d)}, t] + 1$ 
3. for  $k = \max[t_{min}^{(d)}, t]$  to  $t_{max}^{(d)}$  do {involved buffers}
4.   if  $nV^{(d,k)} \geq 0.8$  then {overflow risk}
5.      $nV^{(d)} \leftarrow nV^{(d,k)}$ 
6.      $nAU^{(d)} \leftarrow nAU^{(d,k)}$ 
7.      $QP_{REF}^{(d)} \leftarrow QP^{(d,k)}$ 
8.     break for
9.   else if  $nV^{(d,k)} \leq 0.2$  then {underflow risk}
10.     $nV^{(d)} \leftarrow nV^{(d,k)}$ 
11.     $nAU^{(d)} \leftarrow nAU^{(d,k)}$ 
12.     $QP_{REF}^{(d)} \leftarrow QP^{(d,k)}$ 
13.    break for
14.   else {secure level}
15.      $nV^{(d)} \leftarrow nV^{(d)} + nV^{(d,k)}$ 
16.      $nAU^{(d)} \leftarrow nAU^{(d)} + nAU^{(d,k)}$ 
17.      $QP_{REF}^{(d)} \leftarrow QP_{REF}^{(d)} + QP^{(d,k)}$ 
18.     if  $k = t_{max}^{(d)}$  then {all buffers at secure levels}
19.        $nV^{(d)} \leftarrow \frac{nV^{(d)}}{NumOfInvolvedBuffers}$ 
20.        $nAU^{(d)} \leftarrow \frac{nAU^{(d)}}{NumOfInvolvedBuffers}$ 
21.        $QP_{REF}^{(d)} \leftarrow \text{round} \left[ \frac{QP_{REF}^{(d)}}{NumOfInvolvedBuffers} \right]$ 
22.     end if
23.   end if
24. end for

```

It is worth noticing that, although the given description of the buffer modeling stage is tied to the baseline RC algorithm formulation, the underlying ideas might be adapted to any other RC algorithm for SVC in order to obtain the proper values of the required parameters for QP estimation.

3) *RBF-based QP Increment Estimation*: As in the baseline RC scheme, the four-dimensional input vector given in Eq. (14) is fed into an RBF network to produce a $\Delta QP^{(d)}$ estimation. Actually, two different networks are used, one for K pictures and the other for NK pictures. The architecture

of each RBF network is the same as that given in Eqs. (15) and (16); however, the RBF network parameters must be specifically trained to cope with the proposed IL-MB method, where the buffer and distortion constraints for QP selection are tougher; in particular, the RBF network parameters should be chosen to properly deal with the fact that several buffers have to be simultaneously controlled within a dependency layer.

In order to find the most suitable RBF network parameters, a training data set was previously generated. Subsequently, the training and parameter selection processes were performed. To this purpose, the general methodology described in [43] was followed; nevertheless, the cost function used for data labeling had to be modified so that the desired QP increment would adapt to the IL-MB framework, providing a good tradeoff between the control of the involved buffers and the quality consistency of the corresponding sub-streams. The adapted cost function is given in Appendix A.

The training and validation results led us to select 10 Gaussian-type functions for both (K- and NK-picture) RBF networks, whose parameters are also given in Appendix A.

Finally, the post-processing stage of the output of the NK-picture RBF network given in Eq. (17) is also performed in order to reduce unnecessary QP fluctuations.

IV. EXPERIMENTS AND RESULTS

The Joint Scalable Video Model (JSVM) H.264/SVC reference software version JSVM 9.16 [48] was used to implement the proposed IL-MB VBR controller. Its performance was compared to other two methods: 1) constant QP (CQP) encoding¹, which can be seen as an unconstrained VBR controller [11] and was used as a reference for nearly constant quality video; and 2) our baseline VBR controller described in [43], which can be seen as a particular case of the proposed method when $t_{min}^{(d)} = t_{max}^{(d)}$ for every dependency layer.

In the following subsections, the SVC encoder and RC configurations employed for comparisons are described, the experimental results are given, and a discussion concerning these results is provided.

A. Description of the SVC Encoder and RC Configurations

According to the SVC testing conditions recommended in [49], the mobile live streaming scenario described in [43] was used to assess the aforementioned algorithms. In particular, the following five-dependency layer H.264/SVC encoder configuration was used for the baseline VBR controller:

- a) Number of pictures: 900.
- b) GoP size/Intra period: 8/32 pictures.
- c) GoP structure: hierarchical B pictures.
- d) Search range for motion estimation: 16×16 pixels.
- e) Number of dependency layers: $D = 5$.
 - i) $d = 0$: QCIF, $f_{out}^{(0,1)} = 6.25$ Hz ($T^{(0)} = 2$).
 - ii) $d = 1$: QCIF, $f_{out}^{(1,2)} = 12.5$ Hz ($T^{(1)} = 3$).
 - iii) $d = 2$: CIF, $f_{out}^{(2,2)} = 12.5$ Hz ($T^{(2)} = 3$).

¹CQP encoding means that every temporal layer within a spatial or quality layer shares the same QP value, while the QP value of each spatial or quality layer can be different in order to reach the pre-established target bit rate.

- iv) $d=3$: CIF, $f_{out}^{(3,2)} = 12.5$ Hz ($T^{(3)} = 3$).
v) $d=4$: CIF, $f_{out}^{(4,3)} = 25$ Hz ($T^{(4)} = 4$).

f) Symbol mode: CAVLC.

The RC parameters were set as follows: target buffer fullness $nTF = 50\%$ and buffer size $BD = 3$ s. Henceforth, we will refer to this SVC configuration as *baseline configuration* (BC) and to the rate-controlled SVC (RC-SVC) as *single-buffer BC* (SB-BC).

For the proposed IL-MB VBR controller, the following three-dependency layer H.264/SVC encoder configuration was used:

- a) Number of pictures: 900.
b) GoP size/Intra period: 8/32 pictures.
c) GoP structure: hierarchical B pictures.
d) Search range for motion estimation: 16×16 pixels.
e) Number of dependency layers: $D = 3$.
i) $d=0$: QCIF, $f_{out}^{(0,2)} = 12.5$ Hz ($T^{(0)} = 3$).
ii) $d=1$: CIF, $f_{out}^{(1,2)} = 12.5$ Hz ($T^{(1)} = 3$).
iii) $d=2$: CIF, $f_{out}^{(2,3)} = 25$ Hz ($T^{(2)} = 4$).

f) Symbol mode: CAVLC.

We will refer to this SVC encoder configuration as *compact configuration* (CC) since it consists of only three layers in comparison to BC, which is made of five layers. The RC parameters took the following values: $nTF = 50\%$ and $BD = 3$ s., the same as for SB-BC, and $t_{min}^{(0)} = 1$, $t_{min}^{(1)} = 2$, and $t_{min}^{(2)} = 2$. As can be observed, $t_{min}^{(0)}$ and $t_{min}^{(2)}$ were set such that HRD-compliant sub-streams for QCIF@6.25 Hz ($d = 0$) and high-quality (HQ) CIF@12.5 Hz ($d = 2$) were available, as for SB-BC. Henceforth, this RC-SVC encoder will be referred to as MB-CC.

Furthermore, in order to analyze the behavior of the proposed VBR controller if only one buffer per dependency layer was controlled (that corresponding to the highest frame rate), an additional H.264/SVC encoder and RC configuration with $t_{min}^{(d)} = t_{max}^{(d)}$ for every dependency layer was also studied. We will refer to it as SB-CC.

Two sets of video sequences at 25 Hz exhibiting a variety of complexities were used in our experiments. The first set consisted of four well-known test sequences recommended in [49] for streaming applications: *Bus*, *Football*, *Foreman*, and *Mobile*. These sequences were concatenated to themselves several times to reach the aforementioned number of pictures. The second set consisted of three sequences displaying scene changes: *Soccer-Mobile-Foreman*, *Spiderman* (movie), and *The Lord of the Rings* (movie). *Soccer-Mobile-Foreman* was formed by concatenating 300 frames of each sequence. The other two were extracted from HQ Digital Versatile Disks and downsampled to either QCIF or CIF format, and have been made available on-line in [50]. They show many scene cuts, so they are challenging from the RC point of view.

All the sequences were encoded using the set of constant QP values that best approached some pre-established target bit rates. We will refer to this RC-SVC encoder as CQP-CC. For the first group of sequences, the target bit rates for the highest temporal resolution of each layer d , i.e., QCIF@12.5 Hz (0, 2), low-quality (LQ) CIF@12.5 Hz (1, 2) and HQ CIF@25 Hz

TABLE II
TARGET BIT RATES ASSIGNED TO EACH LAYER OF THE COMPARED RC-SVC ENCODERS

Layer (d,t)	RC-SVC Encoder	Resolution	Assigned $R^{(d,t)}$ from CQP-CC
(0,1)	SB-BC	QCIF@6.25 Hz	$R_{out}^{(0,1)}$
-	-		
(0,1)	MB-CC		
(1,2)	SB-BC	QCIF@12.5 Hz	$R_{out}^{(0,2)}$
(0,2)	SB-CC		
(0,2)	MB-CC		
(2,2)	SB-BC	LQ CIF@12.5 Hz	$R_{out}^{(1,2)}$
(1,2)	SB-CC		
(1,2)	MB-CC		
(3,2)	SB-BC	HQ CIF@12.5 Hz	$R_{out}^{(2,2)}$
-	-		
(2,2)	MB-CC		
(4,3)	SB-BC	HQ CIF@25 Hz	$R_{out}^{(2,3)}$
(2,3)	SB-CC		
(2,3)	MB-CC		

TABLE III
AVERAGE RESULTS ACHIEVED BY THE SB-BC, THE SB-CC, AND THE PROPOSED MB-CC VBR CONTROLLERS. INCREMENTAL RESULTS ARE GIVEN WITH RESPECT TO CQP-CC ENCODING

Layer (d,t)	RC-SVC encoder	$\Delta\mu_{PSNR}$ (dB)	$\Delta\sigma_{PSNR,j}$ (dB)	Bit Rate Error (%)	#O/#U	μ_V (%)
(0,1)	SB-BC	-0.12	0.09	1.00	0/0	52.34
(0,1)	SB-CC	0.05	0.22	2.68	5/0	59.90
(0,1)	MB-CC	0.05	0.19	1.48	0/0	55.72
(1,2)	SB-BC	-0.25	0.15	1.86	1/0	64.77
(0,2)	SB-CC	0.10	0.19	0.94	0/0	55.09
(0,2)	MB-CC	0.08	0.16	0.84	0/0	54.66
(2,2)	SB-BC	-0.16	0.11	0.94	0/0	52.98
(1,2)	SB-CC	0.00	0.09	0.93	0/0	55.26
(1,2)	MB-CC	0.00	0.09	1.02	0/0	55.19
(3,2)	SB-BC	-0.10	0.07	0.59	0/0	52.42
(2,2)	SB-CC	0.05	0.12	1.45	0/0	56.42
(2,2)	MB-CC	0.06	0.11	0.79	0/0	54.17
(4,3)	SB-BC	-0.21	0.06	1.57	0/0	64.82
(2,3)	SB-CC	0.09	0.11	0.48	0/0	54.02
(2,3)	MB-CC	0.08	0.10	0.51	0/0	53.43

(2, 3) were those suggested in [49] for the spatial/CGS testing scenario. For the second group, the following medium-quality target bit rates associated with the highest temporal resolution of each layer d were selected: 96 (0, 2), 192 (1, 2), and 512 kbps (2, 3). The output bit rates $R_{out}^{(d,t)}$ generated by the CQP-CC encoding for the five target spatio-temporal resolutions were used as target bit rates $R^{(d,t)}$ for the three assessed RC-SVC encoders, i.e.: SB-BC, SB-CC, and MB-CC. The same target bit rates were assigned to each involved spatio-temporal layer for all the RC-SVC encoders so that all the compared encoders operated under the same bit rate constraints. The actual $R^{(d,t)}$ values are listed in Table II. It should be noted that the low temporal resolution for both QCIF and HQ CIF layers is not rate-controlled in SB-CC.

B. Experimental Results and Discussion

In order to assess the performance of the proposed IL-MB VBR controller from the quality point of view, the average luminance peak SNR (PSNR) μ_{PSNR} was used. The Bjøntegaard recommendation [51] was followed to properly

TABLE IV

PERFORMANCE COMPARISON AMONG THE SB-BC, THE SB-CC, AND THE PROPOSED MB-CC VBR CONTROLLERS, FOR A SPECIFIC STATIONARY COMPLEXITY VIDEO SEQUENCE, *Bus*. THE RESULTS ACHIEVED BY CQP-CC ENCODING HAVE ALSO BEEN INCLUDED FOR REFERENCE

Layer (d,t)	R ^(d,t) (kbps)	RC-SVC Scheme	μ_{PSNR} (dB)	$\bar{\sigma}_{PSNR,j}$ (dB)	Bit Rate Error (%)	#O/#U	μ_V (%)
(0,1)	73.89	CQP-CC	31.24	0.31	-	0/0	51.97
(0,1)		SB-BC	31.24	0.31	-0.03	0/0	51.89
(0,1)		SB-CC	31.23	0.39	0.95	0/0	56.07
(0,1)		MB-CC	31.22	0.40	0.96	0/0	55.53
(0,2)	101.61	CQP-CC	31.11	0.27	-	0/0	52.70
(1,2)		SB-BC	31.00	0.32	1.27	0/0	63.15
(0,2)		SB-CC	31.10	0.35	0.52	0/0	53.94
(0,2)		MB-CC	31.10	0.36	0.50	0/0	52.88
(1,2)	202.67	CQP-CC	26.94	0.16	-	0/0	51.91
(2,2)		SB-BC	26.86	0.19	-0.09	0/0	51.03
(1,2)		SB-CC	26.92	0.23	0.26	0/0	53.44
(1,2)		MB-CC	26.92	0.23	0.44	0/0	53.15
(2,2)	404.97	CQP-CC	30.01	0.19	-	0/0	52.01
(3,2)		SB-BC	29.99	0.19	-0.04	0/0	52.15
(2,2)		SB-CC	30.01	0.25	0.40	0/0	54.76
(2,2)		MB-CC	30.02	0.24	0.53	0/0	53.66
(2,3)	517.67	CQP-CC	30.05	0.17	-	0/0	52.20
(4,3)		SB-BC	29.91	0.18	1.5	0/0	65.57
(2,3)		SB-CC	30.05	0.22	0.31	0/0	54.43
(2,3)		MB-CC	30.06	0.21	0.46	0/0	53.41

compare the μ_{PSNR} values obtained by the compared algorithms. The average results over all the test video sequences are summarized in Table III in terms of PSNR increments $\Delta\mu_{PSNR}$ with respect to CQP-CC encoding. Three rows per spatio-temporal layer are shown, one for each assessed RC-SVC encoder. As can be observed, the average PSNR achieved by SB-CC and MB-CC at every spatio-temporal layer were similar to that of CQP-CC and higher than that of SB-BC, which, for the same target bit rate $R^{(d,t)}$, is encoding more layers.

A detailed comparison of the algorithms is shown in Tables IV and V. Table IV shows the results achieved for *Bus*, a representative example of video sequence with stationary complexity, and Table V shows the results for *The Lord of the Rings*, a representative example of video sequence with scene changes. The results in terms of average PSNR indicate that, for non-stationary complexity sequences, the performance of either SB-CC or MB-CC improved that of the nearly constant quality system at most spatio-temporal layers. However, for stationary complexity sequences, the performance achieved by the three VBR controllers were very close to that of the nearly constant quality system.

Representative behaviors of the encoder buffer occupancy, PSNR and QP time evolutions corresponding to the two lower spatio-temporal resolutions, QCIF@6.25 Hz and QCIF@12.5 Hz, are depicted in Figs. 5 and 6 for *Bus*, and Figs. 7 and 8 for *The Lord of the Rings*, where the QCP-CC plots have been removed for clarity reasons. High quality plots including those of CQP-CC encoding can be found in [50] for every spatio-temporal resolution. As can be shown, in the stationary scenario the three assessed VBR controllers were able to keep the QP fluctuation low most of the time, thus providing a nearly constant PSNR time evolution. However, some high buffer levels and QP fluctuations were observed

TABLE V

PERFORMANCE COMPARISON AMONG THE SB-BC, THE SB-CC, AND THE PROPOSED MB-CC VBR CONTROLLERS, FOR A SPECIFIC NON-STATIONARY COMPLEXITY VIDEO SEQUENCE, *The Lord of the Rings*. THE RESULTS ACHIEVED BY CQP-CC ENCODING HAVE ALSO BEEN INCLUDED FOR REFERENCE

Layer (d,t)	R ^(d,t) (kbps)	RC-SVC Scheme	μ_{PSNR} (dB)	$\bar{\sigma}_{PSNR,j}$ (dB)	Bit Rate Error (%)	#O/#U	μ_V (%)
(0,1)	66.50	CQP-CC	34.45	0.66	-	42/48	49.70
(0,1)		SB-BC	34.40	0.91	2.31	0/0	53.89
(0,1)		SB-CC	34.75	0.96	5.53	36/0	70.17
(0,1)		MB-CC	34.77	0.94	1.70	0/0	62.76
(0,2)	93.99	CQP-CC	34.36	0.66	-	104/113	46.71
(1,2)		SB-BC	34.23	0.99	2.59	0/0	60.90
(0,2)		SB-CC	34.80	0.97	1.05	0/0	54.38
(0,2)		MB-CC	34.76	0.94	0.20	0/0	50.47
(1,2)	186.51	CQP-CC	32.87	0.90	-	98/113	47.26
(2,2)		SB-BC	32.77	1.09	1.90	0/0	52.02
(1,2)		SB-CC	33.15	1.08	1.00	0/0	55.21
(1,2)		MB-CC	33.12	1.07	1.18	0/0	55.88
(2,2)	385.34	CQP-CC	35.25	0.83	-	93/114	45.16
(3,2)		SB-BC	35.31	0.94	1.73	0/0	51.98
(2,2)		SB-CC	35.54	1.00	3.24	0/0	71.54
(2,2)		MB-CC	35.58	0.94	0.92	0/0	58.58
(2,3)	507.26	CQP-CC	35.29	0.81	-	217/241	45.11
(4,3)		SB-BC	35.27	0.95	2.26	0/0	58.81
(2,3)		SB-CC	35.69	0.99	0.42	0/0	53.58
(2,3)		MB-CC	35.67	0.94	0.04	0/0	49.95

at certain time instants for SB-BC (see Fig. 6) because more layers were encoded for a given target bit rate. In the non-stationary scenario the three assessed algorithms made, with some exceptions that will be discussed, a proper use of the buffer fullness to provide PSNR and QP evolutions closer to those of the nearly constant quality system, as expected for VBR control algorithms, given that larger amount of bits were assigned to more complex scenes. The undesirable buffer levels observed in the SB-CC VBR controller at the layer (0,1) (see Fig. 7) were due to the fact that only the highest temporal resolution buffer associated with the layer (0,2) was considered for QP estimation. Furthermore, as in the stationary scenario, some undesirable buffer levels and QP fluctuations also happened at the highest temporal resolution sub-stream for SB-BC (see Fig. 8), again due to the fact that it is coding more layers.

From the quality consistency point of view, the performance of the VBR controllers was also assessed by means of a time-local version of the PSNR standard deviation, denoted as $\bar{\sigma}_{PSNR,j}$, which attempts to measure the quality consistency within a scene by reducing the impact of the scene cuts on the PSNR standard deviation (the reader is referred to [43] for details). Thus, a low value of $\bar{\sigma}_{PSNR,j}$ indicates good quality consistency, and vice versa. The average results over all the test video sequences in terms of $\bar{\sigma}_{PSNR,j}$ increment with respect to CQP-CC encoding, $\Delta\bar{\sigma}_{PSNR,j}$, are provided in Table III. As can be seen, the three VBR controllers achieved a quality consistency close to that of CQP-CC encoding. Furthermore, the $\bar{\sigma}_{PSNR,j}$ differences among them were not significant either in particular stationary (see Table IV) or non-stationary scenarios (see Table V), as expected, since the VBR controllers were specially designed to provide consistent-quality scalable sub-streams.

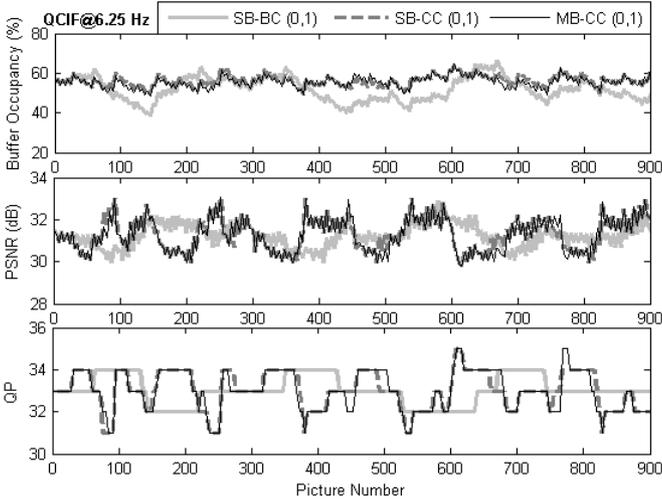


Fig. 5. Encoder buffer occupancy, PSNR, and QP time evolutions corresponding to the spatio-temporal resolution QCIF@6.25 Hz for *Bus*. A high-quality plot is available on-line in [50].

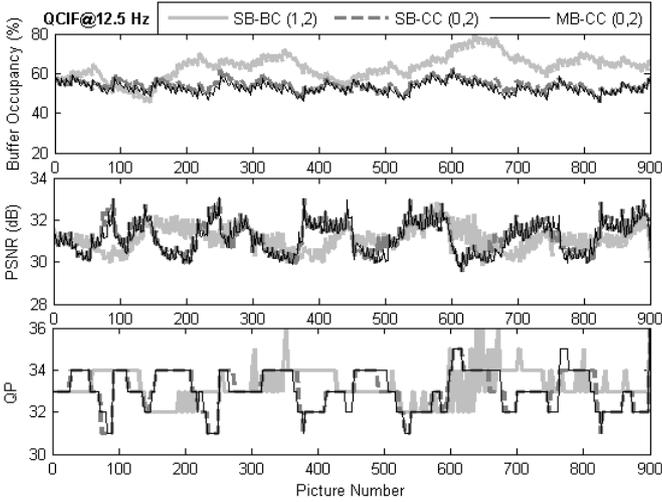


Fig. 6. Encoder buffer occupancy, PSNR, and QP time evolutions corresponding to the spatio-temporal resolution QCIF@12.5 Hz for *Bus*. A high-quality plot is available on-line in [50].

The VBR controllers were also quantitatively compared in terms of target bit rate adjustment and buffer level behavior. To this end, the following metrics were employed: output bit rate error with respect to that of CQP-CC encoding, number of pictures in which either an overflow (#O) or an underflow (#U) occurred, and mean buffer level (μ_V). As can be seen in Table III, the average output bit rate errors achieved by the three VBR controllers at every spatio-temporal layer were generally below 2%, that is the maximum bit rate error recommended in [49] for the spatial/CGS testing scenario. Nevertheless, in some sequences with time-varying complexity, such as *The Lord of the Rings*, higher bit rate errors occurred in some spatio-temporal layers for the SB-BC and SB-CC VBR controllers (see Table V). Specifically, for the SB-BC VBR controller, such bit rate mismatches together with the large μ_V values observed in layers (1, 2) and (4, 3) indicate that the corresponding target bit rates were not high

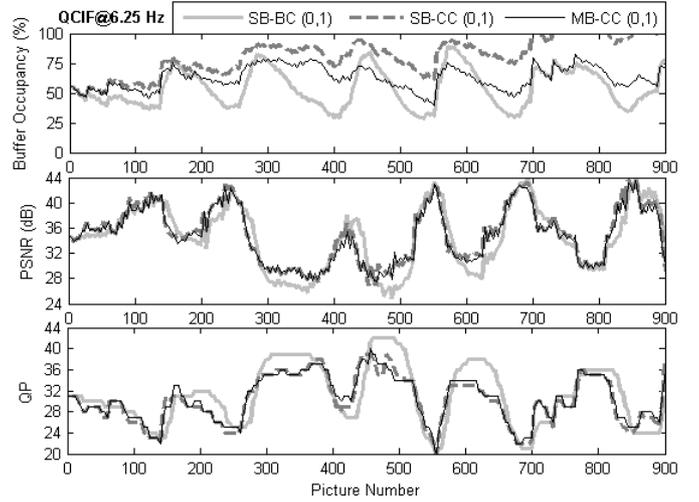


Fig. 7. Encoder buffer occupancy, PSNR, and QP time evolutions corresponding to the spatio-temporal resolution QCIF@6.25 Hz for *The Lord of the Rings*. A high-quality plot is available on-line in [50].

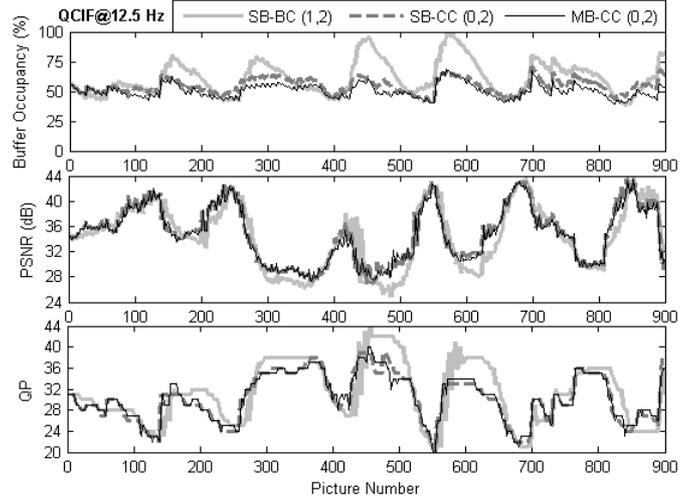


Fig. 8. Encoder buffer occupancy, PSNR, and QP time evolutions corresponding to the spatio-temporal resolution QCIF@12.5 Hz for *The Lord of the Rings*. A high-quality plot is available on-line in [50].

enough to encode all the spatio-temporal layers. For the SB-CC VBR controller, the results in terms of bit rate error, mean buffer level, and number of overflows shown in Table V for layers (0, 1) (see also Fig. 7) and (2, 2), proved the need of simultaneously controlling all the involved buffers within a dependency layer, as in the MB-CC VBR controller, which was able to prevent overflow and underflow in all the encodings (see Tables III–V).

From the complexity point of view, although the computational cost of the IL-MB VBR controller is slightly higher than that of its baseline version, the increment is clearly justified by two facts: 1) the computational cost of the baseline rate controller proved to be significantly lower than those of conventional approaches [43]; therefore, there is room enough to allocate some moderate increment as the one proposed; and 2) the IL-MB approach actually reduces the number of spatio-temporal layers to be encoded; for example, in the simulated

mobile live streaming scenario, MB-CC uses three dependency layers instead of five (used by SB-BC) for delivering HRD-compliant video content to five target terminals, thus substantially improving the overall coding efficiency.

It should be noticed that the good performance achieved by the proposed MB-CC VBR controller, specifically at the lowest and the highest dependency layers, could be partly due to the fact that the total bit rate per dependency layer was optimally distributed among temporal layers since the corresponding $R^{(d,t)}$ values were previously obtained using CQP-CC encoding. In real-time video coding applications, the optimal distribution of the target bit rate among temporal layers is not known in advance because it depends on the video content. For instance, the target bit rate for a sequence with high spatial detail but low motion content should be shared out among temporal layers such that the bit resources are mainly allocated to K pictures. However, for a sequence with medium-low spatial detail but high motion content, a more balanced target bit rate distribution between K and NK pictures is desirable to encode better the motion information.

In order to explore the sensitivity of the proposed MB-CC VBR controller to target bit rate deviations with respect to those obtained by CQP-CC encoding, we performed an “ad hoc” experiment. This experiment involved modifying the target bit rates of the low temporal resolutions of those layers encoded using the IL-MB approach. In particular, assuming that the target bit rates for the highest temporal resolutions can be set in advance following, for instance, the recommendation in [49], the target bit rates for QCIF@6.25 Hz (0, 1) and HQ CIF@12.5 Hz (2, 2) were deviated $\pm 2\%$, $\pm 5\%$, and $\pm 10\%$ from their corresponding reference target bit rates. The average results over all the test video sequences in terms of $\Delta\mu_{PSNR}$, $\Delta\bar{\sigma}_{PSNR,j}$, and bit rate error with respect to those achieved without $R^{(d,t)}$ deviations, as well as the number of overflows and underflows and mean buffer level, are summarized in Table VI. As can be observed, target bit rate deviations of 10% led to noticeable loss of quality consistency (due to the increase of QP fluctuations caused by the sub-optimal target bit rates), bit rate errors above 2%, and mean buffer levels close to either overflow or underflow.

It is also interesting to notice how the sub-optimal distribution of the target bit rate affects to the buffer levels of the involved temporal layers. To this end, let us focus on the results from layers (0, {1, 2}) for an $R^{(d,t)}$ deviation of +10%. As can be observed, the corresponding μ_V took opposite values: the low temporal resolution buffer was close to underflow, while the high temporal resolution buffer was close to overflow. This mirror-like behavior of the buffers is due to the fact that the buffer modeling stage averages the current encoding states of the involved temporal resolutions at many time instants for $nV^{(0)}$, $nAU^{(0)}$ and $QP_{REF}^{(0)}$ computation. Although optimum adjustment to $R^{(0,\{1,2\})}$ or nTF were not achieved, neither overflows nor underflows occurred in most of the assessed video sequences. However, if the highest temporal resolution buffer was only considered for QP estimation (as in SB-CC), a suitable adaptation to both $R^{(0,2)}$ and nTF would be achieved at the expense of a higher underflow risk at the lowest temporal resolution buffer. In short, when the target bit

TABLE VI
AVERAGE RESULTS ACHIEVED BY THE PROPOSED MB-CC VBR CONTROLLER FOR DIFFERENT TARGET BIT RATE DEVIATIONS AT LAYERS (0, 1) AND (2, 2). INCREMENTAL RESULTS ARE GIVEN WITH RESPECT TO THOSE ACHIEVED BY CQP-CC ENCODING

Layer (d,t)	$R^{(d,t)}$ Dev. (%)	$\Delta\mu_{PSNR}$ (dB)	$\Delta\bar{\sigma}_{PSNR,j}$ (dB)	Bit Rate Error (%)	#O/#U	μ_V (%)
(0,1)	+10	0.80	0.18	1.60	0/0	31.90
	+5	0.44	0.02	1.05	0/0	41.56
	+2	0.14	0.00	0.89	0/0	49.93
	-2	-0.28	-0.01	1.75	0/0	59.36
	-5	-0.62	0.03	2.17	0/0	64.51
	-10	-1.20	0.14	2.69	1/0	71.01
(0,2)	0	-0.04	0.22	2.15	1/0	67.42
		0.03	0.04	1.52	0/0	61.16
		0.00	0.01	1.06	0/0	57.29
		-0.08	0.00	0.66	0/0	51.78
		-0.17	0.03	0.72	0/0	48.84
		-0.34	0.17	0.74	0/0	45.09
(1,2)	0	-0.07	0.02	0.75	0/0	55.30
		-0.04	0.01	0.97	0/0	55.23
		-0.04	0.01	0.84	0/0	55.18
		-0.06	0.00	0.91	0/0	54.98
		-0.09	-0.01	0.77	0/0	54.59
		-0.16	-0.02	0.70	0/0	54.88
(2,2)	+10	0.81	0.24	2.29	0/0	29.98
	+5	0.47	0.04	1.29	0/0	39.88
	+2	0.17	0.01	0.42	0/0	48.80
	-2	-0.27	-0.01	1.34	0/0	58.38
	-5	-0.61	0.02	1.88	0/0	64.87
	-10	-1.18	0.17	2.47	0/0	72.20
(2,3)	0	-0.01	0.21	1.87	0/0	67.68
		0.05	0.03	1.28	0/0	61.34
		0.01	0.01	0.78	0/0	57.04
		-0.08	0.00	0.51	0/0	49.75
		-0.17	0.03	0.60	0/0	47.06
		-0.33	0.21	0.77	0/0	43.25

rate distributions among temporal resolutions are not optimally distributed, the proposed method for IL-MB control makes its best to provide a good tradeoff between quality consistency and buffer control in all the involved buffers.

V. CONCLUSIONS AND FURTHER WORK

In this paper a novel IL-MB approach built on top of a baseline VBR controller for H.264/SVC has been proposed. Given a dependency layer, our proposal aims to deliver HRD-compliant sub-streams with different temporal resolutions. In doing so, temporal scalability is fully exploited by reducing the number of dependency layers required to provide the same spatial or quality level for decoding terminals requiring different frame rates. For this purpose, the proposed IL-MB VBR controller estimates, on a frame basis, the most proper QP value such that the virtual buffers, each one associated with a temporal resolution of the same dependency layer, are maintained at secure levels, while minimizing the distortion of the corresponding sub-streams. Furthermore, the decision rules suggested for simultaneously controlling the set of virtual buffers might be used in any other RC algorithm for SVC.

In order to guarantee robust performance, the proposed IL-MB framework requires proper target bit rates for the lower temporal resolution sub-streams to be known in advance. An effective method to estimate such target bit rates is left for future work.

APPENDIX A
RBF NETWORK DESIGN

The methodology described in [43] was followed to find the most suitable RBF network parameters for both K and NK pictures. This methodology may be structured in three stages: *training data generation*, *training process* and *parameter selection process*. These stages are summarized in the sequel.

1) *Training Data Generation*

The first stage focuses on the extraction of a training data set consisting of pairs *input vector-desired output*, i.e.: $\{\mathbf{X}^{(d)}, \Delta QP^{*(d)}\}$. To this end, a representative set of video sequences exhibiting a large variety of spatio-temporal contents was employed and some of their GoPs were encoded using Φ different configurations involving several encoder- and RC-related parameters: number of dependency layers, spatial resolutions, GoP size, target bit rate, minimum available temporal layer identifier, target buffer level, and buffer size.

Given an input vector $\mathbf{X}^{(d)}$ extracted from a picture with identifier (d, t') encoded with a configuration ϕ at the time instant $(j-1)$, the goal was to find, from a set of Q quantization increments $\{\Delta QP_q^{(d)}\}_{q=1, \dots, Q}$, the most appropriate QP increment $\Delta QP^{*(d)}$ for the next picture with identifier (d, t) so that each involved buffer k , with $k = \max[t_{min}^{(d)}, t] \dots t_{max}^{(d)}$, was maintained at secure levels, while minimizing the coding distortion of the corresponding sub-streams. Specifically, $t_{min}^{(d)}$ was fixed to $t_{max}^{(d)} - 2$ so that three buffers (at most) could be simultaneously controlled in an IL-MB framework².

To satisfy these buffer and distortion constraints, the $\Delta QP_q^{(d)}$ value that minimized certain cost function Ψ was chosen as optimum:

$$\Delta QP^{*(d)} = \operatorname{argmin}_{\Delta QP_q^{(d)}} \Psi(\Delta QP_q^{(d)}). \quad (19)$$

The proposed cost function, which was designed “ad hoc” for this problem, balances three conflicting factors: quality consistency, buffer control, and QP consistency. Specifically, Ψ obeys:

$$\begin{aligned} \Psi(\Delta QP_q^{(d)}) = & \lambda_1 \theta \left(\frac{\sum_k \frac{D_j^{(d)} - \bar{D}^{(d,k)}}{255}}{t_{max}^{(d)} - \max[t_{min}^{(d)}, t] + 1} \right)^2 + \\ & \lambda_2 \left(\frac{\sum_k \left(\frac{V_{j+1}^{(d,k)}}{BD_\phi \times R_\phi^{(d,k)}} - nTF_\phi \right)}{t_{max}^{(d)} - \max[t_{min}^{(d)}, t] + 1} \right)^2 + \\ & \lambda_3 \left(\frac{\Delta QP_q^{(d)}}{\Delta QP_{MAX}^{(d)}} \right)^2. \end{aligned} \quad (20)$$

²For a sequence frame rate of 25 Hz, $t_{min}^{(d)} = t_{max}^{(d)} - 2$ means that the minimum output frame rate of encoded video ensuring the HRD constraints is the fourth part (6.25 Hz), which is a sufficient temporal resolution in practical SVC applications [49].

The first term monitors the quality consistency by means of the squared mean of the differences between the distortion $D_j^{(d)}$ of the current picture and the average distortion $\bar{D}^{(d,k)}$ of each sub-stream (d, k) . The distortion metric used was the *mean of absolute error* between the original and reconstructed luminance pictures. Furthermore, θ is a scaling factor so that the dynamic range of this term was similar to the remaining terms. In particular, θ was set to 100 in our experiments.

The second term considers the buffer control through the squared mean of the differences between the normalized current buffer level $V_{j+1}^{(d,k)} / BD_\phi \times R_\phi^{(d,k)}$ associated with each sub-stream (d, k) and the normalized target buffer fullness nTF_ϕ .

The third term watches over the QP consistency by means of the squared ratio between the considered QP increment and the maximum allowed QP increment $\Delta QP_{MAX}^{(d)}$.

Finally, the weight vector $(\lambda_1, \lambda_2, \lambda_3)^T$ is meant to establish a proper tradeoff among the considered conflicting factors.

2) *RBF Network Training and Parameter Selection*

To consider different tradeoffs among the three terms of the cost function for data labeling, a reduced set of tentative weight vectors was previously selected. Subsequently, for each pre-established weight vector, two training data sets, one for K pictures and the other for NK pictures, were generated. Each RBF network was trained several times considering each one of the pre-established weight vectors, different random initializations, and different numbers L of radial basis functions. For this purpose, a training algorithm based on Gaussian processes (GP) [52] was used because it provides a robust solution for the parameters that relies on maximizing a marginal likelihood. In particular, the sparse approximation GP toolbox for Matlab [53] due to Snelson and Gharahmani [54] was used.

Finally, the validation process for parameter selection led us to select the weight vectors $(0.90, 0.09, 0.01)^T$ and $(0.75, 0.24, 0.01)^T$ for K- and NK-picture RBF networks, respectively, as well as a total of 10 Gaussian-type functions for each RBF network. Specifically, their centers, widths and weights are the following (also available on line in electronic format in [50]):

i. *K-picture RBF parameters*

$$w_0 = -2.11439, \quad \mathbf{w} = \begin{pmatrix} -947.17558 \\ 17.91979 \\ 99.79803 \\ -119.72409 \\ 87.53142 \\ 14.05907 \\ 60.23655 \\ -797.73548 \\ 1605.45805 \\ -82.09706 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} 0.92748 \\ 3.20320 \\ 1.23924 \\ 8.84978 \end{pmatrix},$$

$$C = \begin{pmatrix} 0.43803 & 1.27831 & 0.13142 & 2.61346 \\ 0.76851 & 1.13763 & 0.65991 & 2.79565 \\ -0.75232 & 0.79498 & 1.60194 & 1.76489 \\ -1.23805 & -0.62409 & -0.45549 & 2.01148 \\ 0.26089 & 2.77186 & 0.38882 & 0.19505 \\ 0.66948 & 3.32571 & 0.31369 & 2.04133 \\ 0.92787 & 1.04185 & -0.27238 & 1.67820 \\ 0.29267 & 1.88389 & 0.28556 & 2.79760 \\ 0.35347 & 1.49620 & 0.20293 & 2.77878 \\ -0.39515 & 0.50965 & 1.25654 & 0.14932 \end{pmatrix}.$$

ii. *NK-picture RBF parameters*

$$w_0 = -0.25419, \quad \mathbf{w} = \begin{pmatrix} 12511.56054 \\ -54.23826 \\ -30.39539 \\ 26.81211 \\ -4.73220 \\ -16.14186 \\ -12507.11582 \\ 4.66150 \\ 11.06643 \\ 0.66867 \end{pmatrix}, \quad \Sigma = \begin{pmatrix} 0.59234 \\ 2.05179 \\ 1.00957 \\ 3.00504 \end{pmatrix},$$

$$C = \begin{pmatrix} 0.19710 & 1.71061 & 0.12047 & 3.04580 \\ -0.67315 & -0.68530 & -0.17373 & 1.42105 \\ 0.39981 & -0.66020 & 0.89182 & -0.90448 \\ 0.58803 & 1.82533 & 0.24637 & -0.95955 \\ 0.66092 & 0.77316 & 0.57093 & 3.35614 \\ 0.70296 & 1.74486 & -0.15198 & 0.65384 \\ 0.19696 & 1.71090 & 0.12112 & 3.04637 \\ 0.88774 & 0.42078 & 0.61288 & 1.74001 \\ 0.92236 & 2.50876 & 0.15902 & 2.95167 \\ -0.12642 & 0.67930 & 0.67757 & 1.23198 \end{pmatrix}.$$

REFERENCES

- [1] G. Liebl, M. Wagner, J. Pandel, and W. Weng, "An RTP payload format for erasure-resilient transmission of progressive multimedia streams," *Document draft-ietf-avt-uxp-07.txt*, Oct. 2004.
- [2] R. Schaefer, H. Schwarz, D. Marpe, T. Schierl, and T. Wiegand, "MCTF and scalability extension of H.264/AVC and its application to video transmission, storage, and surveillance," in *Proceedings of VCIP 2005, Peking, China*, July 2005, pp. 596 011: 1–12.
- [3] T. Wiegand, L. Noblet, and F. Rovati, "Scalable video coding for IPTV services," *Broadcasting, IEEE Transactions on*, vol. 55, pp. 527–538, June 2009.
- [4] Joint Video Team (JVT), "Advanced video coding for generic audiovisual services," *ITU-T Recommendation International Standard of Joint Video Specification, ITU-T Rec. H.264/ISO/IEC 14496-10 AVC, Version 1, JVT-G50*, May 2003.
- [5] ISO/IEC, "Generic coding of moving pictures and associated audio information - Part 2: Video," *ITU-T Recommendation H.262-ISO/IEC 13818-2, MPEG-2*, Nov. 1994.
- [6] ITU-T, "Video coding for low bitrate communication," *ITU-T Draft Recommendation H.263 Version 1*, Nov. 1995.
- [7] ISO/IEC, "Coding of audio-visual objects - Part 2: Visual," *ISO/IEC 14496-2, MPEG-4 Visual Version 1*, Apr. 1999.
- [8] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, no. 9, pp. 1103–1120, Sept. 2007.
- [9] M. Wien, H. Schwarz, and T. Oelbaum, "Performance analysis of SVC," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, no. 9, pp. 1194–1203, Sept. 2007.
- [10] J. Ribas-Corbera, P. Chou, and S. Regunathan, "A generalized hypothetical reference decoder for H.264/AVC," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 7, pp. 674–687, 2003.
- [11] T. Lakshman, A. Ortega, and A. Reibman, "VBR video: tradeoffs and potentials," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 952–973, 1998.
- [12] A. Ortega, "Variable bit-rate video coding," in *Compressed Video over Networks, M.-T. Sun and A. R. Reibman, Eds. New York: Marcel Dekker*, pp. 343–382, 2000.
- [13] H.-M. Hang and J.-J. Chen, "Source model for transform video coder and its application. I. Fundamental theory," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 7, no. 2, pp. 287–298, 1997.
- [14] J. Ribas-Corbera and S. Lei, "Rate control in DCT video coding for low-delay communications," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 9, no. 1, pp. 172–185, 1999.
- [15] B. Tao, B. Dickinson, and H. Peterson, "Adaptive model-driven bit allocation for MPEG video coding," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 10, no. 1, pp. 147–157, Feb. 2000.
- [16] ISO/IEC, "MPEG Test Model 5," *ISO/IEC JTC/SC29/WG11, MPEG Test Model 5*, April 1993.
- [17] S. Ma, W. Gao, and Y. Lu, "Rate-distortion analysis for H.264/AVC video coding and its application to rate control," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 15, no. 12, pp. 1533–1544, 2005.
- [18] T. Chiang and Y.-Q. Zhang, "A new rate control scheme using quadratic rate distortion model," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 7, no. 1, pp. 246–250, Feb. 1997.
- [19] S. Ma, Z. Li, and F. We, "Proposed draft of adaptive rate control," *JVT-H017, 8th JVT Meeting*, Geneva, Switzerland, May 2003.
- [20] B. Xie and W. Zeng, "A sequence-based rate control framework for consistent quality real-time video," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 16, no. 1, pp. 56–71, 2006.
- [21] Z. Chen and K. N. Ngan, "Towards rate-distortion tradeoff in real-time color video coding," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, no. 2, pp. 158–167, Feb. 2007.
- [22] D.-K. Kwon, M.-Y. Shen, and C.-C. J. Kuo, "Rate control for H.264 video with enhanced rate and distortion models," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, no. 5, pp. 517–529, 2007.
- [23] Z. He, Y. K. Kim, and S. Mitra, "Low-delay rate control for DCT video coding via ρ -domain source modeling," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 11, no. 8, pp. 928–940, 2001.
- [24] N. Kamaci, Y. Altunbasak, and R. Mersereau, "Frame bit allocation for the H.264/AVC video coder via Cauchy-density-based rate and distortion models," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 15, no. 8, pp. 994–1006, 2005.
- [25] S. Sanz-Rodríguez, O. del-Ama-Esteban, M. de-Frutos-López, and F. Díaz-de-María, "Cauchy-density-based basic unit layer rate controller for H.264/AVC," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 8, pp. 1139–1143, Aug. 2010.
- [26] L. Xu, W. Gao, X. Ji, D. Zhao, and S. Ma, "Rate control for spatial scalable coding in SVC," in *Picture Coding Symposium, 2007. PCS 2007*, Nov. 2007.
- [27] Y. Liu, Z. G. Li, and Y. C. Soh, "Rate control of H.264/AVC scalable extension," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 1, pp. 116–121, Jan. 2008.
- [28] A. Leontaris and A. M. Tourapis, "Rate control for the Joint Scalable Video Model (JSVM)," *Video Team of ISO/IEC MPEG and ITU-T VCEG, JVT-W043, San Jose, California*, April 2007.
- [29] Y. Pitrey, M. Babel, and O. Deforges, "One-pass bitrate control for MPEG-4 scalable video coding using ρ -domain," *Broadband Multimedia Systems and Broadcasting, 2009. BMSB '09. IEEE International Symposium on*, pp. 1–5, May 2009.
- [30] M. Liu, Y. Guo, H. Li, and C.-W. Chen, "Low-complexity rate control based on ρ -domain model for scalable video coding," in *Image Processing, 2010. ICIP 2010. IEEE International Conference on*, Sept. 2010, pp. 1277–1280.
- [31] Y. Cho, J. Liu, D.-K. Kwon, and C.-C. Kuo, "Joint quality-temporal (Q-T) bit allocation for H.264/SVC," in *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, May 2009, pp. 2361–2364.
- [32] J. Liu, Y. Cho, Z. Guo, and J. Kuo, "Bit allocation for spatial scalability coding of H.264/SVC with dependent rate-distortion analysis," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 7, pp. 967–981, 2010.
- [33] N. Mohsenian, R. Rajagopalan, and C. A. Gonzales, "Single-pass constant- and variable-bit-rate MPEG-2 video compression," *IBM Journal of Research and Development*, vol. 43, no. 4, pp. 489–509, Jul. 1999.

- [34] M. Rezaei, M. Hannuksela, and M. Gabbouj, "Semi-fuzzy rate controller for variable bit rate video," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 5, pp. 633–645, May 2008.
- [35] A. Jagmohan and K. Ratakonda, "MPEG-4 one-pass VBR rate control for digital storage," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 5, pp. 447–452, 2003.
- [36] M. de-Frutos-López, O. del-Ama-Esteban, S. Sanz-Rodríguez, and F. Díaz-de-María, "A two-level sliding-window VBR controller for real-time hierarchical video coding," in *Image Processing, 2010. ICIP 2010. IEEE International Conference on*, Sep. 2010, pp. 4217–4220.
- [37] P. H. Westerink, R. Rajagopalan, and C. A. Gonzales, "Two-pass MPEG-2 variable-bit-rate encoding," *IBM Journal of Research and Development*, vol. 43, no. 4, pp. 471–488, 1999.
- [38] Y. Yu, J. Zhou, Y. Wang, and C. W. Chen, "A novel two-pass VBR coding algorithm for fixed-size storage application," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 11, no. 3, pp. 345–356, Mar. 2001.
- [39] W. Ding, "Joint encoder and channel rate control of VBR video over ATM networks," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 7, no. 2, pp. 266–278, 1997.
- [40] J. Bai, Q. Liao, X. Lin, and X. Zhuang, "Rate-distortion model based rate control for real-time VBR video coding and low-delay communications," *Signal Processing: Image Communication*, vol. 17, no. 2, pp. 187–199, 2002.
- [41] M. Dai, D. Loguinov, and H. Radha, "Rate-distortion analysis and quality control in scalable internet streaming," *Multimedia, IEEE Transactions on*, vol. 8, no. 6, pp. 1135–1146, 2006.
- [42] H. Lee, Y. Lee, D. Lee, J. Lee, and H. Shin, "Implementing rate allocation and control for real-time H.264/SVC encoding," in *Consumer Electronics (ICCE), 2010 Digest of Technical Papers International Conference on*, 2010, pp. 269–270.
- [43] S. Sanz-Rodríguez and F. Díaz-de-María, "RBF-based QP estimation model for VBR control in H.264/SVC," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 21, no. 9, pp. 1263–1277, Sep. 2011.
- [44] X. M. Zhang, A. Vetro, Y. Shi, and H. Sun, "Constant quality constrained rate allocation for FGS-coded video," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 2, pp. 121–130, Feb. 2003.
- [45] A. Unterweger and H. Thoma, "The influence of bit rate allocation to scalability layers on video quality in H.264 SVC," in *Picture Coding Symposium, 2007. PCS 2007*, Nov. 2007.
- [46] H. Mansour, V. Krishnamurthy, and P. Nasiopoulos, "Rate and distortion modeling of medium grain scalable video coding," in *Image Processing, 2008. ICIP 2008. IEEE International Conference on*, Oct. 2008, pp. 2564–2567.
- [47] M. Hannuksela, H. Zhu, H. Li, and M. Gabbouj, "Congestion-aware transmission rate control using medium grain scalability of scalable video coding," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*, Sep. 2010, pp. 2929–2932.
- [48] J. Vieron, M. Wien, and H. Schwarz, "JSVM 11 software," *24th Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG Meeting, Geneva, Doc. JVT-X203*, Jul. 2007.
- [49] M. Wien and H. Schwarz, "Testing conditions for SVC coding efficiency and JSVM performance evaluation," *16th Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG Meeting, Poznan, Doc. JVT-Q205*, Poznan, Poland, Jul. 2005.
- [50] [Online], "<http://www.tsc.uc3m.es/~sescalona/RbfVbrSvc/MultiBuffer/>."
- [51] G. Bjøntegaard, "Calculation of average PSNR differences between RD curves," *VCEG contribution, VCEG-M33, Austin*, Apr. 2001.
- [52] C. E. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.
- [53] E. Snelson, "Matlab code for sparse pseudo-input Gaussian processes (SPGP), [On-Line] <http://www.gatsby.ucl.ac.uk/~snelson/>, 2005. [Online]. Available: <http://www.gatsby.ucl.ac.uk/~snelson/>
- [54] E. Snelson and Z. Ghahramani, "Sparse Gaussian processes using pseudo-inputs," in *Advances in Neural Information Processing Systems 18*. MIT Press, 2006, pp. 1259–1266.



Sergio Sanz-Rodríguez (M'12) received the degree in Technical Telecommunication Engineering in 2001, the M.S. degree in Telecommunication Engineering in 2005, both from Universidad Politécnica de Madrid, Madrid, Spain, and the Ph.D degree in Multimedia and Communications in 2011 from Universidad Carlos III de Madrid, Madrid, Spain.

His primary research interests include rate control for video coding, scalable video coding, perceptual video coding and video signal processing.



Fernando Díaz-de-María (M'97) received the M.S. degree in Telecommunication Engineering in 1991 and Ph.D. degree in Telecommunication Engineering in 1996, both from Universidad Politécnica de Madrid, Madrid, Spain.

Since October 1996, Dr. Díaz-de-María has been an Associate Professor with the Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Madrid, Spain. His current research interests include image and video analysis and coding. He has led numerous projects and contracts in these fields. He is co-author of several papers in prestigious international journals and quite a few papers in revised national and international conferences.