PUBLIC

IST R&D. FP6-Priority 2.

SPECIFIC TARGETED RESEARCH PROJECT

Project Deliverable

SUIT Doc Number	SUIT_205
Project Number	IST-4-028042
Project Acronym+Title	SUIT- Scalable, Ultra-fast and Interoperable Interactive Television
Deliverable Nature	Report
Deliverable Number	D3.2
Contractual Delivery Date	31 January 2007
Actual Delivery Date	6 March 2007
Title of Deliverable	Design of Scalable Multiple Descriptions Video Coder
Contributing Workpackage	WP3
Project Starting Date; Duration	01/02/2006; 27 months
Dissemination Level	PU
Author(s)	Augustin Ion Gavrilescu, Kenneth Vermeirsch, Olivier Guye, Erwan Le Bras, Stewart Worrall, Antonio Navarro, Peter Schelkens, Fabio Verdicchio, Adrian Munteanu

Abstract

This document is an updated version of a previous document (D3.1), describing the block diagrams of three types of Multiple Description Coding (MDC) systems. The starting point in developing the MDC schemes is represented by a Scalable Video Coding (SVC) algorithm. Additionally the document is presenting the MDC combiner (central decoder in a classical MDC scheme) which will be either one component of the Gateway (in the scenarios that require a Gateway) or the one component of the terminal (in the case of scenarios that are not requiring WLAN transmission). Preliminary experimental results obtained with the MD-SVC are reported, demonstrating that the proposed approach enables both scalability and resilience against transmission errors.

Keyword list: Multiple Descriptions Coding, Joint Source Channel Coding, Quality Scalable Video Coder

PUBLIC

Design of scalable Multiple Descriptions Video Coder

SUIT_205

06-03-2007

Table of Contents

1 Introduction

Amongst several novelty points, SUIT attempts to converge two broadband mobile networks, IEEE 16e and DVB-T/H. SUIT will deliver one layered description over each of those last mile networks. Thus, SUIT intends to push video scalability into broadcasting and network networks in a fruitful way thereby using two video descriptions. As a final objective, SUIT intends to demonstrate an end-to-end communication system from the playout to the terminal where the terminal can be an HDTV set or a small size terminal.

Deliverable D3.2 is an updated version of the first deliverable (D3.1) of workpackage 3 (WP3). The main objectives of WP3 are:

1. To design scalable multiple-description video coding approaches adaptable to the dynamic network characteristics and the multitude of types of user terminals.
2. To design adaptive joint source and channel coding techniques for optimal network resource allocation so as to take advantage of the source scalability and channel conditions;
3. To optimize the overall rate-distortion performance for a given user preferences (in terms of resolution and frame-rate) and network conditions.

D3.1 and D.3.2 depict the steps to be followed in order to achieve the first task mentioned above. These deliverables follow two previous deliverables, D1.1 and D1.3 and will influence maybe other deliverables, namely some WP5 (Components for the testbed) deliverables.

SUIT will set up four base stations in two cells, where they will be co-sited in pairs. So, a cell will have a DVB-T/H and a WiMAX basestations. This network scenario allow us to test different type of services and functionalities.

In D1.1, the following service scenarios associated to service bit rates and depending on two different DVB-T/H multiplexers. In the tables below, we are only considering one base station in operation in each cell. One cell has a DVB-T/H base station and the other cell has a WiMAX base station.

DVB-T		WiMAX	
Service	Bit Rate (Mbps)	Service	Bit Rate (Mbps)
1 D SVC Real Time Broadcasting	6.25	2 D Real Time Broadcasting	6.25
1 D SVC Broadcasting	6.25	2 D Broadcasting (on QoS demand)	0;6.25
1 D Hyperlinked Video	0.5	2 D Hyperlinked Video	0.5
Internet	0-?	Internet	0-?
		Streaming	0.5-4.25
Total	13	Total	14-17.75

D= Description;

SVC= HD: 1280x704p-25 Hz (4.25 Mbps) ; SD: 640x352x25 (1.5 Mbps) ; CIF: 320x176x25 (0.5 Mbps)

Table 1-Network/services scenarios (13 Mbps IRT Multiplexer)

DVB-T		WiMAX	
Service	Bit Rate (Mbps)	Service	Bit Rate (Mbps)
1 D SVC Real Time Broadcasting	10	2 D Real Time Broadcasting	10
1 D SVC Broadcasting	10	2 D Broadcasting (on QoS demand)	10
1 D Hyperlinked Video	0.5	2 D Hyperlinked Video	0.5
Internet	0-?	Internet	0-?
		Streaming	0.5-8
Total	20.5	Total	21-28.5

D= Description;

SVC= HD: 1280x704p-50 Hz (8 Mbps) ; SD: 640x352x25 (1.5 Mbps) ; CIF: 320x176x25 (0.5 Mbps)

Table 2- Network/services scenarios (20.5 Mbps IT Multiplexer)

The service described in the first row in the tables is a real time broadcasting composed by two descriptions, each delivered to a particular network. In the case of transmission over error prone channels a terminal receiving both descriptions will be able to display a better quality video. In the second row, a pre-recorded material will be broadcasted over DVB and its second description can be unicasted over WiMAX to a particular terminal requesting better quality video. In other words, this service is multicasted to all terminals requesting a better quality video. This situation can occur mainly in the cities where WiMAX can cover DVB dead zones or when the mobile is moving at high speed. This main pre-recorded material has a hyperlink to a short video. For instance, the viewer is watching a football match and wants to watch a short spot (<10 min) of the best goal scored by one player. The hyperlinked video will then be displayed on a corner on the top of the main video. The hyperlinked video requires a low delay communication and is unicasted (downloaded) to a particular terminal. Therefore, the intelligent playout will upload the hyperlinked video descriptions through both networks, selecting them intelligently in order to ensure low latency. If needed to ensure low latency, the playout should reduce the bit rate allocated to each broadcasted service described in the two top rows in the tables.

In the forth row, the SUIT playout will serve (unicast), again intelligently, a terminal with internet contents. The playout will then select the most appropriate network depending on available empty slots (packets) in each network or by reducing the bit rate associated to the broadcasted material with negligible quality loss.

Finally, any terminal may request a video streaming service from the playout server or even from outside the playout. This service will be unicasted only via WiMAX. Again, the palyout may need to reduce the bit rate associated to the broadcasted material delivered over WiMAX.

Despite the service scenarios in the tables above describe 3 spatial layers from quasi-CIF to quasi-HD, for the first year SUIT will restrict the services to the above tables second line since we want to demonstrate the advantages of having layered MDC as soon as possible. Besides, in the first year we will only demonstrate two layers, at CIF format, the base layer plus a FGS layer. This testbed will be upgraded progressively by adding spatial scalability in order to serve different terminal screens.

1.1 Scope

This document is developed in the framework of SUIT FP6 project. The scope of the document is to describe the design of Multiple-Description Scalable Video Coding (MD-SVC) systems simultaneously enabling (1) temporal, quality, and spatial scalability, and (2) resilience against transmission errors by relying on MDC principles. It has to be mentioned that the proposed system represents the first milestone in building a hybrid error-resilient joint source and channel video coder, which will cascade the proposed MD-SVC system with Forward Error Correction.

1.2 Objective

The main objective of this document is to detail the design of MD-SVC enabling the three forms of scalability. i.e. quality, temporal and spatial scalability.

The main building block of the proposed MDC system consists in the scalable video coding (SVC) system. The SVC enables the media providers to generate, in a single compression step, a unique bitstream from which appropriate subsets, producing different visual qualities and frame-rates can be extracted to meet the preferences and the bit-rate requirements of a broad range of clients. Additionally, The MD module will provide error resilience for data transmission over error prone channels. Since this coder is based on the Scalable Video Coder (SVC) and additionally employs multiple descriptions (MD) techniques in order to provide multi-path transmission will we hereinafter refer to as Multiple Description Scalable Video Coder (MD-SVC).

The design of error resilient methods tailored specifically to the SUIT complex framework (involving transmission over WiMax and DVB and additionally last mile connection over WLAN), a number of three MD approaches have been proposed. Each of the approaches has its specific characteristics that can be quantified in algorithm complexity, flexibility and ability to provide fast encoding/decoding capabilities.

The three proposed MDC approaches are:

1. Unbalanced MDC
2. MDC based on redundant slices
3. MDC based on embedded multiple descriptions scalar quantizers (EMDSQ)

By providing this range of choices the proposed system can adapt to different transmission scenarios characterized by specific error rates and error patterns. Additionally, for applications implying tight time transmission constraints one can chose the most suitable solution among the present algorithms.

Finally this document presents the Gateway and the Terminal module, highlighting the similarities and differences between the two. The purpose is to provide a flexible solution that can be easily adapted for both scenarios (i.e. whether the terminal is connected to the Gateway, or whether the Terminal is directly connected to the WIMax – DVB network).

2 Scalable Video Coding

2.1 Introduction

For economical transmission over networks, It is desirable to send the compressed data progressively, and to refine the image quality at the decoder side as more information is received. Additionally, the user may require decoded data at different resolutions and the coding techniques employed have to support transmission of efficiently compressed bit-streams capable of providing appropriate resolutions. In this context, embedded coding enables the progressive transmission of the compressed data by starting with an economical initial transmission of a low quality image version, followed by gradual transmission of the refinement details, without adding any bit-rate overhead compared to that needed for the lossless reconstruction. From this perspective, scalability of the source representation, coupled with robustness to transmission errors, are two important features for facilitating adaptation to the inherently variable network conditions, user's needs, and terminal characteristics. The previous overview of the Scalable Extension of H.264/AVC is now including the spatial scalability tools currently proposed by the JVT as the extension of the combination of different scalabilities in the future standard.

2.2 Overview of the Scalable Extension of H.264/ AVC

While the H.264/AVC standard provides state-of-the-art compression performance for single layer video coding, it does not support any form of scalability (SVC). In parallel, the JVT group is also working on a scalable version of H.264 which would have close similarities with H.264/AVC video coding standard. The current working draft combines the coding primitives of H.264/AVC with an open-loop coding structure allowing to merge together spatial, temporal and quality scalabilities .

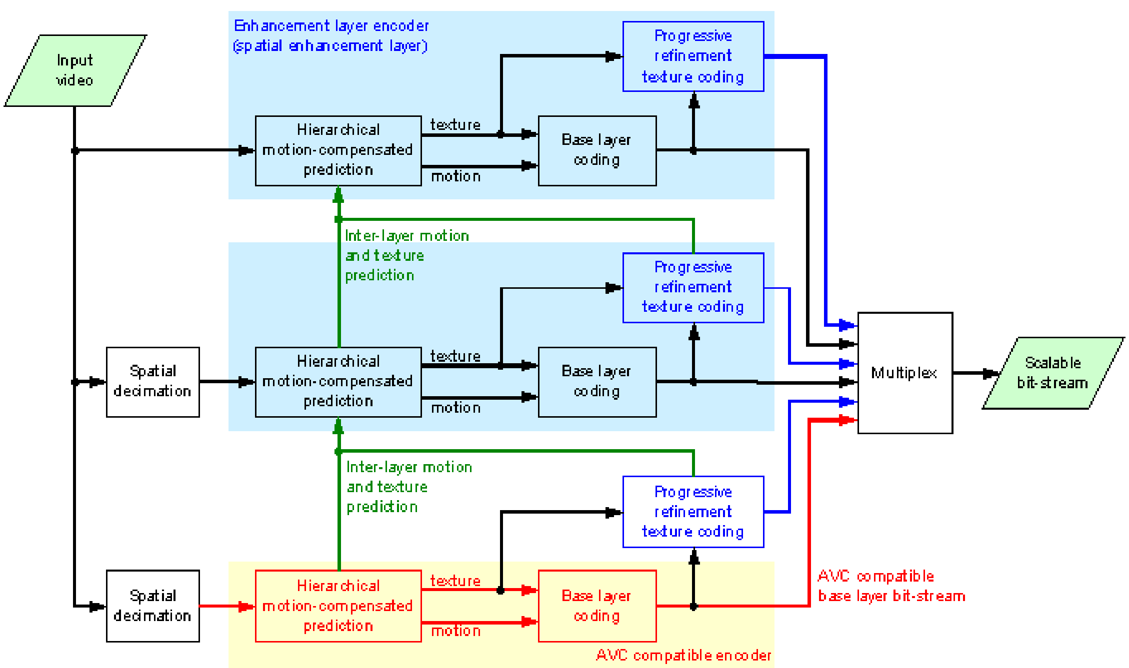


Figure 1: SVC Encoding Architecture

The key features of the scalable extension of H.264 / MPEG-4 AVC are:

- hierarchical prediction structure ;

- layered coding scheme with switchable inter-layer prediction mechanisms ;
- base layer compatibility with H.264 / MPEG-4 AVC ;
- fine granular quality scalability using progressive refinement slices ;
- usage and extension of the NAL unit concept of H.264 / MPEG-4 AVC.

The basic coding scheme for achieving a wide range of spatio-temporal and quality scalability can be classified as layered video codec. The coding structure depends on the scalability space that is required by the application. Just above, a block diagram for a typical scenario with 3 spatial layers is depicted.

Spatial scalability is modelled using a Laplacian pyramid to compute the different spatial layers applying successive decimations from the highest resolution down to the lower one ([4]).

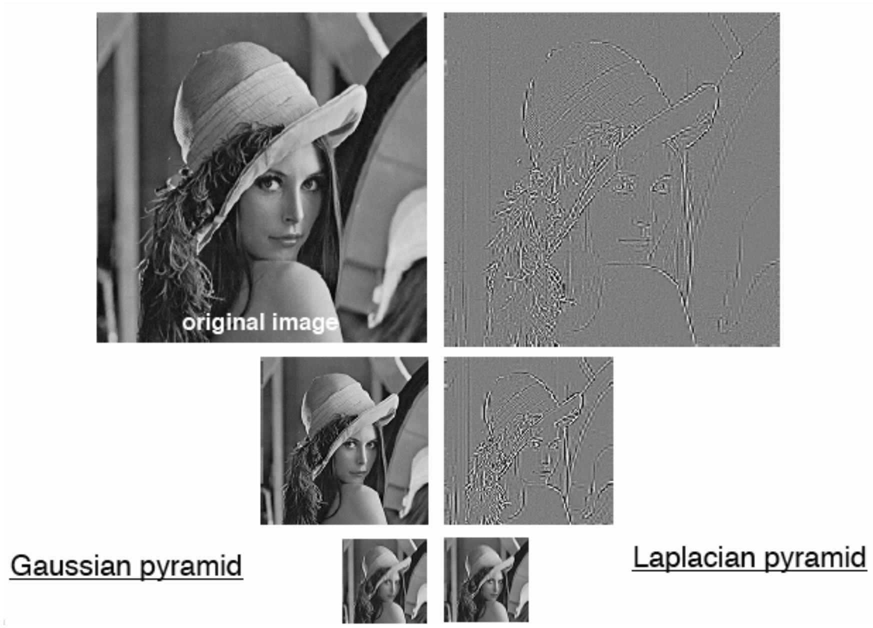


Figure 2: Spatial Decimation

The following figures show the successive steps that enable encoding and decoding of the different scalable layers of an SVC encoded stream of pictures.

At each refinement stage of the encoding process:

- the difference picture computed at the coarser level is re-quantized in the transform domain with a finer resolution to provide a new refinement layer;
- the information present in this refinement layer is withdrawn from the difference picture that is sent to the next refinement stage.

As a first interpretation, the pictures for different layers are coded independently with layer-specific motion information. Although, the following techniques turned out to provide gains and were included into the scalable video codec as **Inter-layer Prediction** techniques:

- prediction of intra-macroblocks using up-sampled base layer intra blocks ;
- prediction of motion information using up-sampled base layer motion data ;
- prediction of residual information using up-sampled base layer residual blocks.

Inter-Layer Intra Texture Prediction may be applied according three different manners, using:

- unrestricted prediction, where any block of any lower level is chosen (still not in the draft standard);
- constrained prediction, where prediction can applied only from intra-coded blocks of base layer;

- constrained for single loop decoding, where prediction can applied only from intra-coded blocks of base layer in any picture of the temporal decomposition as well as the key picture.

Up-sampling techniques are then applied between two successive spatial layers, by using:
- a half-pel interpolation for a ratio-of-2;
- a quarter-pel interpolation for a ratio between 1 and 2.

Inter-Layer Motion Prediction can be applied from a previous layer by using:
- same image reference indices;
- same macroblock partition;
- same motion vectors which are up-sampled;
- a possible quarter-pel refinement.

Inter-Layer Residual Prediction may enhance motion information in conjunction with Inter-Layer Motion Prediction.

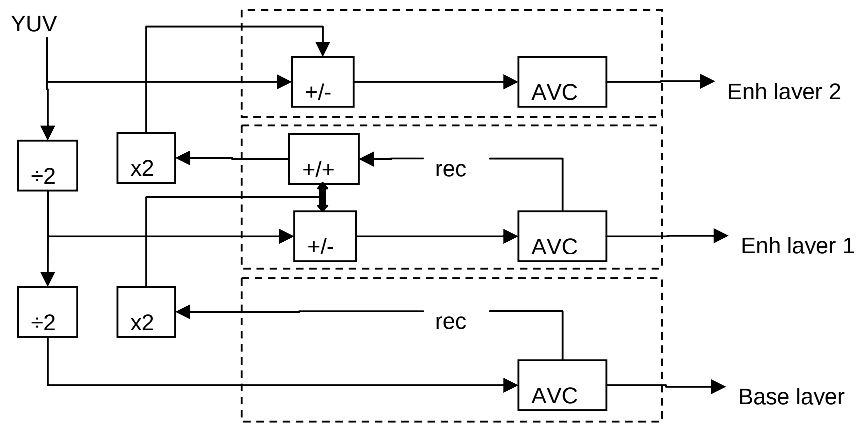


Figure 3: Spatial Scalable Encoding

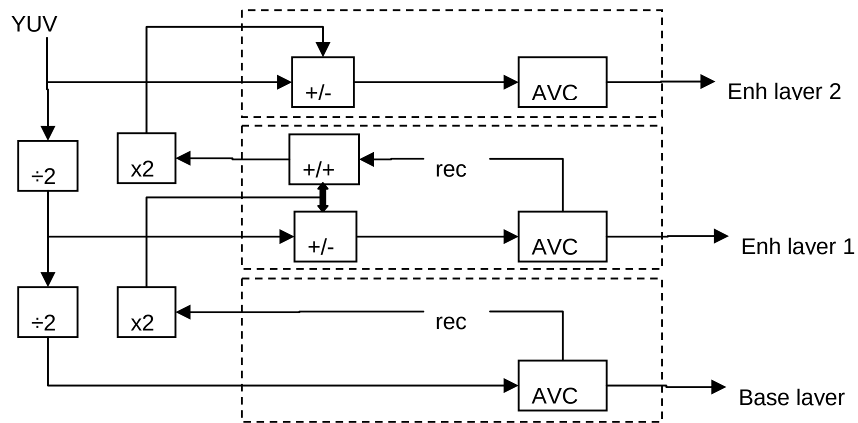


Figure 4: Spatial Scalable Decoding

At reconstruction time, all the available refinement layers are successively de-quantized to sum up all the enhancement information.

Temporal scalability is implemented through a hierarchical prediction structure is realised according to H.264 B pictures (**BH**).

In the SVC draft standard, two **SNR scalability** schemes have been proposed: **CGS** and **FGS** scalabilities. The FGS scheme offers a convenient framework to introduce advanced multiple description schemes, through adaptation of the quantization step size. With this aim in mind, the quality scalable encoding process of SVC will now be examined in greater depth.

The FGS scheme, begins with a base layer that is fully compliant with the AVC syntax and usually provides a coarse quality description of the incoming picture in terms of visual quality. This base layer is progressively improved with each successive refinement layers until the finest quality layer is reached. By decoding all of the layers, from the coarsest to the finest, a full quality picture can be retrieved. The finest quality layers can be discarded without incurring significant degradation in perceptual quality. This feature can be used to adapt the visual content to the available bandwidth when transmitted over a narrow-band or time varying channel.

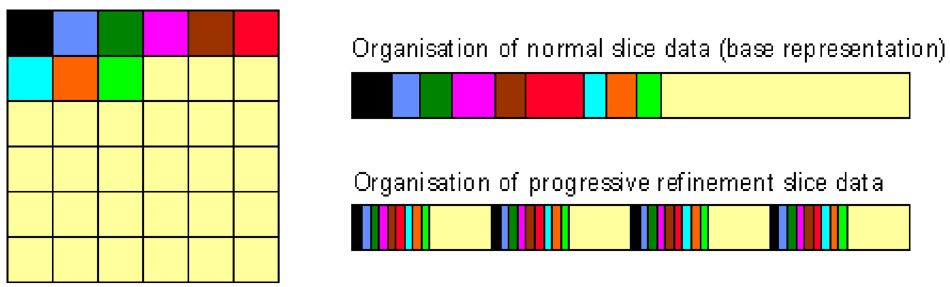


Figure 5: FGS data organization

The following figures show the successive steps that enable encoding and decoding of the different quality layers of an FGS encoded stream of pictures.

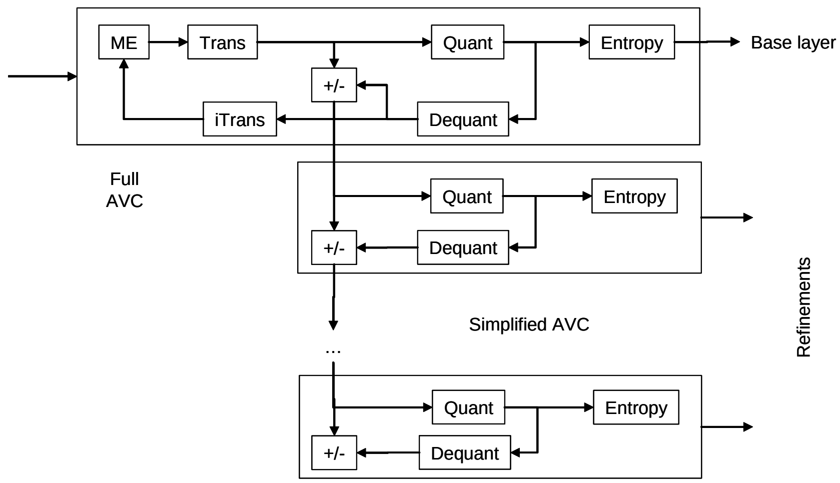


Figure 6: Quality scalable SVC encoder

At the base layer of the encoder:

- as in a conventional coder, motion estimation is computed to reduce temporal redundancy and a direct domain transform is applied to reduce spatial redundancy;
- transformed coefficients are coarsely quantized and the difference between the original picture and the reconstructed one is computed and used to build progressive refinement layers.

At each refinement stage of the encoding process:

- the difference picture computed at the coarser level is re-quantized in the transform domain with a finer resolution to provide a new refinement layer;
- the information present in this refinement layer is withdrawn from the difference picture that is sent to the next refinement stage.

At the output of each encoding stage, entropy coding is used to remove syntactical redundancy.

At the input of each decoding stage, entropy decoding is applied to retrieve the encoded information. At reconstruction time:

- all the available refinement layers are successively de-quantized to sum up all the enhancement information;
- the base layer is reconstructed by applying the inverse transform and inverse motion compensation to recover the pictures belonging to the original stream.

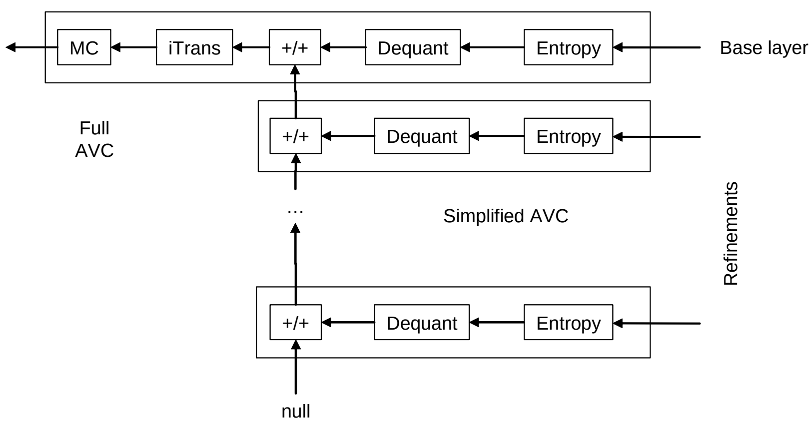


Figure 7: Quality scalable SVC decoder

The loss of fine quality refinement layers has a less perceptible impact on visual restitution than the loss of coarse quality refinement layers. The loss of the base layer prevents reconstruction of those pictures included in the lost segment. To avoid this drawback, multiple description schemes could be beneficial, especially schemes that are based on embedded scalar quantization techniques.

It allows access to and truncatation of the bit-stream with increased flexibility, as shown underneath, by combining the different kinds of scalability present in the scalable extension of H.264/AVC. Further details on Multiple Description Coding improvements may be found in the following section.

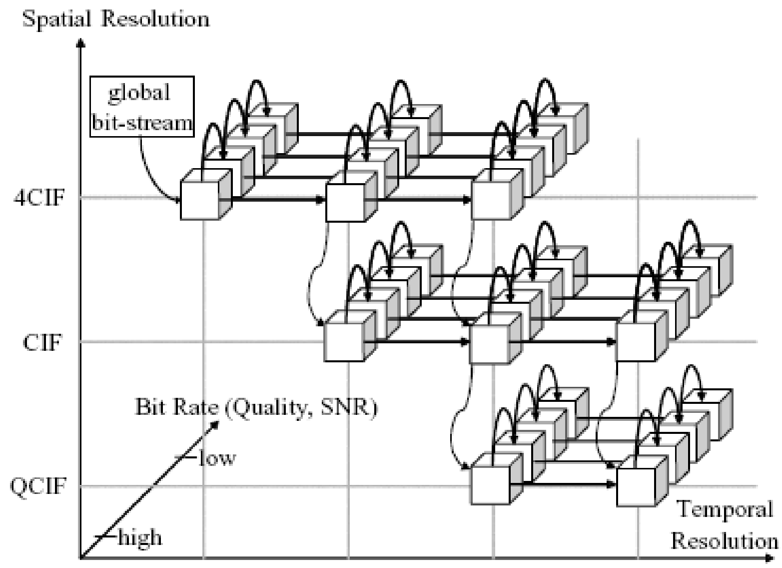


Figure 8: Visiting a Scalable Bitstream

3 Multiple Description Coding

The ever increasing demand for efficient transmission of multimedia content over best effort networks and error-prone channels (e.g. packet networks, low-power wireless links) has fuelled intensive research in the area of robust communication techniques. In this context, MDC is a competitive solution to overcome the channel impairments using diversity [5][6][7][8]. This coding paradigm relies on generating more than one description of the source such that:

- each description independently describes the source with a certain fidelity,
- when more than one description is available at the decoder, one can combine them to enhance the decoded quality.

This has the inherent advantage that the quality of the reconstructed data gracefully degrades with increasing probability of failure on the transmission channel. The produced descriptions are independently decodable, allowing the decoder to reconstruct the source with a certain fidelity, from a subset of initial number of descriptions. Under the scope of the SUIT project, two descriptions will be provided by MDC, for transmission over two different networks i.e. DVB and WiMAX, respectively.

It is important to note that either of the DVB and WiMAX networks with the corresponding description should be able to function independently. However, as described in Section 1.1, scalability is a desired feature in the context of efficient data transmission. Consequently, the rate adaptation of each description is a desirable feature and the design of an scalable MD-SVC system is of paramount importance. The MD-SVC will be able to deliver video content over best-effort error-prone packet networks, and, due to its scalable erasure-resilient compression capabilities, it is able to

- meet the users' requirements in terms of quality and resolution,
- dynamically adapt the rate to the available channel capacity, and
- provide robustness to data losses as retransmission is often impractical.

It is important to note that in a video streaming scenario, selective retransmission of lost packets is often not desired because of the timing requirements and low delay that are expected from the system. This is especially true for broadcast and multicasting where the streaming server would be burdened by a potentially very large amount of retransmission requests. The traditional approach taken in such a scenario is to use Forward Error Correction (FEC), meaning some kind of channel coding is added at the sender side. This allows receivers to autonomously correct bit errors or packet erasures caused by the lower layers of the network, without the need for retransmission of information. Some examples of FEC codes are block codes, convolutional codes, and LDPC codes.

The basic protection scheme incorporating FEC codes are designed from a worst case point of view, and the amount of added redundancy is fixed even under ideal network conditions. In this case, FEC codes suffer from what is known as the cliff effect: by design, they show a constant excellent performance up to a well-defined number of erasures; once the number of actual erasures exceeds this figure, performance drops very sharply to a very low level. Multiple Description Coding (MDC) provides a much more gradual performance vs. erasures curve and wastes less bandwidth resources under good network conditions.

3.1 Unbalanced MDC

A Balanced Multiple Description (BMD) system, in which all descriptions are of equal rate and equally important, suffers from some inherent drawbacks. First, in predictive video coders, if a prediction signal is present in only one description and this description was lost during transmission, the prediction signal will not be available, resulting in decoder drift. Secondly, since all descriptions are of equal rate, bandwidth utilization is often not optimal: the rate must be kept below the minimum available bandwidth among all channels. This is especially apparent when networks with considerably different throughputs are being used.

These issues are absent in an Unbalanced Multiple Description (UMD) approach: in a two-channel scenario, an UMD coder generates a coded video stream at full quality, along with a second version at reduced quality. These versions make up the descriptions and are each transmitted over a different network. The low quality version constitutes a 'base' version and is essentially redundant: it is only used to conceal errors in the high quality version when transmission loss occurs in the high quality description. The descriptions are unbalanced, since the low quality version will typically be of much lower rate than the full quality description. UMD enables improved utilization of available bandwidth in the underlying networks. Furthermore, since both descriptions can be independently coded, it is easy to design a UMD system that is inherently drift-free.

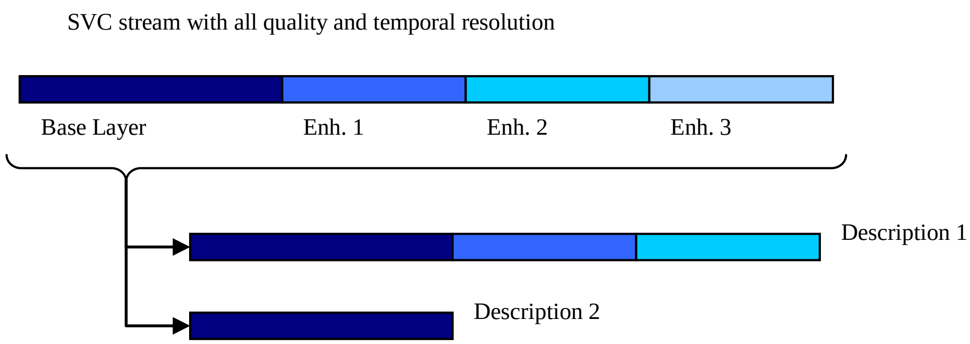


Figure 9 Two descriptions Unbalanced MDC generated from a full quality SVC stream

In an SVC context, the low fidelity version could differ from the high fidelity version along all three axes of scalability: temporal, spatial, and/or quality (SNR). Obtaining a lower SNR version can be accomplished by requantization or by simply dropping (part of the) residual texture. However, to maintain acceptable video quality when either description is lost, the use of SVC's inter-layer prediction across both descriptions must be avoided. In addition, since the low fidelity version is essentially completely redundant, the degree of redundancy across both descriptions will become very large when both networks offer comparable bandwidth. In contrast, in the Balanced MDC case both descriptions mutually refine each other, and the amount of redundancy does not depend on the difference in available bandwidth across both networks.

3.2 MDC based on redundant slices

Using redundant slices, multiple descriptions of a video sequence can be generated at a high level in the video coding process, namely, at the NAL unit level. The fundamental idea is that (depending on the amount of redundancy that is desired) some or all coded slices are sent in both descriptions. At the receiving side, it is sufficient that only one copy of every slice is received in order to fully reconstruct the video sequence.

In practice, two approaches can be taken to implement MDC based on redundancy:

1. A separate module generates both descriptions, based on the single description output of the SVC encoder. This could happen by a straightforward copying of NAL units. Therefore, the encoder does not need to be aware of the existence of the MDC system. At the receiving side, a separate module recombines both descriptions, sending the resulting single description coded video stream to the decoder, which is unaware of the MDC process.
2. We use the provision in H.264/AVC and SVC to signal, in the slice header, whether the slice is redundant or not. In this manner, both descriptions can be merged at the receiver side and sent to the decoder in a single, compliant bitstream. This simplifies the task of the MDC combiner, but requires a decoder that supports the redundant slice syntax. It also means more traffic will be sent over the last-mile network in the gateway scenario. At the

playout side, a separate MDC module could still be used; however, it would need to parse and modify slice headers. Alternatively, an encoder with built-in support for generating redundant slices could be used.

Within the context of SUIT two descriptions are used; therefore, the following text will mainly focus on this case, although most techniques can also be used with three or more descriptions.

It should be stressed that, since the unit of granularity in these techniques is a slice, the proposed methods can be applied to any layer in a scalable bitstream. Which layers will be protected by redundant slices is purely a matter of policy.

The following sections will highlight several schemes based on redundant slices, illustrating both of the above methods. The overview starts with the most straightforward solution (1:1 duplication of slices), and then moves to the more complex modes involving redundant slices of different SNR quality. The following subsection first presents a more in-depth discussion of the H.264/AVC and SVC provision of signalling redundant slices; this pertains to the second of the two approaches given above.

Syntax and semantics of H.264/AVC and SVC redundant slice signaling

H.264/AVC and its scalable extension introduce the *Access Unit* concept. An Access Unit consists of a *Primary Coded Picture*, followed by zero or more *redundant pictures* containing redundant slices pertaining to that picture. It is not required by the specification that all slices of a redundant picture are present. While the Primary Coded Picture conveys the coded slices that are needed for a baseline reconstruction of the video sequence, redundant slices contain an additional coded representation for certain macroblocks. This could be a lower-quality representation compared to the coding used in the Primary Coded Picture. More specifically, the encoder has the freedom to set coding modes and quantization for the redundant slices that are completely different from the ones used in the Primary Coded Picture. Even the slice and slice group structure of a redundant picture can be totally different from its Primary Coded Picture's. As a result, there is much freedom in what data is conveyed in the redundant slices.

To signal the presence of a coded slice, **redundant_pic_cnt_present_flag** shall be equal to one in the Picture Parameter Set referenced by the slice. This allows the **redundant_pic_cnt** syntax element to be included in the slice header. It shall be equal to zero for a non-redundant (Primary Coded Picture) slice, and between one and 127, inclusive, for a redundant slice.

There is no normative requirement about how a compliant decoder should handle redundant slices. However, the standard advises that, if available, a redundant slice be used when it covers an area that cannot be reconstructed normatively due to the loss of a slice of the Primary Coded Picture. When multiple redundant slices are eligible, the one having the lowest value of **redundant_pic_cnt** should be used.

3.2.1 Duplication of slices among both descriptions

The easiest way to realize redundant slices is to simply take the NAL units produced by the SVC encoder and send them, in unmodified form, over both networks. At the terminal/gateway, after reception and recovery of the NAL units from the transport layer, two NAL unit streams result. These then need to be synchronized to each other and duplicate NAL units need to be removed. The resulting NAL unit stream is again SVC compliant and can be decoded (in the case of a terminal) or sent over the last-mile network (in the case of a gateway).

This technique makes no use of the H.264/AVC syntax for signalling redundant slices; therefore, a great benefit of this technique is that encoder and decoder do not need to be aware of (or be able to handle) AVC/SVC redundant slices, since the entire process takes place in between.

By choosing which pictures or slices to duplicate, it is possible to control the amount of redundancy. For example, we could choose not to duplicate B-predicted slices. This is especially true in a scalable bitstream, where we could choose to duplicate only the slices of a certain set of layers, or we could choose not to duplicate Progressive Refinement slices. This can be seen as a form of unequal error protection (UEP), where the most ‘important’ data (e.g. parameter sets; data used as a reference in future coded pictures; …) is given better protection than the less important.

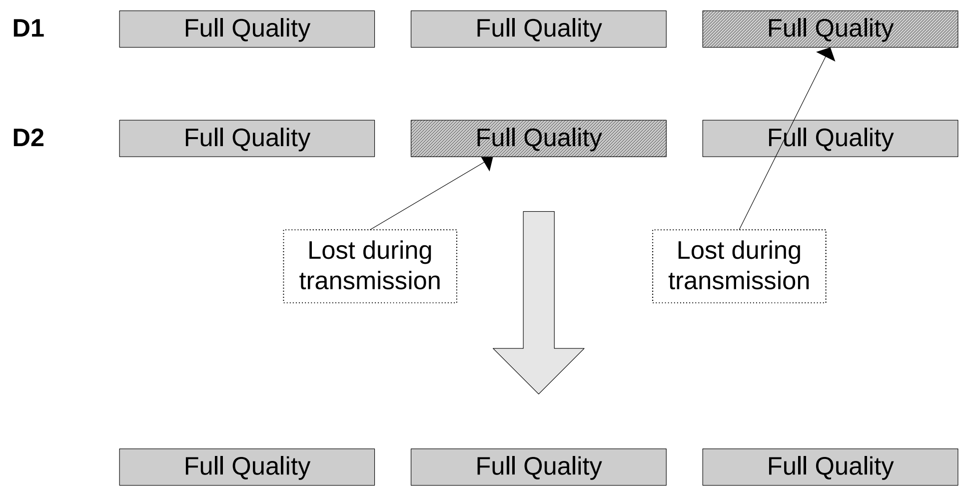


Figure 10 MDC combining of two descriptions of equal quality.

3.2.2 Duplication of slices with H.264/AVC redundant slice signalling

A variant of the previous technique follows the same principles, but now the nature of the slices (original or redundant) is signalled in the bitstream, using the H.264/AVC redundant slice syntax. A primary slice and its redundant copy contain identical data, but have a slightly different slice header. More precisely, as explained earlier, the **redundant_pic_cnt** syntax element shall be zero in the primary slice and nonzero in redundant slices.

From the perspective of the terminal/gateway, the use of this scheme implies that we no longer need to eliminate duplicate NAL units ‘by hand’. However, we do need a decoder that supports redundant slices. Also, in the gateway scenario, all NAL units from both descriptions need to be sent over the last-mile network, thus greatly increasing its bandwidth requirements.

At the playout side, the use of this technique can mean one of two things:

1. An encoder is needed that is capable of producing the redundant slices itself, and outputs two NAL unit streams corresponding to both descriptions.
2. The encoder has no support for redundant slices; as in the previous technique, a separate module generates the descriptions from the single NAL unit stream produced by the encoder. However, now this module not only duplicates the NAL units, but also modifies the slice headers, making sure the result complies to H.264/SVC. These modifications require the parsing of slice headers, picture parameter sets and sequence parameter sets, so it must be stressed that this is programmatically much more complex than the previous approach.

3.2.3 Redundant slices of lower texture quality

When using redundant pictures, there is no hard requirement that a redundant slice should contain data identical to its corresponding primary slice. In other words, we could imagine a situation where the redundant slices are of a lower quality than their original. This can be very useful in case one description is sent over a network with considerably lower bandwidth compared to the other description.

Here as well, we can choose whether to use the syntax elements in the slice header to signal the redundant slices; or to split and combine the descriptions without the knowledge of encoder and decoder.

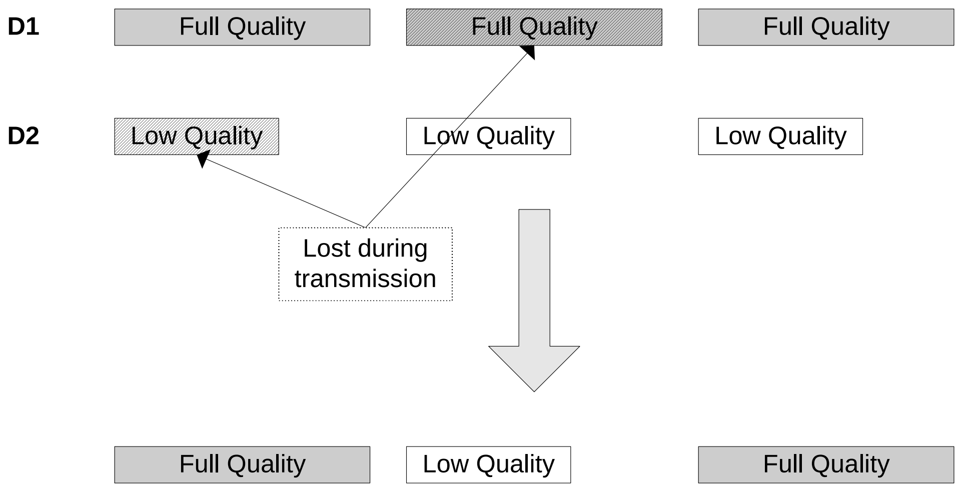


Figure 11 MDC combining of two descriptions of unequal quality.

When image data from a slice is used as a prediction signal for future coded slices, but the slice is replaced by a version with lower texture quality, drift errors may result:

1. Drift errors in other slices of the same picture, due to H.264/AVC intra prediction. This also affects the H.264/AVC-compatible base layer of scalable (SVC) bitstreams. This drift will not propagate into future pictures;
2. Drift errors in future pictures (in coding order), due to motion compensation when the current picture is used as a reference picture. Again, this affects H.264/AVC as well as SVC streams. This drift can propagate until the affected image areas are refreshed by intra coded slices;
3. Drift errors in higher layers of a scalable bitstream, due to inter-layer prediction when the current layer is used as a reference for motion or texture information in higher layers.

In other words, there is a trade-off between the amount of redundancy and the quality of not only the current picture, but also of other layers and (predicted) pictures.

3.2.4 Two slice groups and different texture qualities

While the previous technique leads to unbalanced multiple descriptions, it is also possible to think of a balanced scheme with redundant slices of lower quality. Such a scheme could be as follows:

1. The picture is partitioned in two slice groups of equal total size;
2. Two versions of the picture are coded: a low-quality version, coded as redundant slices; and a full-quality version, coded as Primary Coded Picture slices.
3. Low-quality slices of the first slice group and full-quality slices of the second slice group are sent in Description 1. In Description 2, the inverse is true: it contains the full-quality version of the first slice group and the low-quality version of the second slice group.

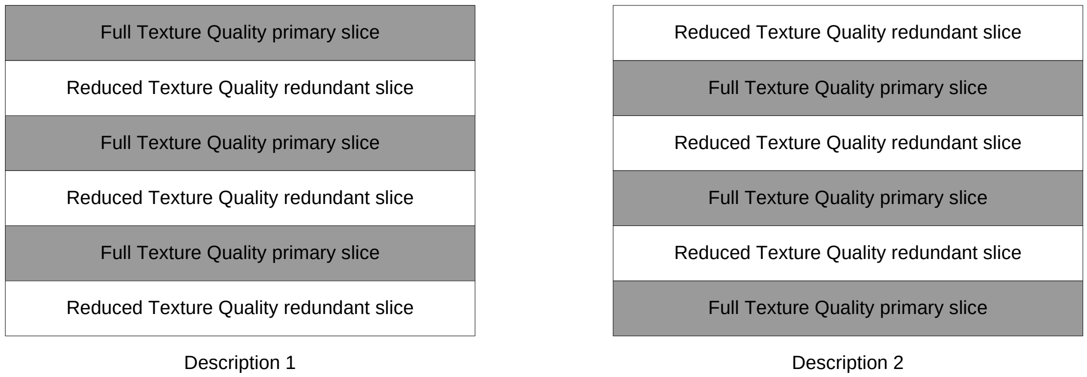


Figure 12 Example partition of a picture in slices of unequal quality for transmission over two description.

A redundant picture-aware decoder will decode the primary slices whenever available. This means that, if one of both descriptions is lost, one half of the reconstructed picture will consist of low-quality redundant slices, and the other half will be of full quality. If both descriptions were correctly received, all redundant slices are ignored and the entire picture is of full quality.

3.2.5 SUIT and redundant slice coding

The list of redundant slices techniques presented here is in no way exhaustive. Because of the freedom that exists in exploiting redundant slices, many more schemes are imaginable; for example, the choice of the location of redundant slices within a picture could be driven by Region Of Interest (ROI). The previous sections only illustrate the most interesting solutions in the context of two descriptions and the SUIT project. It is important to keep in mind that the more advanced techniques are only viable when encoder and decoder support for redundant pictures is present. In contrast, the simpler schemes presented in 3.2.1 to 3.2.3 are sufficiently basic to be realized by a separate module, without the standards-based support of encoder or decoder. (The technique in 3.2.3, however, would require a dual-encoder bank in order to generate the exact same bitstream in two different texture qualities.)

3.3 MDC based on EMDSQ

3.3.1 MDC based on scalar quantization

The first practical MD coding system based on scalar quantization is proposed by Vaishampayan in [5] where the concept of MD scalar quantizers (MDSQ) was introduced. The MDSQ can be seen conceptually as a pair of independent scalar quantizers that give as an output two descriptions of the same input real source sample. From a constructive point of view the MDSQ problem can be divided in two as follows.

First, the central quantizer has to be determined leading to an optimal partitioning of the contained cells. Secondly, given the central quantizer partitioning, the problem of an index assignment scheme, which efficiently allocates the indices of the two individual side quantizers has to be tackled. In brief, a MDSQ consists of two main components: (a) a *scalar quantizer* (b) an *index assignment* (IA). Figure 13 illustrates the structure of an MDSQ with the two main components as described above.

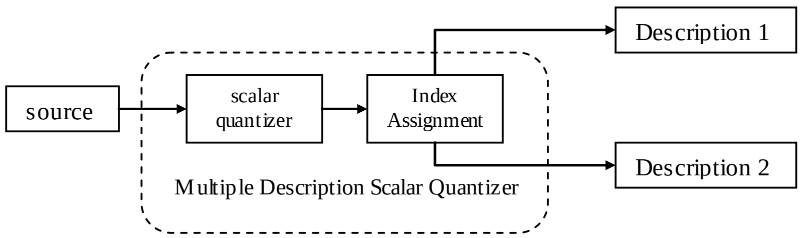


Figure 13 The MD coding system based on multiple descriptions scalar quantizers

Consider a two-description MD system based on MDSQ and characterized by an N-level central quantizer. In this case the IA can be defined as a mapping function $\delta : N \longrightarrow J^2$. It is noticeable that in order to be able to recover the original index the mapping function has to be injective.

Direct IA.

$n \longrightarrow (n_1, n_2)$

$S = \left\lfloor \frac{n}{NrDiag} \right\rfloor$;

$offset = n \% NrDiag$;

If ($offset$ is even) then $n \longrightarrow (S, S + \frac{offset}{2})$;

If ($offset$ is odd) then $n \longrightarrow (S + \left\lceil \frac{offset}{2} \right\rceil, S)$;

Invers IA

1. Central	**Side 1**	**3. Side 2**

$(n_1, n_2) \longrightarrow \hat{n}$
$S = \min(n_1, n_2)$;
If ($n_1 > n_2$) $offset = 2 \cdot S - 1$;
If ($n_1 \geq n_2$) $offset = 2 \cdot S$;
$\hat{n} = S \cdot NrDiag + offset$
	$(n_1, -) \longrightarrow \hat{n}$
	For $0 \leq i \leq \left\lfloor \frac{NrDiag}{2} \right\rfloor$
	$S_i = NrDiag \cdot n_1 + 2i$;
	For $1 \leq i \leq \left\lfloor \frac{NrDiag}{2} \right\rfloor$
	$val = NrDiag - i(NrDiag - 2) - 1$;
	If (val>0) then $S_{i + \left\lfloor NrDiag/2 \right\rfloor} = val$;
	else $S_{i + \left\lfloor NrDiag/2 \right\rfloor} = 0$;
	If any $S_i = 0$ then $\hat{n} = 0$
	else $\hat{n} = \frac{1}{NrDiag} \sum_{i=0}^{NrDiag} S_i$
		$(-, n_2) \longrightarrow \hat{n}$
		$S_0 = NrDiag \cdot s_2$
		For $1 \leq i \leq \left\lfloor \frac{NrDiag}{2} \right\rfloor$
		$S_i = NrDiag \cdot n_2 + 2i - 1$;
		For $1 \leq i \leq \left\lfloor \frac{NrDiag}{2} \right\rfloor$
		$val = NrDiag - i(NrDiag - 2)$;
		If (val>0) then $S_{i + \left\lfloor NrDiag/2 \right\rfloor} = val$;
		else $S_{i + \left\lfloor NrDiag/2 \right\rfloor} = 0$;
		If any $S_i = 0$ then $\hat{n} = 0$
		else $\hat{n} = \frac{1}{NrDiag} \sum_{i=0}^{NrDiag} S_i$

Consider additionally that the central scalar quantizer maps the input source to a finite number of points. According to this, the map δ can be thought of as a matrix of size $J \times J$, in which only N locations are occupied (the occupied locations correspond to the central quantizer indices).

In this way, to each central quantizer indices mapped in the IA matrix, we can allocate a pair representing the column and row indices. The resulting pair represents the side quantizer indices, constituting the two refinable descriptions of the input source.

The first IA design for multiple description scalar quantizers is found in [5] where two families of *diagonal index assignment* matrices are proposed. Both families of diagonal IA are build under the constraint that the difference between two indices sharing the same description has to be minimized (minimize the *spread*). The asymptotic soundness of this criterion is proved under the assumption of equal and high description rates and for the case of squared error distortion measure. The proposed IAs are called *diagonal* based on the propriety that the indices are distributed over the main diagonal and the neighbour diagonals. This will result in having the mapped indices placed along the d main diagonals.

It is noticeable that the same number of indices can be mapped into IA matrices of different size and by doing so we can control the redundancy allocation between the two descriptions. Basically, using more diagonals in the IA matrix will lead to a better central distortion performance at the expense of reducing the side distortion performance.

The major problem regarding the IA approach resides in the fact that the range of the quantized indices can be significantly large. This, from an implementation point of view, will lead to a big look-up table that has to be stored in memory. Hence an algorithmic approach that is general enough to allow for a variable number of diagonals (in order to tune the redundancy) has to be designed. In the pseudo code presented the Tables **Direct IA** and the **Inverse IA** is an analytical approach that solves the direct IA (generating the pair of indices) and as well the inverse IA (the reconstruction) for the case of central and side descriptions.

In order to shed light on the technique presented above, let us consider a MDSQ based on a uniform scalar quantizer and a staggered IA matrix (two diagonals). We first define the so-called *central quantizer* by partitioning the input sample range into a number of cells. The central quantizer reflects the video quality we would like to attain when all descriptions arrive at the receiver side. However, the centrally-quantized coefficients are not transmitted to the receiver; instead, neighboring central quantizer cells are grouped together in two distinct ways, resulting in two distinct *side quantizers*, each belonging to a description. This grouping is determined by the Index Assignment (IA) matrix. For each side quantizer, the resulting quantized coefficient is transmitted through the corresponding description.

The grouping shall be done in such a way that the receiver is able to inverse quantize the coefficient according to the central quantizer when all descriptions are received. In case one or more descriptions are missing, the receiver will be unable to narrow down to a single central quantizer cell; hence, the coefficient will be reconstructed to a lesser degree of precision. As a result, the loss of descriptions will result in a graceful progressive loss of picture quality. An example of how the quantizers could be defined in a two-description system is shown in the figure below. Suppose Description 1 signals S^1_0, and Description 2 signals S^2_1, then the coefficient will be reconstructed as C_1.

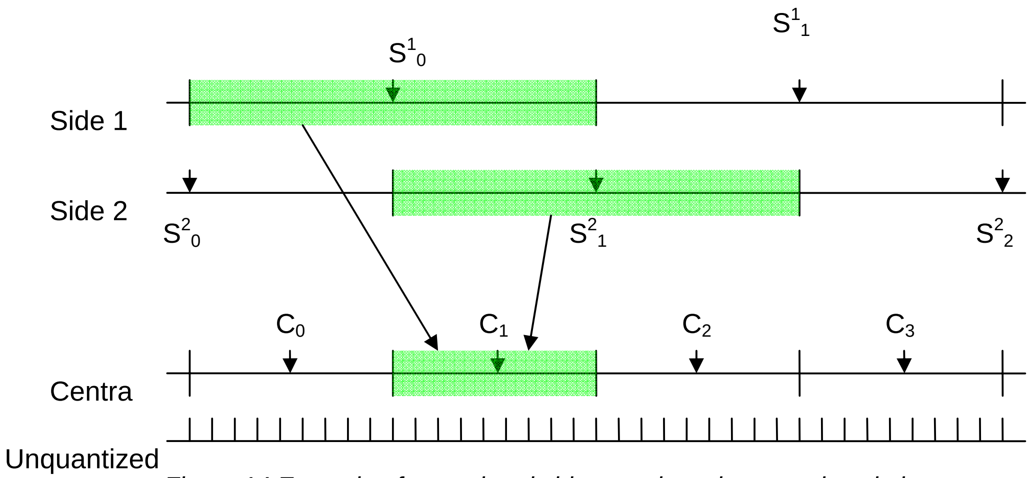


Figure 14 Example of central and side quantizers in a two-description system (only part of axis shown)

Following the pseudo code presented above in the case $NrDiag = 2$ we will obtain for the side reconstruction the centroid of the following reconstructed side cells:

- Side 1 $S_0 = 2n_1$, $S_1 = 2n_1 - 1$
- Side 2 $S_0 = 2n_2$, $S_2 = 2n_2 + 1$

3.3.2 EMDSQ

The main difference between EMDSQ and classical MDSQ consists in the ability of the first to produce layered descriptions allowing for progressive transmission of each distinct description. As described in the previous section a coding system based on MDSQ has the ability to change the amount of redundancy allocated between the descriptions by changing the corresponding set of central and side quantizers. For EMDSQ we have a corresponding set of side and central quantizers at each distinct level. Therefore, in the EMDSQ one can tune not only the overall redundancy, but the redundancy at each distinct quantization level as well. Another property of the EMDSQ consists in their ability to yield uniform embedded central quantizers at each quantization level.

In order to produce layered descriptions we have to rely on the so-called *embedded* IA. The principle behind such an embedded IA consists in designing a recursive matrix decomposition. Consider for instance the IA matrix at level p of dimension $L_p \times L_p$ with a number of N_p indices mapped within. The IA matrix corresponding to the next level $p-1$ is obtain from the IA matrix at level p as follows. Each index different from zero in the IA at level p is considered as block matrix of size $L_{p-1} \times L_{p-1}$ where a number of N_{p-1} elements are mapped. The recursive matrix decomposition is depicted in Figure 15.

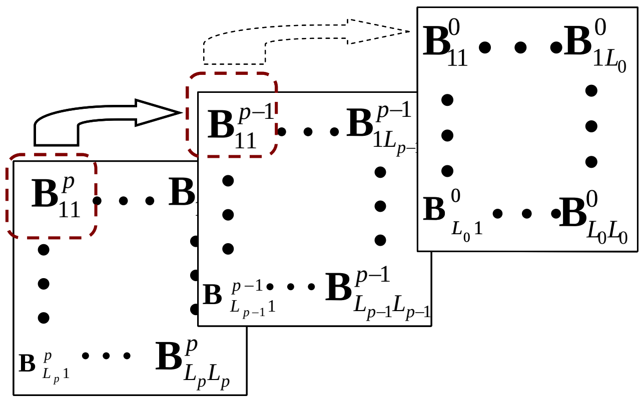
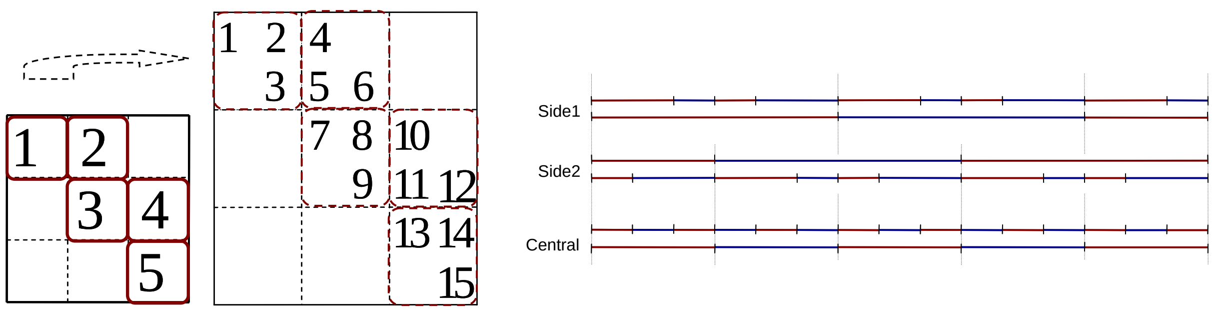


Figure 15 Recursive matrix decomposition



(a) (b)

Figure 16 (a)EMDSQ index assignment strategy for two quantization levels employing disconnected 2 diagonals for the base layer and 3 diagonals for the enhancement layer. (b) Corresponding side and central embedded quantizers

Figure 16 (a) depicts an example of a two level embedded IA matrix. For the first stage we have a two diagonal IA matrix. Each of the indices that are non-zero at the next level are considered as a 2x2 size block matrix, where a number of three indices are mapped. The resulting number of diagonals is three therefore the redundancy was reduced in comparison with the first level. The corresponding side and central embedded quantizers are depicted in Figure 16 (b).

Consider the block matrix defined by its blocks as follows $\mathbf{B}_{ij}^{p+1}=[\mathbf{B}_{mn}^{p}]_{1\le m,n\le L_p}$. The conditions that need to be satisfied by each $[\mathbf{B}_{mn}^{p}]\neq 0$ when designing such a recursive matrix in order to obtain uniform central quantizers are:

- The central quantizer at the finer quantization level is uniform.
- The number of indices mapped in each such block is constant
- The mapped indices each such block are consecutive

The redundancy allocation among the two descriptions for different quantization levels can be tuned by varying the parameters L_p and N_p with $N_p \le (L_p)^2$.

3.3.3 Enabling Quality and Temporal Scalability in MD-SVC based on EMDSQ

This section presents a quality and temporally scalable video coder which employs EMDSQ in order to output two descriptions. One of the descriptions will be send via the DVB and the second description will be send via the WiMax networks.

The starting point is the SVC video coder as presented in Section1.1. However, in order to obtain a MD-SVC coder, a corresponding MD block has to be included into the video coding scheme. In Figure 17 we depict the block scheme of such video coder. The principle behind is that the quantized indices instead of being directly entropy encoded are first processed by an IA matrix. Further, each separate description is entropy encoded and the motion vector information is added. In this way the two resulting descriptions are independently decidable, since they contain all the necessary information. Additionally, we can tune the level redundancy introduced between the indices contained in the two descriptions corresponding to the same original macroblock. It is important to notice that the motion vector information is completely redundant and as a result, the loss of one description will not affect the motion vectors.

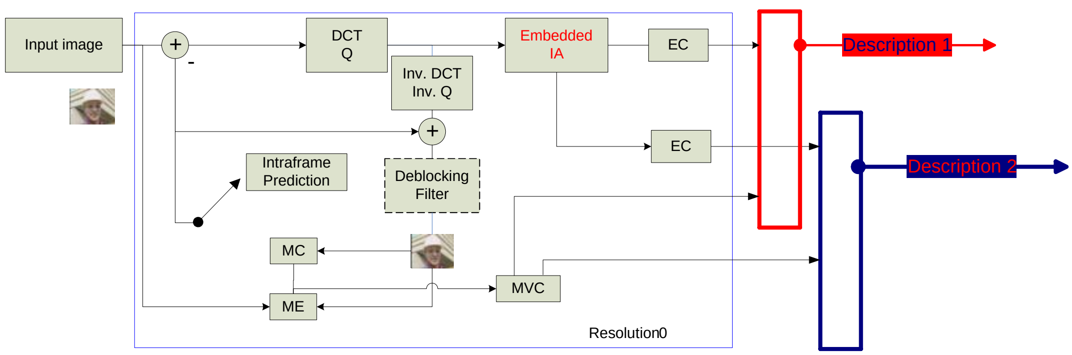


Figure 17 MD-SVC block scheme

In order to have as output fine-grain quality scalable descriptions, the employed IA has to be an embedded IA. The SNR base layer is produced by mapping in the quantized indices corresponding to the coarser scalar quantizer into a corresponding IA matrix. This matrix is designed according to the rules described in Section 3.3.1. Further, in order to provide the next quality layer, each index mapped into the IA matrix is considered as a block matrix. The resulting multiple description indices are further send to the entropy encoder.

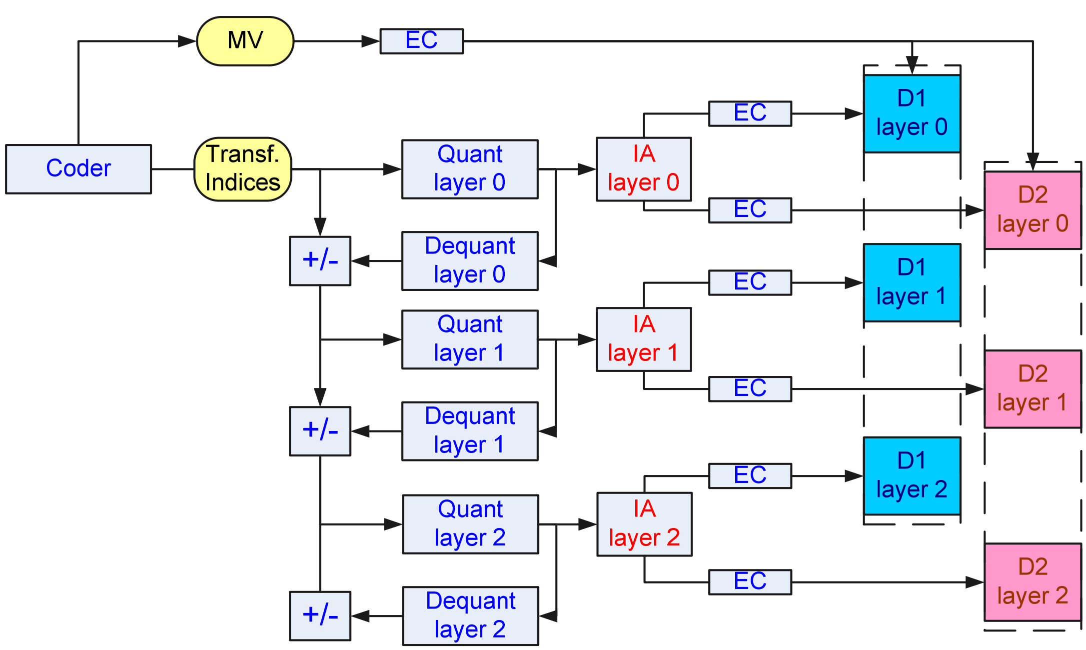


Figure 18 Block scheme describing the way scalable MDC is enabled via embedded IA in the considered SVC architecture.

A block scheme synthesizing the way we provide both scalability and MDC via an embedded IA performed at each quantization level is depicted in Figure 18. Practically, the first central quantizer, which is the coarsest quantizer employed by SVC, produces the base-layer in an identical manner as in SVC. The IA employed on the base-layer (layer 0) generates the two side-descriptions for the base-layer. One notes that in a full-redundancy case, the two side-descriptions for the base-layer can be identical. The subsequent step includes the inverse quantization step performed by using the inverse quantizer of level 0. The error in between the original (unquantized) transform coefficients and their base-layer version is fed into the central quantizer of level 1 and the process is repeated.

One observes that multiple descriptions are created at each quantization level by using an IA per level. This approach preserves scalability (i.e. both central and side descriptions are scalable), and enables resilience against transmission errors via MDC.

We may conclude that the proposed MD-SVC has the following features:

- Provides two distinct descriptions due to the MD module incorporated into the video coding scheme. Each of the distinct description is independently decidable.
- Each of the description has a layered representation allowing for rate adaptation according to the available bandwidth and users' needs.
- The redundancy between the two descriptions can be independently adjusted for every different quality layer providing unequal error protection based on the importance of the information contained in the transmitted stream.

It is important to remark that in the case of such MD video coding system the adaptation of the amount of redundancy to variable network conditions can be achieved in two ways: (1) employ different multiple description quantizers for different redundancy levels, or (2) control the number of description produced and transmitted for each layer. The use of the second approach in an scalable MD-SVC setting has been demonstrated in a recent paper of ours [9].

Figure 19 shows initial measurement of the central and side PSNRs (Luminance) obtained for the BUS sequence.

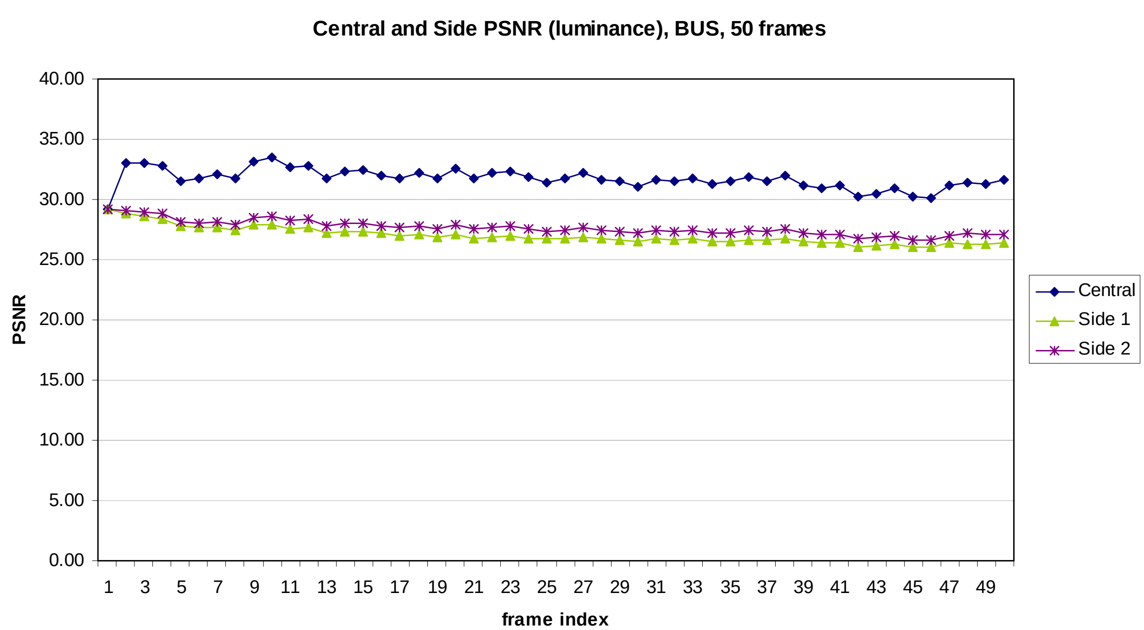


Figure 19 The central and side PSNR (Luminance) for the BUS sequence shows avarage PSNR rates of 31.70 dB for the central description, 26.99 dB for side description 1 and 27.64 dB for side description 2.

These results demonstrate that receiving any of the two side descriptions results in a relatively similar compression performance, while receiving both, results in a significantly improved performance (more than 4 dB), at an expense in rate. Further experiments with the proposed MD-SVC system in error-prone conditions will reveal the attainable performance in the considered transmission scenario.

3.3.4 Enabling Spatial Scalability in MD-SVC based on EMDSQ

Figure 20 illustrates the proposed approach that will be followed in order to enable spatial scalability and in the same time produce scalable multiple descriptions at each spatial layer.

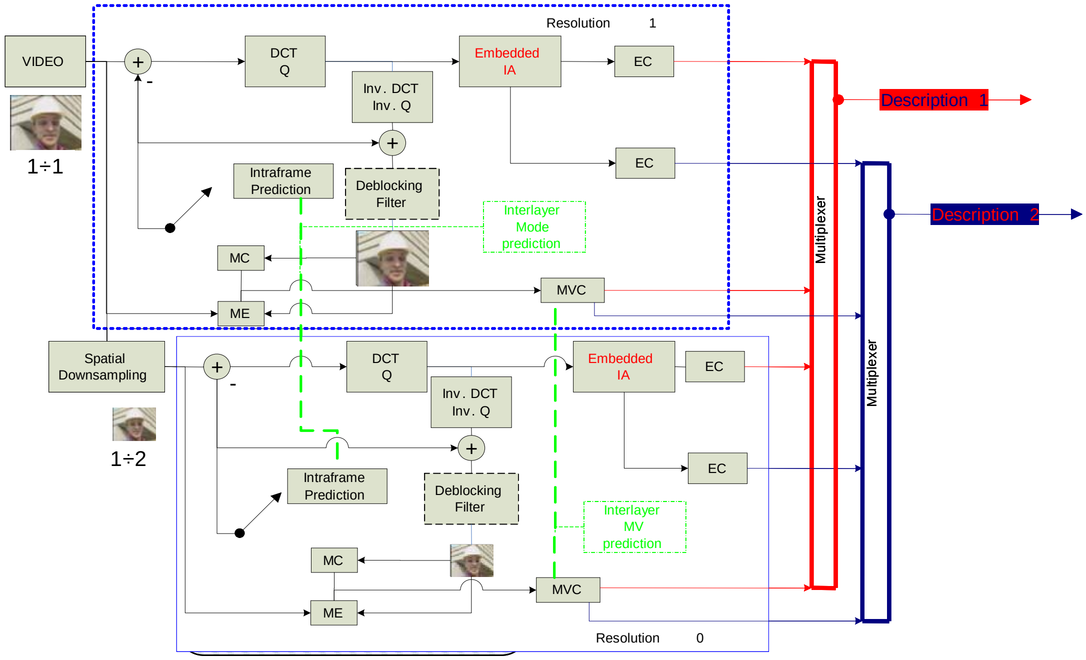


Figure 20 Approach simultaneously enabling spatial scalability and scalable multiple descriptions at each spatial layer.

Basically, the system extrapolates the basic steps followed in order to enable spatial scalability in SVC. Each lower-resolution layer is produced by spatial down-sampling by starting from the higher-resolution spatial layer. Inter-mode and inter-layer MV prediction are (optionally) used in between the consecutive spatial layers. As depicted in the figure, at each spatial layer, an embedded IA is used in order to enable MDC and in the same time provide quality and temporal scalability.

4 Gateway and Decoder

This section describes the design of the video data path of the Gateway (in the scenario of retransmission to a last-mile network) and the Terminal (in the scenarios without a last-mile network) components. More specifically, the synchronization between both received descriptions and the combination of both descriptions into a single coded video stream shall be investigated.

It is noticeable that the Gateway and the Decoder should follows the same procedure till a certain point represented by the so called MD combiner. After the MD combiner bloc we either have the last mile WLAN that will retransmit one description video stream either the SVC decoder. Note that the SVC decoder has to be build under the constraint of being able to bypass its own entropy decoder as described in D1.1 “User terminal requirements”.

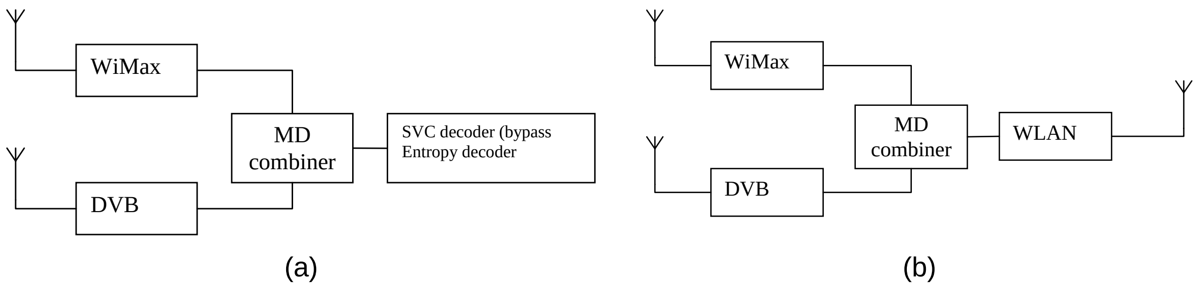


Figure 21 Basic structure of H2.64/AVC for a macro-bloc Comparative block scheme between SUIT Terminal and Gateway

Figure 21 depicts the terminal block scheme and the gateway block scheme emphasizing on the main difference between the two.

In the following we will describe the data flow that is common both to the Terminal and Gateway.

Correctly received link-layer packets of both descriptions shall be reassembled and their UDP and IP headers removed as they traverse the UDP/IP protocol stack of the corresponding transceiver. For each of both descriptions, the resulting stream of RTP packets is delivered to a RTP depacketization and decapsulation module. This module shall perform the following tasks:

- Reordering of the received RTP packets based on their RTP sequence number, since the User Datagram Protocol guarantees no in-order arrival of packets;
- Depacketization of RTP Aggregate Packets (if used); i.e. recovering the NAL units one by one;
- Delivery of the NAL units in decoding order.

The resulting streams of NAL units may not be synchronized with each other due to unequal delays in the WiMAX and DVB transmission chains. Furthermore, due to packet loss in one of both networks NAL units that are recovered from Description 1 may have been erased in Description 2, or vice versa. Therefore, synchronization between both NAL unit streams is required before they are delivered to the MDC Combiner.

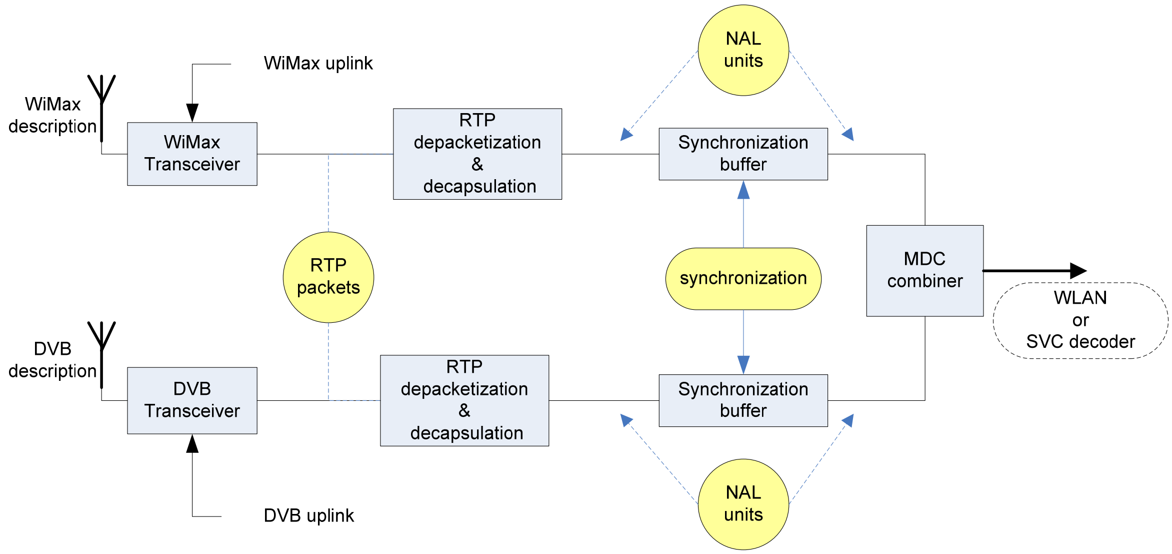


Figure 22 schematic view of Gateway and Terminal common blocks (video data path)

In the following we will describe the MD combiner for the three specific MD coding schemes, namely Unbalanced MD, MDC based on redundant slices and MDC based on EMDSQ.

For the **Unbalanced MDC** the at level of the MD combiner are arriving synchronized packets representing the output of the two synchronizations buffers. The role of the MD combiner is reduced to dropping the redundant NAL units and signalizing the missing NAL units (in the case of errors in both descriptions).

When using **MDC based on redundant slices**, combining both descriptions into a single compliant bitstream can be performed in the network abstraction layer (NAL). At the playout side, the MDC generator will tag each NAL unit with a sequence number, a frame number and/or a timestamp. This will aid in synchronizing and combining both descriptions with each other.

The requirements for the MDC combiner in the terminal/gateway depend upon the approach used:

1. When using the H.264/AVC support for signalling redundant slices, the combiner has to parse the slice header of all coded slices in order to find out whether or not the slice is redundant, and must then decide whether it can be discarded. Unavoidably, this increases the computational requirements and hence the delay of the combiner. As an alternative, when a decoder with support for H.264/AVC redundant slices is available, the task of removing the redundant slices can be offloaded to this decoder. However, this will increase the bandwidth requirement for the last-mile WLAN network, since both redundant and non-redundant slices need to be transmitted through this network.
2. When not using the H.264 syntax to mark redundant slices, the MDC combiner will instead investigate the numbers with which each NAL unit was tagged at the playout side. This allows the combiner to detect which slices were lost during transmission, and hence to correctly replace lost slices with their redundant copy.

In the case of and **MDC based on EMDSQ** the MD combiner module for use with the EMDSQ technique requires access to the quantized coefficients of both descriptions; therefore entropy decoding shall first be performed on both incoming video streams. Then, the actual reconstruction of both side-descriptions into a single description is performed, according to the number of descriptions available. Entropy coding will have to be re-applied in order to obtain an SVC compliant bitstream to be sent over the local WLAN. The EMDSQ combiner is in fact an inverse IA matrix which is mapping pair of coefficients into the reconstructed central descriptions as described in Section 3.3.

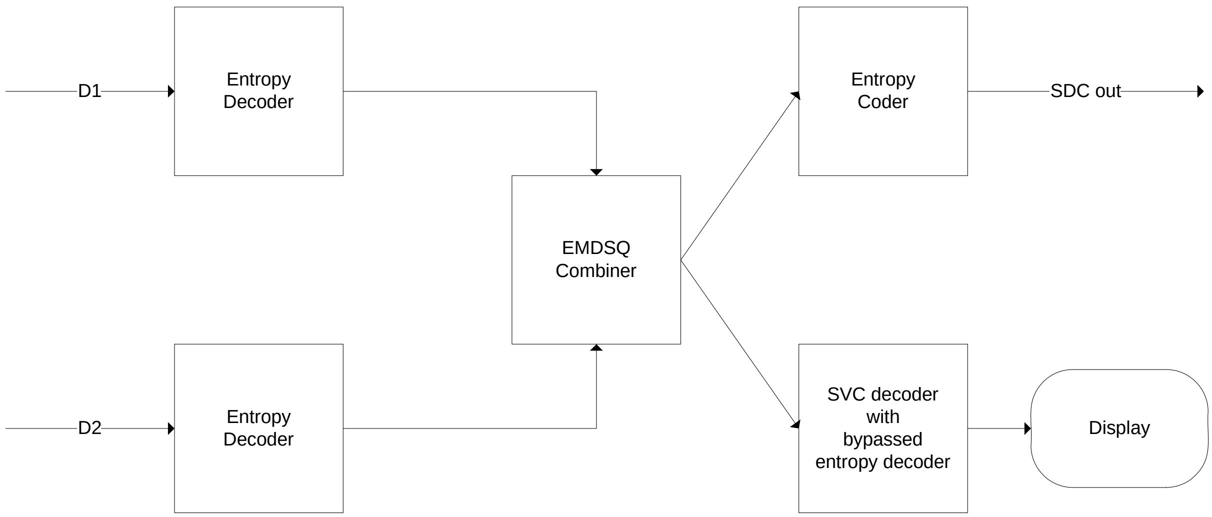


Figure 23 EMDSQ Combiner

5 Conclusions

This document presents the design steps that have been followed in the design of scalable multiple description video coding systems enabling the three forms of scalability (temporal, quality and spatial scalability), as well as resilience against transmission errors by relying on MDC principles.

The proposed MD-SVC approach is able in the first place to provide layered descriptions, allowing for rate adaptation among each of the available networks, namely WiMax, DVB and WLAN. This is possible since the coder inherits its scalable proprieties from the SVC coder. The choice was based on the coder performances and also in the fact that it represents a standardized solution in scalable video coding.

Additional to providing scalability, the proposed MD-SVC relied on MDC principles. These are enabled by following three different techniques:

1. Unbalanced MDC
2. MD based on redundant slices,
3. MD based on EMDSQ.

All employed MDC methods have as starting point the MD paradigm of providing several descriptions in order to overcome the channel impairments each one of them is conceptually different. Therefore the proposed solution will not be limited in just tuning a specific approach to the SUIT framework but to provide a broader solution based on conceptually different approaches.

6 Acronyms

AVC	Advanced Video Coder
ASO	arbitrary slice ordering
CABAC	Context Adaptive Binary Arithmetic Coding
CAVLC	Context-Adaptive Variable Length Coding
CIF	Common Intermediate Format
DCT	Discrete Cosine Transform
DVB	Digital Video Broadcasting
DVB-IPI	Digital Video Broadcasting- Internet Protocol Infrastructure
DVB-H	Digital Video Broadcasting-Handhelds
DVB-RCT	Digital Video Broadcasting- Return Channel Terrestrial
DVB-T	Digital Video Broadcasting-Terrestrial
EMDSQ	Embedded Multiple Descriptions Scalar Quantizers
FEC	Forward Error Correction
FMO	macro-block ordering
FGS	Fine Grain Scalability
IA	Index Assignment
JSVM	Joint Scalable Video Model
HDTV	High Definition Television
MCTF	motion-compensated temporal filtering
MDC	Multiple Description Coding
MDSQ	Multiple Descriptions Scalar Quantizers
MD-SVC	Multiple Description Scalable Video Coder
MPEG	Moving Picture Expert Group
NAL	Network Abstraction Layer
SVC	Scalable Video Coding
SDC	Single Description Coding
VCL	Video Coding Layer
WIMAX	Wireless Local Area Network
WLAN	Wireless Local Area Network

7 References

[1] R.Schäfer, T.Wiegand and H. Schwarz, The emerging H.264/AVC standard – EBU Technical Review – January 2003

[2] H. Schwarz, D. Marpe and T. Wiegand, MCTF and Scalability Extension of H.264/AVC – Proc. Picture Coding Symposium – San Francisco, USA, December 2004

[3] H. Schwarz, T. Hinz, D. Marpe and T. Wiegand, Constrained Inter-Layer Prediction for Single-Loop Decoding in Spatial Scalability – Proc. ICIP, Genova, Italy, September 2005

[4] H. Schwarz, D. Marpe, T. Schierl and T. Wiegand, Combined Scalability Support for the Scalable Extension of H.264/AVC – Proc. VCIP, Beijing, China, July 2005

[5] V. A. Vaishampayan, “Design of multiple description scalar quantizers," IEEE Trans. Inform. Theory, vol. 39, no. 3, pp. 821-834, 1993.

[6] V. K. Goyal, “Transform coding with discrete transforms," IEEE Transactions on Information Theory, vol. 46, no. 2, pp. 465-473, 2000.

[7] W. Jiang and A. Ortega, “Multiple description coding via polyphase transform and selective quantization," in SPIE Int. Conf. Visual Comm. Image Proc. VCIP'99, San Jose, USA, 1999, pp. 998-1008.

[8] A. Albanese, J. Blomer, J. Edmonds, M. Luby, and M. Sudan, \Priority encoding transmission," IEEE Transaction Information Theory, vol. 42, pp. 1737-1744, 1996.

[9] F. Verdicchio, A. Munteanu, A. Gavrilescu, J. Cornelis, and P. Schelkens, "Embedded Multiple Description Coding of Video," IEEE Trans. on Image Processing, vol. 15, no. 10, pp. 3114-3130, Oct. 2006.