|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  9th Meeting: Geneva, CH, 27 April – 7 May 2012 | Document: JCTVC-I0190r1 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Low Complexity Scalable Extension of HEVC intra pictures based on content statistics** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Sébastien Lasserre,  Fabrice Le Léannec,  Eric Nassor  Canon Research Centre France Rue de la Touche Lambert  35510 CESSON-SEVIGNE, FRANCE | Tel: Email: | +33(0)299876800  [sebastien.lasserre@crf.canon.fr](mailto:sebastien.lasserre@crf.canon.fr)  [fabrice.leleannec@crf.canon.fr](mailto:fabrice.leleannec@crf.canon.fr)  eric.nassor@crf.canon.fr |
| *Source:* | Canon Research Centre France | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

This contribution presents a new approach for scalable extension of HEVC INTRA pictures. This scalable INTRA codec design targets coding efficiency together with very low complexity. Spatial random access and a high degree of parallelism are two additional targeted features. The proposed scalable INTRA codec employs only one coding mode, which is inter-layer intra prediction, which provides low complexity. Coding efficiency is obtained through statistical modeling of DCT channels to encode, rate distortion optimal quantifiers that are pre-computed off-line, coupled with a distortion allocation process between DCT channels. Overall, bitrate increase of 5.7% is obtained relative to HEVC single layer coding on tested sequences in dyadic spatial scalability mode and 2.1% in SNR QP+3 scalability.

Finally, non-contextual, non-adaptive arithmetic coding provides the spatial random access feature.

# Introduction

## Low complexity

This contribution proposes a scalable extension of HEVC for INTRA pictures and it follows a preceding contribution [5] presented a few months ago.

Some already proposed scalable extensions perform an HEVC-like encoding of enhancement pictures, with the introduction of some inter-layer prediction tools [2][3]. In these approaches, the enhancement layer coding process involves RDO choice between multiple intra predictions modes, where inter-layer prediction modes are added to modes already present in HEVC. This, in addition to the base layer coding process, results in an encoder complexity that is higher than that of non-scalable HEVC encoding.

On the contrary, the main idea behind this contribution is that a successful scalable standard should specify a scalable codec that provides **low complexity** together with **good coding efficiency**. Scalability should be used as **a means of reducing HEVC complexity**. Such feature is particularly useful for embedded real time UHD video encoders in handset devices, where complexity and memory bandwidth are limited.

Practically, it is desired to design a scalable video coding system, with an overall complexity that is closer to the complexity of the dyadic base layer than that of single-layer encoding. This complexity requirement is illustrated by **Figure 1**.

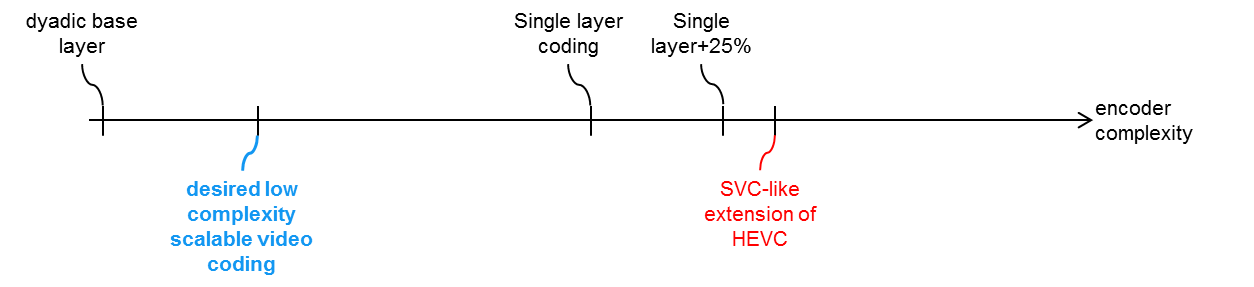


Figure 1: Complexity target for the proposed scalable video coding method

To this end, in this contribution, the enhancement layer encoding **employs only one INTRA prediction mode**. This single coding mode corresponds to **inter-layer intra prediction**, also known as I\_BL in the former scalable extension of H.264/AVC [1].

As a consequence, no coding mode decision is required in the enhancement layer, which strongly reduces the encoder complexity, compared to standard video coding schemes, where a number of directions for spatial prediction are being evaluated.

## Parallelism and spatial random access

An additional requirement for the proposed scalable approach is to provide spatial random access feature in the compressed video bit-stream, as well as a high degree of parallelism in the decoding and encoding processes. These two features are obtained through the removal of any dependency between successive spatial entities in the picture. Therefore, in addition to the suppression of spatial directional prediction, **non-contextual and non-adaptive entropy coding/decoding** is required.

The scalable video coding approach introduced in this document conforms to the requirements listed above.

## Codec overview

**Figure 2** shows the overall architecture of the proposed INTRA scalable encoder of the enhancement layer above a block-based classical encoder for the base layer. Of course, in the framework of the meeting, the base layer encoder is a HEVC encoder, but our method is not limited to this particular case. It is not even limited to a block-based encoder because the enhancement Intra encoding works on the residual pixel picture only without using coding information from the base layer.

The basics of the proposed architecture are the following. The input full resolution original picture is down-sampled to the base layer resolution level, and is encoded with HEVC (using HM 4.0 for this contribution). For the down-sampling, standard H.264/SVC filters are used.

Then, a spatial residual picture is computed as the difference between the original picture and the base layer reconstructed picture, which is up-sampled to the top layer resolution in case of spatial scalability (in the SNR case, there is no such up-sampling). The DCTIF interpolation filters of quarter-pixel motion compensation in HEVC are used for the up-sampling. As can be seen, this global architecture corresponds to classical scalable INTRA coding, where the spatial intra prediction and coding mode decision steps have been removed.

The resulting residual picture is then encoded according to the new low complexity approach described in much more details in the rest of the document. The resulting coded enhancement layer consists in coded residual data as well as some parameters used to model DCT channels of the residual picture.

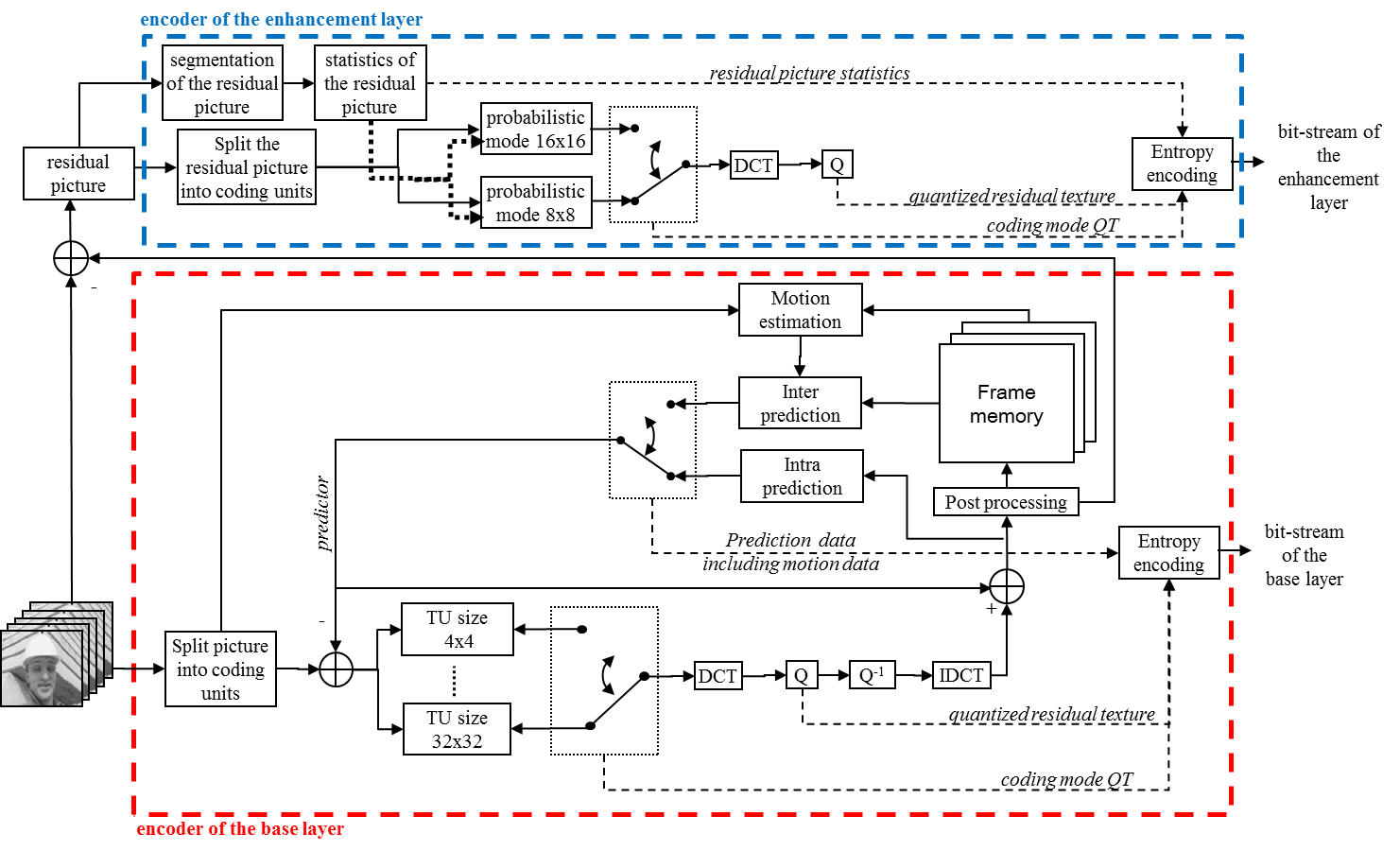


Figure 2: overview of the enhancement INTRA picture coding process

**Figure 3** shows the overall architecture of the associated scalable INTRA video decoder. The input bit-stream to that decoder consists in the HEVC-coded base layer, enhancement residual coded data, and parameters of the DCT channels in the enhancement residual picture.

At first, the base layer is being decoded, which provides a reconstructed base picture. The reconstructed base picture is up-sampled to the enhancement layer resolution. Then, the enhancement layer is decoded.

The associated residual data decoding process, described in details later in the document, is used, and provides successive de-quantized DCT blocks. These DCT blocks are then inverse transformed and added to their co-located up-sampled block. The so-reconstructed enhancement picture finally undergoes HEVC low-complexity post-filtering processes, i.e. de-blocking filter and sample adaptive offset.

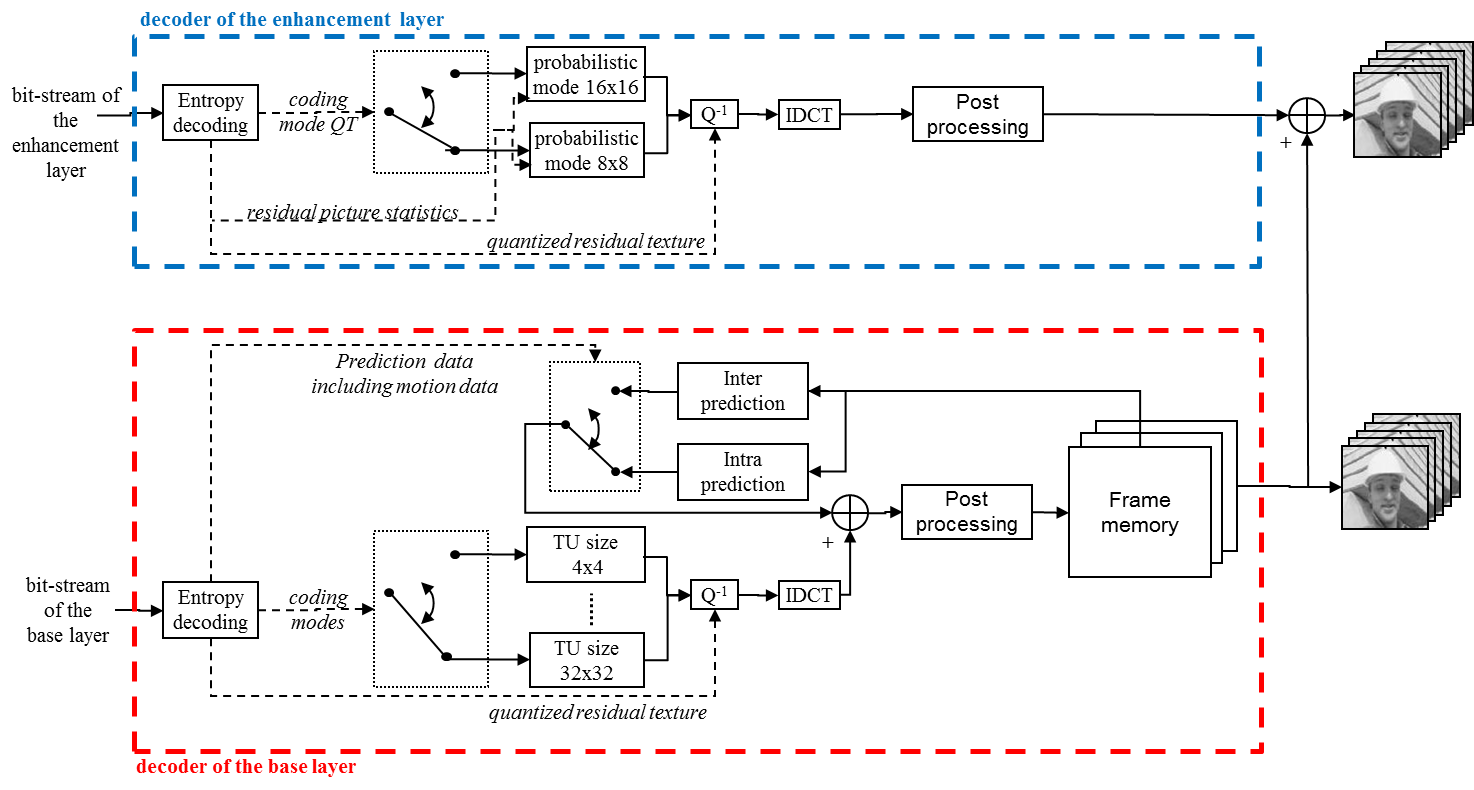


Figure 3: overview of the enhancement INTRA picture decoding process

## Residual data coding/decoding process

This section explains the main principles of the residual texture coding/decoding performed in the enhancement layer and gives an overview of the content of the document.

The input to the encoder consists in a set of DCT blocks. With respect to the transform, 2 DCT transform sizes are supported (16x16 and 8x8) but blocks are classified into so-called **block types** which consist in a given block size and block energy band. The block type is decided through an initial guess based on the (morphological) activity analysis of the residual picture (see section 3) and this guess is refined thanks to a **competition process between block types** (section 11).

Then, the corresponding DCT transform is applied to each block and, for each block type, statistics (taken on all the blocks of the block type) of the DCT coefficients are computed. The transforms used are the 8x8 and 16x16 DCT of HEVC.

By using the computed statistics, **Generalized Gaussian Distribution** (GGD) is fitted to each DCT coefficient channel of each block type (section 4). Following its GGD, a quantifier (taken from a pool of quantifiers) is assigned to each DCT channel through a process of balancing the merits of encoding between channels of each block types, between block types and between frames and colour components (sections 7, 8 and 9). Quantifiers are **non-uniform scalar quantifiers** defined by a set of quantization intervals and associated de-quantized sample values. The pool of quantifiers is available on both the encoder and on the decoder side; this pool is made of many quantifiers (corresponding to various rates of encoding) that are pre-computed off-line, through the Chou-Lookabaugh-Gray rate distortion optimization process [4]. This allows being optimal (in the rate-distortion sense) on GGD channels (sections 5 and 6). The global encoding process is described in section 10.

We first give some experimental results in section 2 showing the efficiency of the method.

# Experiment

## Residual data coding/decoding process

Two test sets were used to run the proposed INTRA scalable codec: one for spatial scalability and one for SNR scalability. Class A 8 bits and class B sequences were tested. Testing conditions were chosen as close as possible to those of [2]. Original class B sequences were cropped to size 1920x1072 in order to make the base layer a multiple of 8 pixels.

With respect to spatial scalability, 2 dyadic layers were encoded. The JSVM down-sampling process was used to generate the original base layer. The base layer was encoded with HM4.0 in Intra-Loco configuration. Three encoders were compared in the encoding of the enhancement layer.

1. **HM4.0 single layer.** The original sequence was encoded with HM4.0 in non-scalable mode, with QP values equal to 22, 27, 32 and 37.
2. **HM4.0 simulcast.** The HM4.0 was used to encode both base and enhancement layer, without any interaction between them. The obtained bit-rate corresponds to the sum of base and enhancement layer bitrates. The base layer QP was set to the top layer QP minus 2. Of course the obtained PSNRs are equal to those of single layer configuration.
3. **Low complexity INTRA codec.** The proposed low complexity INTRA scalable codec was used to encode the enhancement layer, at a PSNR value that is close to the PSNR obtained with simulcast and single-layer configurations.

Regarding SNR scalability, the 3 above coding configurations were used in SNR scalability context. The base layer QP was set equal to the top layer QP plus 3.

Next section explains the comparative BD-PSNR results obtained with each coding configuration listed above.

## Obtained performance

Table 1 lists the BD-PSNR results obtained in the driven experiments for spatial dyadic scalability. Average bit-rate increases, respectively relative to HM4.0 simulcast and HM4.0 single layer, are shown in the two right columns. As can be seen, an average bitrate increase of 5.0% and 5.9% is obtained respectively for class A and class B sequences, compared to HM4.0 single layer coding. An average bitrate saving of 31.4% and 26.2% is obtained over simulcast.

Table 2 depicts the BD-PSNR results obtained in SNR scalability mode. It shows that the average rate distortion cost of scalability compared to single layer coding is 2.9% on class A sequences, and 1.7% on class B sequences.



Table 1: obtained rate distortion performances with dyadic spatial scalability



Table 2: obtained rate distortion performances with SNR scalability

As a conclusion to this section, the proposed INTRA scalable coding method shows very good rate distortion performance, despite its low complexity.

# Initial segmentation into block types

The concept of block size has been generalized to what will be called **block types**.

At first, the segmentation of the residual frame into various block type is performed following a criterion of **residual activity**. It is well known that, most of time, a part of a frame with a lot of activity is better compressed by using small blocks; on the contrary less active (flat region) parts are better compressed by using large blocks. In HEVC, transform unit size is either 32x32, 16x16, 8x8 or 4x4. We used a similar approach but only with two sizes (16x16, 8x8) for the luminance. This choice is not restrictive and any size (even rectangular) of blocks may be used.

## Morphological gradient

The residual activity is measured by a **morphological gradient**. Other methods like local energy or the Laplace’s operator have exhibited very similar performance. In our case, the morphological gradient is the difference between a dilatation and an erosion of the luminance residual frame. The basics of mathematical morphology can be found at [7] or in [8]. A morphological gradient frame is computed and the segmentation is decided based on this frame.

## Generalized quad-tree of block types

A block type is not only defined by its size but also by its **index of energy**. We found out that, for the luminance, the following list of block types (this is just a list of type labels) is a good trade-off between the cost of the quad-tree and the improvement of block compression:

* 16x16 bottom
* 16x16 low
* 16x16
* 8x8 low
* 8x8
* 8x8 high

Other combinations are possible and the optimal one may depend on the video content.

The splitting of the block size is performed by a top-down algorithm. We start from a 16x16 grid and we compute the integral of the morphological gradient on each 16x16 block. If this integral is higher than a given threshold, the block is divided into four smaller 8x8 blocks.

Once the block size is decided, we group blocks into bands of morphological integral; three bands for the 16x16 blocks and three other bands for the 8x8 blocks. Basically, these bands are delimited by threshold on the morphological gradient integral. The choice of the thresholds is a very sensitive problem which can dramatically impact the compression performance of the codec.

This provides a quad-tree for the luminance block types. The chrominance block type is entirely inferred from the luminance block type such that there is no need for a chrominance quad-tree.

# Modelling of the residual enhancement frames by block type

For a given block type, which determines in particular the block size, we fit a model for the channel of the DCT coefficients of the block type.

## Generalized Gaussian Distributions

It is common to model the noise residual of video data by a Generalized Gaussian Distribution (GGD). Naturally, this distribution has zero mean, such that one writes



where the GGD follows the two-parameter distribution



The standard deviation of the noise is proportional to the parameter.



The parameter  is called the exponent of the distribution.

## Fitting of Generalized Gaussian Distributions

Let us have the moment of order k of the absolute value of a GGD defined as follows

 .

By simple integral calculation, one finds an analytical formula for the moments.



One also observes that well-chosen ratios of moments do not depend on the parameter; for instance we get



So, if one models a random variable by a GGD, it is easy to estimate the value of the parameter by computing the above ratio of the two first and second moments and inverse the above function of. Practically, one may tabulate this inverse function instead of computing costly Gamma functions.

Then, the still-to-be-determined parameteris estimated thanks to the second moment by the equality

.

This method is very simple and robust for determining the best GGD model of a random variable.

## Modelling of transformed blocks

We observed experimentally that the DCT coefficients of the spatial residual picture (see section 1.3) are well modelled by GGD’s. Of course, they cannot be all modelled by the same parameters and, practically, the parameters depend on

* the video content; this means that the parameters must be computed every frame or every *n* frames for instance
* the index of the DCT coefficient; each DCT coefficient has its own behaviour
* the base encoding mode; typically Intra blocks do not behave the same way as Inter blocks, and blocks with a coded residual in the base layer do not behave the same way as blocks without such a residual.

If the enhancement frame is divided into 8x8 blocks, one has to determine the parameters of 64 channels for each base mode. Moreover, luminance component Y and chrominance components U and V have dramatically different source contents; they must be encoded in different channels. If one decides to encode Y on one channel and UV on another channel, 128 channels are needed for each base mode.

This may seem to be a lot of parametric data to be put into the video stream, but the experience proves that this is quite negligible compared to the volume of data needed to encode the residues of Ultra High Definition (4k2k or more) videos. It is evident that such a technique could not be used for very small videos because the parametric data would be too costly.

# The rate-distortion coding problem per block type

## PSNR and control of DCT coefficients

Usually, the quality of a video is measured by the so-called PSNR which is a measure of the L2-norm of the error of encoding in the pixel domain. However, most of video codecs compress the data in the DCT-transformed domain in which the energy of the signal is much better compacted. Here, we show the direct link between the PSNR and the error on DCT coefficients.

Let us considerer a residual block and let  be its IDCT pixel base in the pixel domain as shown on **Figure 4**. If one uses the so-called IDCT III for the inverse transform, this base is orthonormal

.

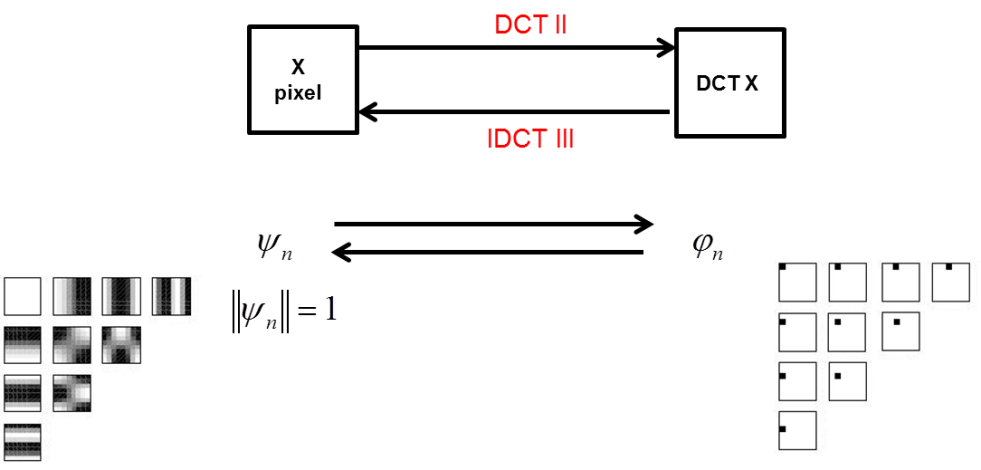


Figure 4: pixel and DCT bases of a block

On the other hand, in the DCT domain, the unity coefficient values form a base  which is orthogonal. One write the DCT transform of the pixel block X as follows



where  is the value of the n-th DCT coefficient. A simple base change leads to the expression of the pixel block as a function of the DCT coefficient values.



Let us denote  the value of the de-quantized coefficient  after decoding. By linearity, one sees that the pixel error block is given by



And the mean L2-norm error on all blocks, the PSNR in other words, is just



where  is the mean quadratic error of quantization on the n-th DCT coefficient. So, one understands that the control of the video quality is equivalent to the control of the sum of the quadratic errors on DCT coefficients. In particular, the individual control of each of the DCT coefficient is a priori a sub-optimal control.

## Rate minimization under distortion constraint

Let us suppose that the PSNR target corresponds to a given L2 distortion value . The minimization of the rate under this pixel distortion constraint can be stated as

 **(A)**

where  is the total rate made of the sum of individual rates  for each DCT coefficient. In case the quantization is made independently for each DCT coefficient, the rate  depends only on the distortion  of the associated n-th DCT coefficient.

One also understands that the above minimization problem is fulfilled only by optimal quantifiers which are solution of the problem

 **(B)**

This statement is simply proven by the fact that if a quantifier is not optimal, then another quantifier with lower rate but the same distortion can be constructed (or obtained). So, if one uses this other quantifier, the total rate has been diminished without changing the total distortion; this is in contradiction with a minimal solution of the first problem **(A)**. We have proven that the rate-distortion minimization problem **(A)** can be split into two consecutive sub-problems without changing the optimality of the solution.

* first, determining optimal quantifiers and their associated rate-distortion curves  following the problem **(B)** for GGD channels. The solution to this problem is described in section 6.
* second, we solve the problem **(A)** changed into the problem **(Aopt)** where quantifiers are optimal quantifiers

 **(Aopt)**

The solution to problem (Aopt) is described in sections 7, 8 and 9

# Off-line determination of optimal quantifiers

This section deals with the solving of the problem which consists in finding optimal quantifiers. We consider only 1D quantifiers for a given DCT coefficient, i.e. one quantifier will be used per DCT coefficient for the quantization process.

## Voronoi cell based quantifiers

A quantifier is made of  Voronoi cells which are intervals , called quantum , and each cell has a centroid , as shown on **Figure 5**. The intervals are used for quantization while centroids are used for de-quantization: a value falling in an interval is coded by the index of this interval and is de-quantized into the centroid value .

Let us suppose that the probability distribution  of the coefficient is given; for instance a GGD as seen on the preceding **Figure 5**. The distortion  of the quantization, on the quantum , is the mean error  for a given distortion function . Usually, this function is a -distance on the real line. So the distortion on one quantum is given by



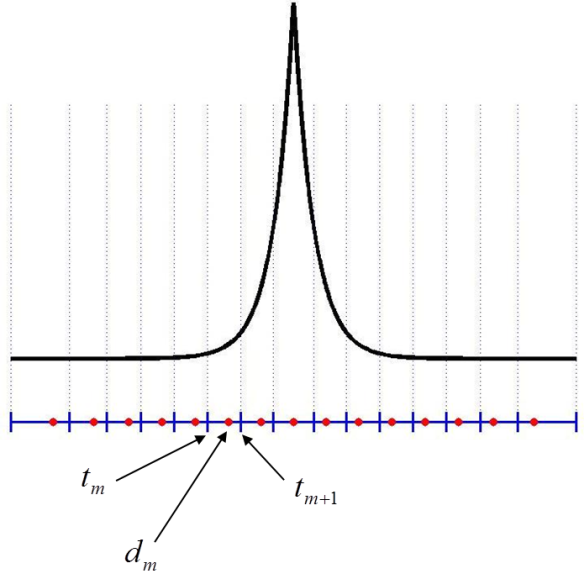


Figure 5: a 1D quantifier made of Voronoi cells

and the total distortion is



The total rate  is the cost of encoding by a Variable Length Code (VLC). At best, this rate reaches the entropy of the quantized data



where  is the probability of  to be in the quantum  and is simply the following integral

.

Practically, the rate-distortion problem may be stated in two different ways.

* rate control: for a given rate , find the quantization which minimizes the distortion 
* distortion control: for a given distortion , find the quantization which minimizes the rate 

This problem is hard to solve because it is a non-linear minimization problem.

## Lagrangian formulation (B\_lambda) of the problem (B)

Instead of the previous complex formulation, we use the so-called Lagrange formulation of the problem. For a given parameter , we determine the quantization in order to minimize the cost function . This way, we get an optimal rate-distortion couple . In case of a rate control for a given target distortion , the optimal parameter  is determined by



and the associated minimum rate is

.

So, it is possible to plot a rate distortion curve which may be computed off-line as well as the associated quantifications. So, we restated problem (B) into a continuum of problems (B\_lambda)

 (B\_lambda)

One understands that the cost can be changed from  to without loss of generality thanks to the extra parameter .

## The algorithm by Chou-Lookabaugh-Gray

Still remains the minimization of the linear problem (B\_lambda) for  to determine the quantization. The well-known Chou-Lookabaugh-Gray algorithm is a good practical way to perform it. It can deal with any distortion distance , but here we present a simplified algorithm for the -distance. This is an iterative process from any given starting guessed quantization.

The position of the centroids  is easy to determine because they must minimize the distortion  inside a quantum, in particular one must verify that . We easily deduce

.

Now, let us consider the cost function



which must be minimized. In particular its derivatives must vanish for an optimal solution.



Let us set the value of the probability distribution at the point . From simple variational considerations, see **Figure 6**, we get

 and .

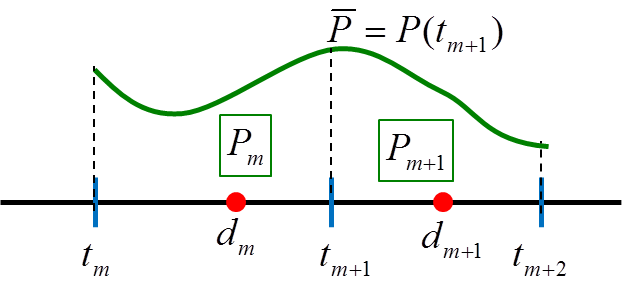


Figure 6: effect of the variation of the position of point tm+1

Then, a bit of calculation leads to



as well as

.

Now, the derivative of the cost is explicitly calculated

,

and this leads to a useful relation between the quantum boundaries and the centroids

.

This formula is already well-known, but a simple demonstration has been given for the sake of completeness.

The algorithm by Chou-Lookabaugh-Gray is an iterative process as described below.

1. Start with arbitrary quanta 
2. Compute the probabilities  by the formula



1. Compute the centroids  by the formula



1. Compute the new quanta  by the formula



1. Compute the cost  by the formula



1. Loop to 2. until convergence of the cost 

So, for a given number  of quanta, each problem (B\_lambda) is solved by this algorithm. As can be seen latter, the problem will be solved for several given values of M.

## Normalization

One easily understands that the parameter alpha  (or equivalently the standard deviation of the GGD) does not play a role on the determination of optimal quantifiers because it is an homothetic parameter which can be moved out of the distortion parameter .

So, only optimal quantifiers with unity standard deviation  need to be determined and GGD will be normalized before quantization, and will be de-normalized after de-quantization. Of course, this is possible because the parameters of the GGD models have been sent to the decoder through the video bit-stream.

## The upper envelope of optimal quantifiers. Tabulation

For a given parameter , the problems (B\_lambda) are solved for many values of the Lagrange parameter  and for various odd (by symmetry) values of the number  of quanta. Thus, one gets a rate-distortion diagram for the optimal quantifiers, as shown on **Figure 7**.

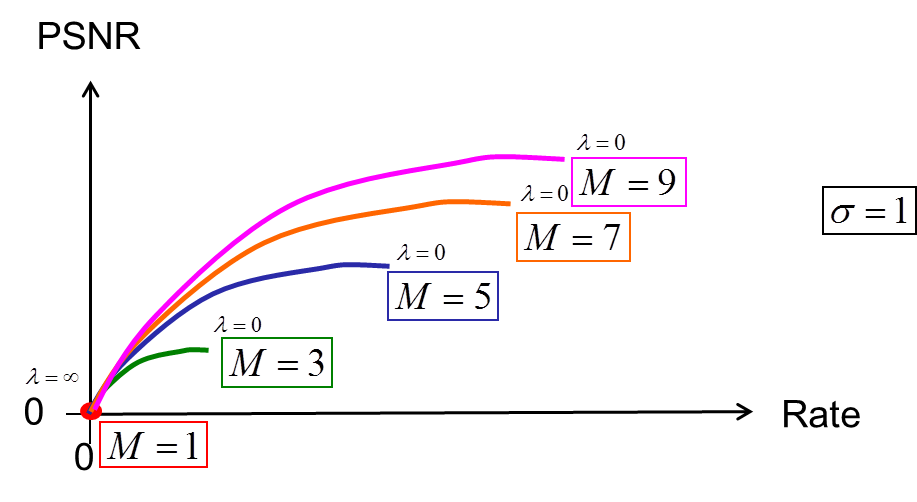


Figure 7: the rate-distortion curves associated to optimal quantifiers for the problems (B\_lambda)

It turns out that, for a given distortion, there is an optimal number of needed quanta for the quantization. In brief, one may say that optimal quantifiers of the general problem (B) are those associated to a point of the upper envelope of the rate-distortion curves making this diagram. This upper envelope is illustrated on **Figure 8**.

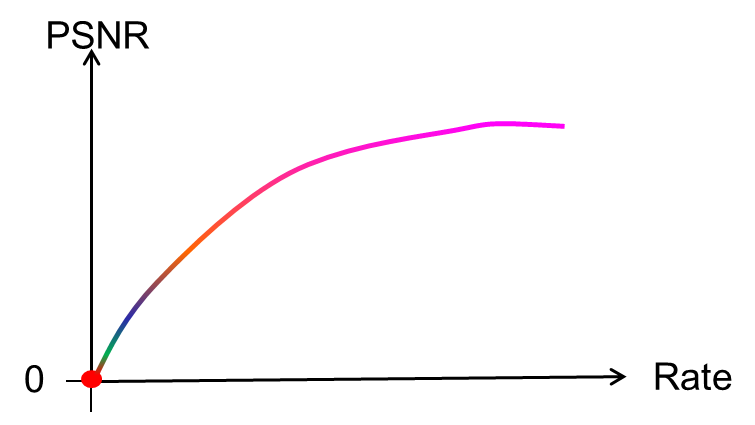


Figure 8: the upper envelope of the rate-distortion curves associated to optimal quantifiers for the problems (B\_lambda)

Practically, one cannot compute optimal quantifiers for all parameters  and all target distortions . We observed that the GGD modelling provides ’s almost always between 0.5 and 2, and that only a few discrete values are enough for the precision of encoding:  is tabulated every 0.1 in the interval [0.2, 2.5]. For each value of , a rate-distortion curve is obtained as shown on **Figure 9**.

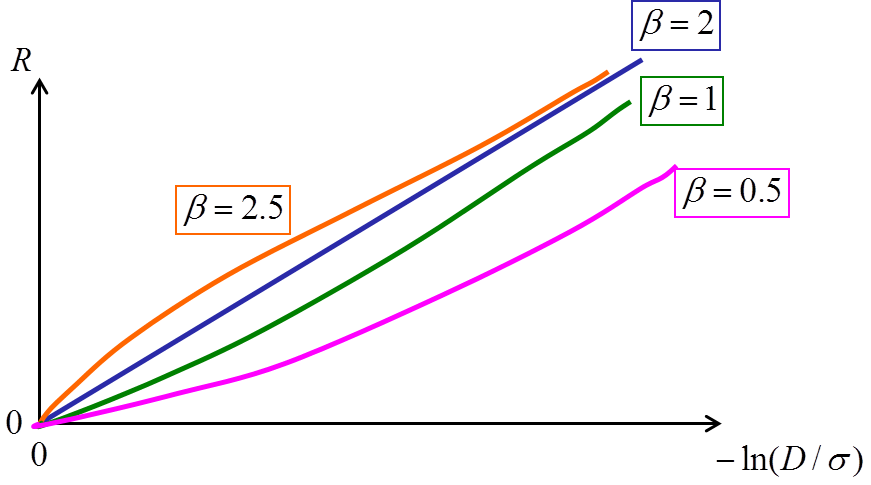


Figure 9: the rate-distortion curves of optimal quantifiers for GGD channels

For instance, a few hundreds of quantifiers are stored for each  up to maximum rate of 6 bits or so.

# Balanced encoding of DCT coefficients of a block type

## Rate-distortion model of optimal quantifiers

Again, let us have the rate-distortion curves on **Figure 9**. One notices that, as expected, the rate decreases monotonously as a function of the distortion induced by the quantifier. So, one can write



where  is the normalization factor of the DCT coefficient, i.e. the GGD model associated to the DCT coefficient has  for standard deviation. In particular, without encoding (equivalently zero rate) leads to a quadratic distortion of value  and we deduce that

.

The monotonicity gives



and finally, one observes that the curves are convex for parameters  lower than two.



## Merit of encoding

Now, let us consider the merit of encoding a DCT coefficient. More encoding basically means

* more rate , in other words “the price to pay”;
* less distortion , in other words “the gain”.

When one wants to dedicate a bit more rate to the encoding of the video, one must determine on which DCT coefficient this “extra rate” is the most efficient. The merit of encoding is the ratio of benefit of distortion on cost of encoding

.

Let us suppose that we reduce the distortion by an amount of , then a first order development of distortion and rates gives



and

.

Then, the ratio of the first variations provides an explicit formula for the merit of encoding.



For  lower than two, the convexity of the rate distortion curves teaches us that the merit is an increasing function of the distortion.



The initial merit is, by definition, the merit of encoding at zero rate, i.e. before any encoding.



## The theorem of equal merit encoding

It can be determined, by using the so-called KKT (Karash Khun Tucker) method, the optimal amount of encoding of DCT coefficient such that a given distortion is reached. We skip the details and we claim that one gets

  .

whereis the set of coded DCT coefficients and is the distortion target. This equation is implicit in the ’s. We rearrange this equation to obtain



From this, we deduce the theorem of equal merits.

**The theorem of equal merits**

*All encoded DCT coefficient of a block type have the same merit after encoding. The common merit is called* ***the merit of the block type****.*

*Furthermore, all non-coded coefficients have a merit bigger than the merit of the block type.*

*Proof*: The formula from the KKT algorithm is a proof itself, but we provide another heuristic proof here.

Let’s suppose that there are two encoded DCT coefficients with two different merits M1 < M2. Now, let’s take an infinitesimal amount of rate from coefficient 1 to put this rate on coefficient 2. This is possible because coefficient 1 is encoded and this does not change the total rate.

Because M1<M2, the distortion gain on coefficient 2 is strictly bigger than the distortion loss on coefficient 1. So we get a better distortion with same rate. We just have found a better R(D) point; this would be in contradiction with optimality.

Conclusion if the two coefficients 1 and 2 are encoded and if M1 < M2, then the solution is not optimal. This completes the proof.

Note that the theorem of equal merits is also known as the Pareto condition in the field of source coding [6].

Practically, the model of DCT coefficients always provide  and the merits are increasing functions of the distortion. As a consequence as shown on **Figure 10**, for a given target block merit , for each DCT coefficient,

* either the initial merit of the coefficient is lower than the target block merit and the coefficient is not encoded
* or there is a unique distortion  such that .

This unique solution is easily found by dichotomy on the distortion .

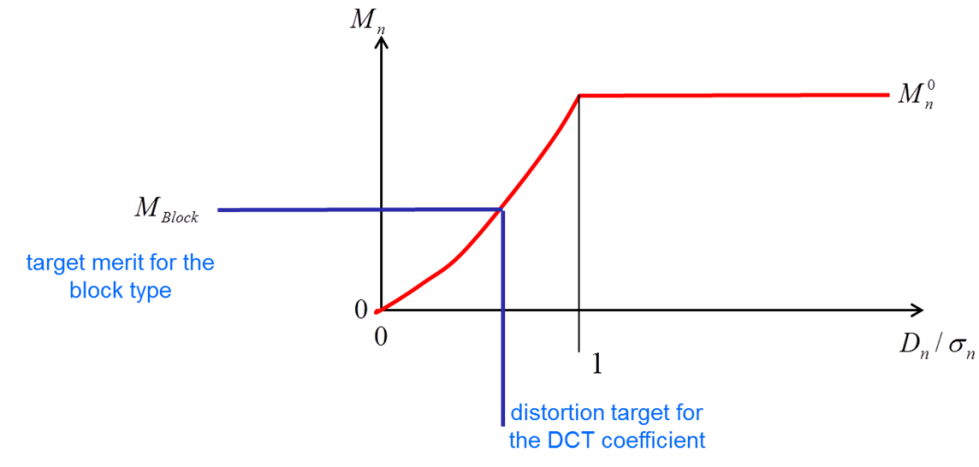


Figure 10: how to find distortions from the target merit for the block type

# Balanced encoding between block types

Due to their different sizes, all block types cannot have the same block merit in an optimal coding. Here we show that there is a very simple relation between block merits and the frame merit.

## A few definitions

Let us start from a few straightforward definitions.

* DCT index number 



* block type 
* distortion of a DCT coefficient 
* rate of a DCT coefficient 
* pixel distortion of a block type 
* rate of a block type 
* merit of encoding of a block type 
* density of a block type in the frame 
* number of blocks per unit of surface 
* rate per unit of surface 
* frame rate 
* frame distortion 

The correspondence between pixel distortion and DCT coefficient distortion becomes simply

,

and the mean rate of encoding for a block type is provided by



The theorem of equal merits states that optimal encoding must fulfil the following relation.



Now, let us care about the effect of the block sizes. First, because the entire frame is covered by blocks, the sum of densities must be unity.



The rate per unit of surface of a block type is just the rate of one block multiplied by the number of blocks per unit of surface.



Then, the mean rate per unit of surface for the whole frame is the sum of surface rates per block type weighted by the densities.



Finally, the mean frame distortion is the distortion per block type weighted by the densities.



## Merit of the frame and merit of the block type

The balance of encoding between block types can be stated very easily. The merit of encoding per pixel, also called **the merit of the frame** , must be the same for all block types, i.e.

.

Clearly, just considering the merit of DCT coefficients does not take the block surface of the blocks into account. From the equal merit property, we get

.

The variation of rate and distortion for a block type is provided by the coefficient variations as follows.





So, we deduce a relation between the variation of pixel distortion and the variation of the rate.



Now, using this equality in the definition of pixel merit, we obtain

.

This proves the very simple relation between the frame merit and the block merit.



# Balanced encoding between frames and colour components

One may think that optimal encoding of a video is obtained by encoding all frames with a constant frame merit. This is not true, for instance in standardization, where the quality of a frame is measured by using the PSNR, which is just the logarithm of the quadratic pixel distortion, and the quality of a video is the mean of frames PSNR values.

So, because the quality of the video is not the mean quadratic pixel distortion of all frames but the mean of logarithms, we must balance the merit between frames.

## Case of the luminance component Y only

Let us write the PSNR as the logarithm of the distortion

.

We dropped the constant because they do not change the final result as the reader can easily verify.

A variation of encoding induces a variation of distortion which induces a variation of PSNR as follows,

,

such that the we get

.

Now, we find the relation between the variation of distortion and the variation of rate. Again, the theorem of equal merits provides



and we find

.

Then, the variation PSNR can be restated as



and we define **the video merit** as the PSNR merit as follows.



The video merit is a constant provided to the encoder. It replaces the parameter QP used in standard codecs. We just have proven the implicit relation between video merit and frame merit.



## Case of the three components YUV

In case of colored video with three components YUV, it is common to judge the compression capability by looking at the total rate, i.e. the rate of Y,U and V, and the PSNR of the luminance Y only.

Thus, the video merit becomes

.

Unfortunately, this relation is not closed: one needs a relation between the luminance distortion and the chrominance distortion. We decide, a priori, to set the quality of UV as the same quality as Y.



One may argue that most of standard codec based on a QP parameter got a much better quality for UV than Y. This is just a side effect of the inability to control the distortion and it is very unlikely to be useful from a video quality point of view.

We also decide to use a unique merit for chrominance.



The correspondence between distortion variation and rate variation is found as above.







Then, one deduces the relation between Y rate variation and frame rate variation.



So, the rightmost term of the YUV video merit expression can be solved as



and one finds an implicit relation between the Y frame merit and the video merit



which can be restated as follows.



## How to find the frame merit from the video merit: graphical method

We will discuss only the most complex case with three components YUV. The case with Y only can be deduced by taking the limit



The merit balance is an implicit relation in ,

,

where the video merit  is a given. Finding the luminance merit is equivalent to finding the rightmost zero of the function

.

Actually, the merit zero is a solution and corresponds to a raw video coding. We are not interested in this solution; that is why we talk of the “rightmost” zero.



Now, one must understand that there are basically three regimes of encoding for a YUV frame. From the least to the most encoding order, we have

* the high merit regime. There is no encoding in the enhancement layer. The UV merit is infinite and the distortion is just equal to the base distortion
* the intermediate merit regime. The component Y is encoded but not the component UV. This often happens because, with standard codec, the PSNR UV of the base layer is much better than the PSNR of Y. The UV merit is infinite but the distortion  decreases with the luminance merit .
* the low merit regime. Both Y and UV are encoded. Here the UV merit is finite and decreases with the luminance merit , as well as the distortion .

The function is not continuous when passing from the intermediate merit regime to the low merit regime, because there is a discontinuity in  which abruptly changes from infinity to a finite value. See **Figure 11**.

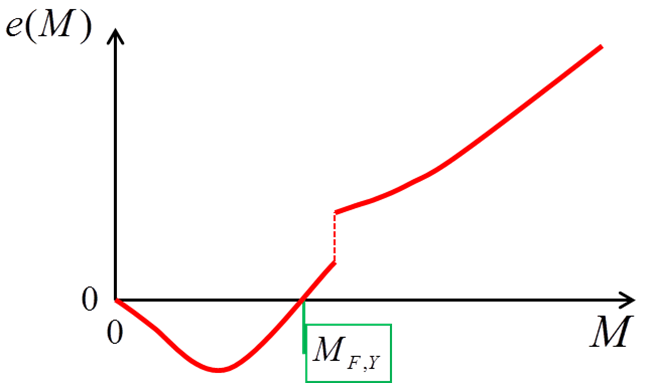


Figure 11: the luminance merit as a zero of the function e

We use a simple dichotomy scheme to find the rightmost zero of the function , see **Figure 12**. This automatically excludes the raw video coding solution by taking a strictly positive lower bound for  at the start of the algorithm. This also allows us to find the “zero” (strictly speaking this not a zero…) when the discontinuity of passes through the horizontal axis.

Basically, the algorithm is as below

1. set initial Mlow and Mhigh
2. take Mmiddle = (Mlow+Mhigh)/2
3. depending on the sign of e(Mmiddle), update bounds Mlow and Mhigh
4. loop to 2. until convergence

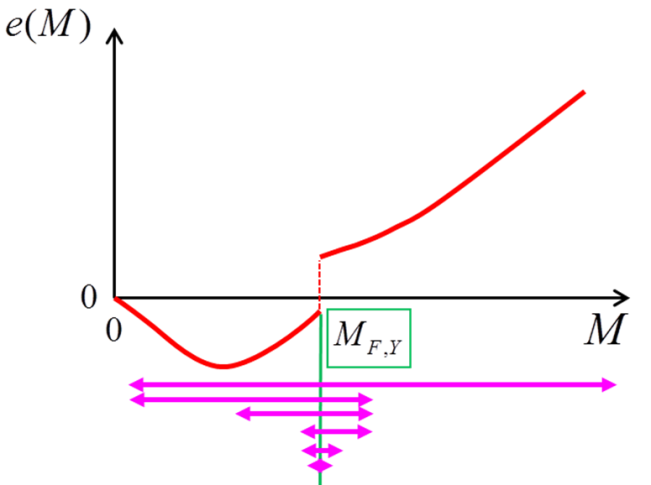


Figure 12: dichotomy scheme to find the luminance merit

## How to find the frame merit from the video merit: the algorithm

The dichotomy algorithm presented above must be detailed a bit more because the evaluation of e(Mmiddle) is not a trivial task. The complete algorithm implemented in the codec is as follows, see also **Figure 13**.

1. set initial Mlow and Mhigh
2. take Mmiddle = (Mlow+Mhigh)/2
   1. compute the luminance merit of block types by using 
   2. determine the distortion of luminance quantifiers by using the equal merit theorem
   3. determine the pixel luminance distortion  as the sum of the DCT distortions
   4. determine the chrominance merit
      * if the chrominance distortion of the base layer is smaller than luminance distortion, then no chrominance coding is needed and we set 
        + otherwise find the chrominance merit  by using a secondary dichotomy scheme depending on the sign of 
   5. determine the value of e(Mmiddle) by using ,  and 
3. depending on the sign of e(Mmiddle), update the two bounds Mlow and Mhigh
4. reiterate from step 2. until convergence or until a predetermined number of iterations is reached

One must understand that this algorithm is not costly because the evaluation of the distortions  and  is performed directly from the distortion of the quantifiers; there is absolutely no need to encode the video to evaluate these two distortions.

The output of the algorithm is the merits of the luminance and the chrominance, as well as the quantifiers for the luminance Y and chrominance UV.

On the decoder side, the two merits and the channel statistics are sufficient to recover the quantifiers by using the equal merit theorem and the balance between block types.

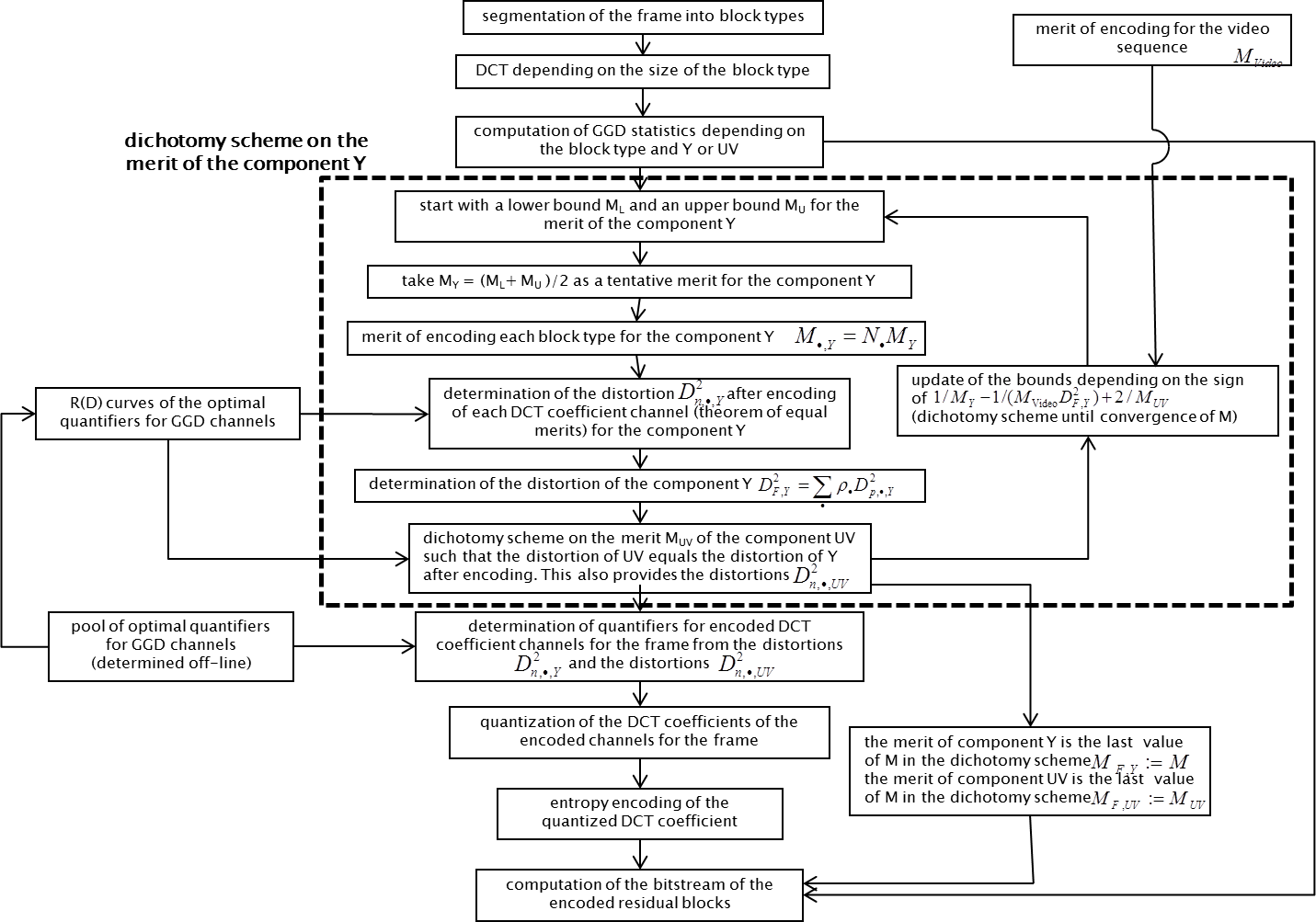


Figure 13: encoding process for a YUV video sequence with several types of blocks

# Choice of quantifiers and quantization

The figure 13 describes the complete encoding process of a residual frame.

An initial segmentation in different block types is computed (section 3), then the DCT of each block adapted to its size is computed. The statistics of each DCT channel is computed and modeled by GGD (section 4)

The target quality (encoding merit of the video equivalent to the classical quantization parameter) is then used with the algorithm of section 9 to compute the distortion target  (deduced from the block merit and the theorem of equal merits) of each DCT coefficient. Then, one chooses the best optimal quantifier associated to this distortion (section 6). For instance one may take, from list of optimal quantifiers corresponding to the associated parameter  of the DCT channel model, the quantifier with least rate among quantifiers having distortion less than the target distortion .

Then, quantization is performed by the chosen quantifiers to obtain the quantized data of transformed coefficients. Practically, these data are just symbols corresponding to the index of the quantum (or interval or Voronoi cell in 1D) in which the value of the coefficient falls in.

Finally, the quantized data are coded by an entropy encoder following the statistical distribution of the DCT channels. For instance, the entropy coding may be performed by any known coding technique like a context-free arithmetic coding.

# Block type competition for segmentation optimization

As noted in section 3, the efficiency of the method described in previous sections, is very sensitive to the initial block type segmentation. The segmentation method described in section 3 is not optimal and can be improved by a competition process between block types.

## Iterative improvement of the segmentation

The principle of the competition is simple: one starts from an initial segmentation (found from the morphology of the residual as already described) into block types and tries to improve it iteratively by evaluating the effect of a change of the belonging to a block for a one block of the residual frame. This evaluation will be called “block type competition” hereafter.

In order to decide which block is the best for the compression of the data, one needs an objective criterion of evaluation; we will use an elaborated formulation of the very standard Lagrangian cost D²+λR used in many codecs.

An overview of the competition process is shown on the block diagram of **Figure 14**. It can be summarized as follows:

1. start from an initial segmentation into block types (based on the content of each block of the frame)
2. apply DCT and compute GGD statistics for each block type
3. determine the merit of the frame, merits of block types, merits of encoding of DCT channels and quantifiers
4. for each block of the frame, perform a Lagrangian cost competition between block types for this block; the cost is computed from
   1. the bit-rate needed for encoding by using the quantifiers of the competing block type
   2. and the distortion after quantization and dequantization by using the quantifiers of the competing block type;

and change the type of the block into the type with the best (smaller cost)

1. loop to 2. until a fixed number of iteration is reached or until the segmentation does not evolve anymore
2. by using the last segmentation computed by the iterative process, apply DCT and compute GGD statistics for each block type
3. determine the merit of the frame, merits of block types, merits of encoding of DCT channels and quantifiers
4. quantize and encode (by using an entropic encoder) the coefficients of the DCT channels

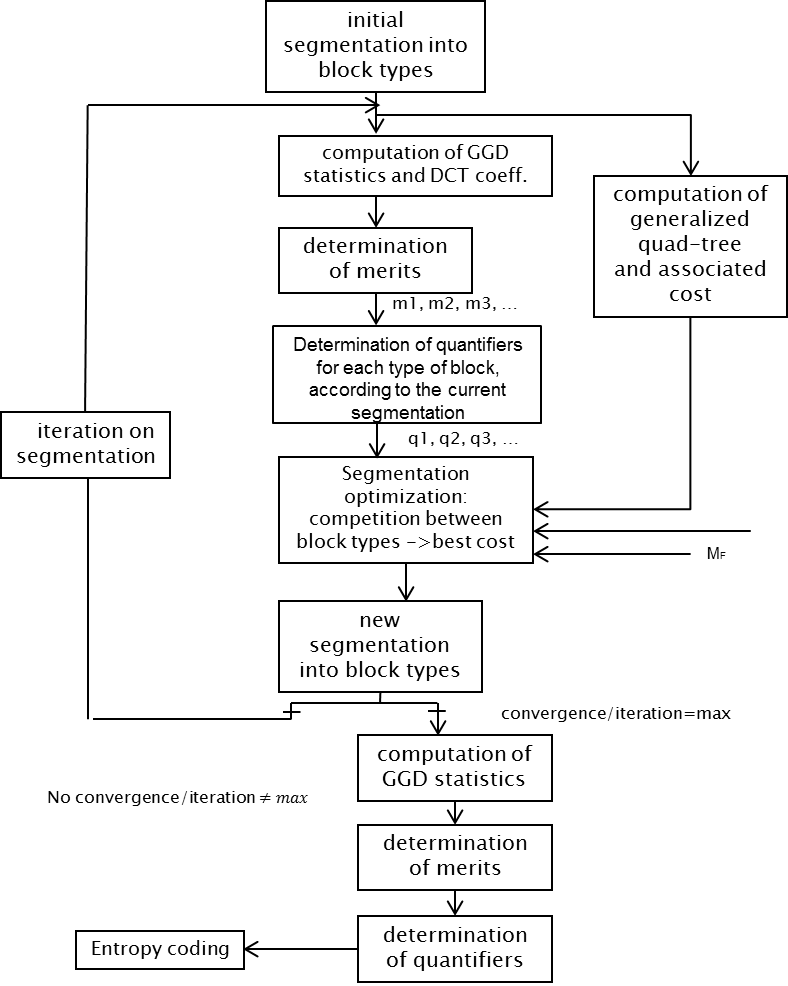


Figure 14: block diagram of the block type competition for segmentation optimization

The computation of GGD statistics needs to be located inside the loop because, after the first iterations, the statistics are not consistent anymore after having performed block type competition. However, after a small number of iterations (typically from 5 to 10), one observes a convergence of the iterative process to a local optimum for the segmentation.

## General formulation of the Lagrangian cost

It turns that, instead of using the formulation D²+λR, it is simpler to use the formulation D²/λ+R when dealing with several color components. From here, we suppose that we encode a frame made of three YUV components and, as described above, luminance and chrominance are encoded separately with their own merits. We also suppose that the block type segmentation is the same for luminance and chrominance, but one can easily generalize to the case where the segmentations do not match; only an extra generalized quad-tree would be needed.

One can put the cost under the very general form



where

*  is the cost of the block type for the selected block
*  is the luminance distortion of the selected block after quantization and dequantization
*  is the chrominance distortion of the selected block after quantization and dequantization
*  is the bit-rate generated by the encoding of the luminance of the selected block
*  is the bit-rate generated by the encoding of the chrominance of the selected block
*  is the bit-rate associated to the parsing of the generalized quad-tree to mark the type the selected block

The two Lagrange’s parameters  and will be deduced exactly from the characteristics of the block types and the merits of encoding.

## Lagrange’s parameter for the luminance

The cost of encoding for the luminance can be written



where the Lagrange’s parameter is given by

.

From the definition of the block merit and the size of the block, we get  and  such that we deduce



and the luminance cost becomes

.

## Lagrange’s parameter for the chrominance

The cost of encoding for the chrominance can be written



where the Lagrange’s parameter is given by

.

Let us recall the quality coupling between luminance and chrominance



as well as the common encoding of U and V

 and .

From the definition of the merit of the chrominance, one gets

 and ,

then by combining these two equalities



Now, by using the common size of the block U and the block V, we find the chrominance block merit



and we finally get a closed form for the Lagrange’s parameter

.

The chrominance cost can be rewritten



under the assumption that there is no quad-tree cost because the chrominance segmentation is inferred by the luminance segmentation.

## Combined luminance + chrominance cost

The combined cost, taking into account both luminance and chrominance, is the sum of the two associated costs

.

However, one must think of the coupling between luminance and chrominance: the merit of chrominance is computed such that the quality (on the whole frame) of the chrominance matches the quality of the luminance. As a consequence, a variation of the luminance distortion  in one block has a global impact on the average distortion of the chrominance on the whole frame. Due to the quality equality, this impact is

.

Classically, this coupling can be put into the combined cost by adding a coupling term as follows

.

## Practical computation of the cost

The distortions and  are computed by applying the quantifiers of the block type, then by applying the associated dequantization and finally by comparing the result with the original residual. This last step can be done in the DCT transform domain because the IDCT is a L2 isometry: total distortion in the DCT domain is the same as the total pixel distortion.

Bit-rates  and  can be evaluated without performing the entropy encoding of the quantized coefficients. Actually, one knows the rate cost of each quantum of the quantifiers; this rate is simply computed from the probability of falling into this quantum and the probability is provided by the GGD channel modelling.

## Effect of the block size on the cost

The size (more precisely the area) of a block impacts the cost formula through the geometrical parameters  and . For instance, let us consider the case of a 16x16-pixel unit area and a 4:2:0 YUV colour format. The cost of a 16x16 block becomes



because  and . This last value comes from the fact that one needs two couples of 4x4 UV blocks to cover a unit area of size 16x16 pixels. Similarly, the cost of a 8x8 block becomes



## Bottom-to-top competition

Let us suppose that we have many types of blocks with only three different sizes: 4x4, 8x8 and 16x16. So, for a 16x16 area to be coded, one has to decide both

* the segmentation of this area into 16x16, 8x8 and 4x4 blocks,
* the choice of the type for each block,

such that the cost is minimized. This may lead to a very big number of possible configurations to evaluate. Fortunately, by using the classical so-called bottom-to-top competition technique, one can dramatically decrease the number of configurations to deal with.

The fundamental tool is the additivity of costs, see **Figure 15**. On this figure (left), we show a 8x8 block segmented into four 4x4x blocks. By using 4x4 cost competition, we determine the best type (=the most competitive type = the type with the smallest cost) for each 4x4 block. Then, the cost associated with the 4x4 segmentation is just the addition of the four underlying best 4x4 costs.

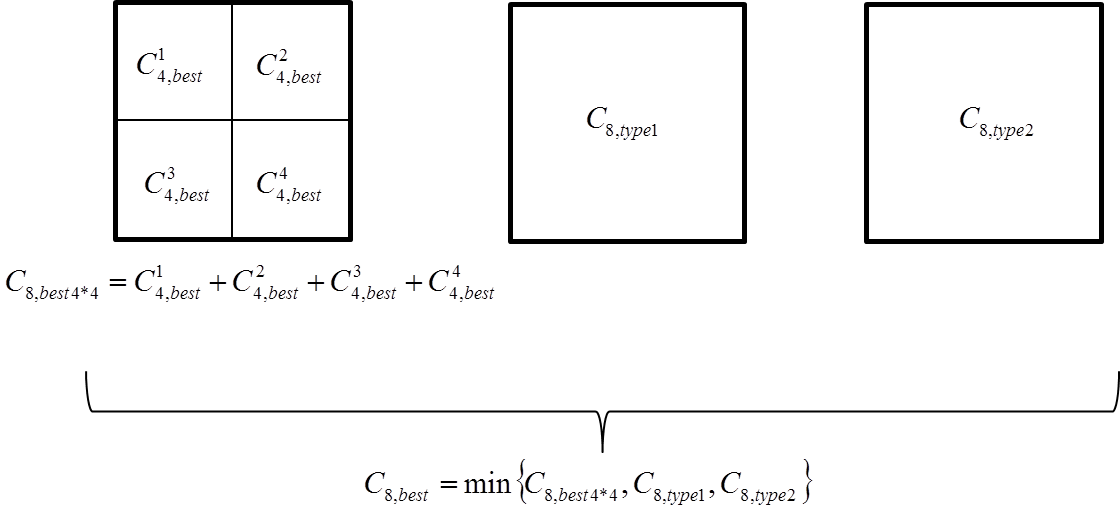


Figure 15: bottom-to-top competition for 4x4 and 8x8 blocks

Now, one starts the bottom-to-top process by comparing this best 4x4 cost for the 8x8 block to the costs of 8x8 block types. On the figure, we supposed that there are two 8x8 block types. So, one has to compare three costs:

* the best 4x4 cost deduced from cost additivity,
* the 8x8 cost from 8x8 type 1,
* the 8x8 cost from 8x8 type 2.

The smallest cost decides the segmentation and the types of the 8x8 block. Then, one can then push the bottom-to-top process at a bigger scale: once the segmentation and types of 8x8 block has been decided, one can use competition to decide the segmentation of 16x16 blocks by performing a competition between the best 8x8 cost (again by using additivity) and 16x16 costs of 16x16 block types.

Of course, one understands easily that the bottom-to-top competition is not limited to three different sizes, not even to square blocks, and can be generalized.

# Conclusion and perspective

As a conclusion to this contribution, **it has been shown that an excellent rate distortion performance can be obtained with a low complexity INTRA scalable coding process**.

The main characteristics of this low complexity INTRA scalable codec are a unique prediction mode and statistical modelling of DCT channel to encode. Low complexity is obtained through the off-line computing of most of heavy processes, which are the design of rate distortion optimal quantifiers, rate distortion models associated to these optimal quantifiers, probabilities used in the arithmetic coder. All the processing involved on the encoder and the decoder can easily be performed in real time.

In addition to low complexity, the proposed scalable codec design provides **spatial random access** feature, as well as a **high degree of parallelism**. Moreover, backward compliance with an **H.264/AVC base layer** is also easily provided, since the scalable coding relies on reconstructed base layer pictures.

The obtained rate distortion level of performance shows that high coding efficiency can be obtained with low complexity. Therefore, we strongly recommend that the JCTVC committee considers low complexity requirements in the design of the scalable extension of HEVC. In particular, test conditions imposing limiting encoder complexity is desired. Because of its limited memory requirements, scalability testing conditions in all intra coding mode should also be considered.

1. **References**

[1] ITU-T Recommendation H.264, “Advanced Video Coding for Generic Audiovisual Services,” March, 2010.

[2] D. Hong, W.Jang, J.Boyce, “Scalability support in HEVC”, JCTVC-F290, Torino, IT, July 2011.

[3] H.M. Choi, J.Nam, D.Sim, “Scalable structures and inter-layer prediction for HEVC scalable extension”, JCTVC-F096, Torino, IT, July 2011.

[4] P. Chou, T. Lookabaugh, R. Gray, Entropy Constrained Vector Quantization, IEEE Trans. on acoustics, speech and signal processing, September, vol. 37(1), January 1989.

[5] S.Lasserre, F. Le Léannec, E. Nassor, "Low Complexity Scalable Extension of HEVC intra pictures", JCTVC-G248, November 2011

[6] T.Wiegand, H.Schwarz, “Source Coding: Part I of Fundamentals of Source and Video Coding”

[7] <http://en.wikipedia.org/wiki/Mathematical_morphology>.

[8] “Image Analysis and Mathematical Morphology”, Vol. 1, Jean Serra

# Patent rights declaration(s)

**Canon Research Centre France may have current or pending patent rights relating to the technology described in this contribution and, conditioned on reciprocity, is prepared to grant licenses under reasonable and non-discriminatory terms as necessary for implementation of the resulting ITU-T Recommendation | ISO/IEC International Standard (per box 2 of the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form).**