(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
6 September 2013 (06.09.2013)

WIPO | PCT

(10) International Publication Number
WO 2013/128010 A2

(51) International Patent Classification:
H04N 7/26 (2006.01)

(21) International Application Number:
PCT/EP2013/054198

(22) International Filing Date:
1 March 2013 (01.03.2013)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

1203706.5	2 March 2012 (02.03.2012)	GB
1206527.2	13 April 2012 (13.04.2012)	GB
1215430.8	30 August 2012 (30.08.2012)	GB
1217464.5	28 September 2012 (28.09.2012)	GB
1217554.3	1 October 2012 (01.10.2012)	GB
1223385.4	24 December 2012 (24.12.2012)	GB

(71) Applicant: **CANON KABUSHIKI KAISHA** [JP/JP]; 3-30-2, Shimomaruko, 3-chome, Ohta-ku, Tokyo (JP).

(72) Inventors; and

(71) Applicants *(for US only)*: **LE LEANNEC, Fabrice** [FR/FR]; 4, rue des Magnolias, F-35250 Mouaze (FR). **LASSERRE, Sébastien** [FR/FR]; 11 bd Saint Conwoïon, F-35000 Rennes (FR). **OUEDRAOGO, Naël** [FR/FR]; Coutouse, F-35330 Maure De Bretagne (FR). **LAROCHE, Guillaume** [FR/FR]; Le Haut Launay, F-35520 Melesse (FR).

(74) Agent: **SANTARELLI**; PO BOX 237, 14, avenue de la Grande-Armée, F-75822 Paris Cedex 17 (FR).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *without international search report and to be republished upon receipt of that report (Rule 48.2(g))*

(54) **Title**: METHOD AND DEVICES FOR ENCODING A SEQUENCE OF IMAGES INTO A SCALABLE VIDEO BIT-STREAM, AND DECODING A CORRESPONDING SCALABLE VIDEO BIT-STREAM

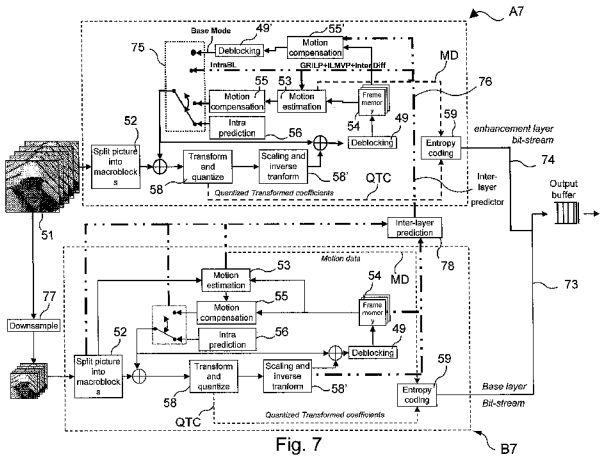


Fig. 7

(57) **Abstract**: The invention relates to the field of scalable video coding, in particular to scalable video coding that would extend the High Efficiency Video Coding (HEVC) standard. An encoding method comprises encoding a base layer and an enhancement layer, including encoding an enhancement original INTRA image using intra-frame prediction only by: obtaining a residual image as a difference between the enhancement original INTRA image and a decoded corresponding encoded base image in the base layer, the residual image comprising blocks of pixels, each having a block type; transforming pixel values for a block into a set of coefficients each having a coefficient type, said block having a given block type; determining an initial coefficient encoding merit for each coefficient type; selecting coefficients based, for each coefficient, on the corresponding initial coefficient encoding merit and on a predetermined block merit; quantizing the selected coefficients into quantized symbols; and encoding the quantized symbols.

WO 2013/128010 A2

METHOD AND DEVICES FOR ENCODING A SEQUENCE OF IMAGES INTO A SCALABLE VIDEO BIT-STREAM, AND DECODING A CORRESPONDING SCALABLE VIDEO BIT-STREAM

FIELD OF THE INVENTION

The invention relates to the field of scalable video coding, in particular to scalable video coding that would extend the High Efficiency Video Coding (HEVC) standard. The invention concerns methods, device and computer-readable medium storing a program for encoding and decoding digital video sequences made of images (or frames) into scalable video bit-streams.

BACKGROUND OF THE INVENTION

Video coding is a way of transforming a series of video images into a compact digitized bit-stream so that the video images can be transmitted or stored. An encoding device is used to code the video images, with an associated decoding device being available to read the bit-stream and reconstruct the video images for display and viewing. A general aim is to form the bit-stream so as to be of smaller size than the original video information. This advantageously reduces the capacity required of a transfer network, or storage device, to transmit or store the bit-stream code.

Common standardized approaches have been adopted for the format and method of the coding process, especially with respect to the decoding part. One of the more recent agreements is Scalable Video Coding (SVC) wherein the video image is split into smaller sections (called macroblocks or blocks) and treated as being comprised of hierarchical layers. The hierarchical layers include a base layer, equivalent to a collection of images (or frames) of the original video image sequence, and one or more enhancement layers (also known as refinement layers) also equivalent to a collection of images (or frames) of the original video image sequence. SVC is the scalable extension of the H.264/AVC video compression standard.

A further video standard being standardized is HEVC (standing for High Efficiency Video Coding), wherein the macroblocks are replaced by so-called Coding Units and are partitioned and adjusted in size according to the characteristics of the original image sequence under consideration. This allows more detailed coding of areas of the video image which contain relatively more information and less coding effort for those areas with fewer features.

The video images were originally processed by coding each macroblock individually, in a manner resembling the digital coding of still images or pictures. Later coding models allow for prediction of the features in one frame, either from neighbouring macroblocks, or by association with a similar macroblock in a neighbouring frame. This allows use of already available coded information by exploiting the spatial and temporal redundancies of the images in order to shorten the amount of coding bit-rate needed overall. These powerful video compression tools, known as spatial (or intra) and temporal (or inter) predictions, make the transmission and/or the storage of video sequences more efficient.

Differences between the source area and the area used for prediction are captured in a residual set of values which themselves are encoded in association with the code for the source area. Many different types of predictions are possible. Effective coding chooses the best model to provide image quality upon decoding, while taking account of the bit-stream size each model requires to represent an image in the bit-stream. A trade-off between the decoded picture quality and reduction in required code, also known as compression of the data, is the overall goal.

A context of the invention is the design of the scalable extension of HEVC. HEVC scalable extension will allow coding/decoding a video made of multiple scalability layers.

These layers comprise a base layer that is often compliant with standards such as HEVC, H.264/AVC or MPEG2, and one or more enhancement layers, coded according to the future scalable extension of HEVC. The teachings of the invention as described below with reference to an enhancement layer, for example the Intra-frame coding, may however be applied to the base layer.

It is known that to ensure good scalable compression efficiency, one has to exploit information coming from a lower layer, in particular from the base layer, when encoding an upper enhancement layer. For example, SVC standard already proposes to exploit redundancy that lies between the base layer and the enhancement layer, through so-called inter-layer prediction techniques. In SVC, a block in the enhancement layer may be predicted from the spatially corresponding (i.e. co-located) block in the decoded base layer. This is known as the Intra Base Layer (BL) prediction mode.

In case of Intra frames, i.e. frames to be coded using only spatial prediction to be self sufficient for decoding, known coding mechanisms for encoding the residual image are not fully satisfactory.

In case of Inter frames, i.e. frames coded using the Inter or temporal prediction, one may selectively predict successive blocks of the image through intra-layer Inter prediction, intra-layer spatial Intra prediction, inter-layer Inter prediction and inter-layer Intra prediction. In classical scalable video codecs (encoder-decoder pairs), this takes the form of block prediction choice, one block after another, among the above mentioned available prediction modes, according to a rate distortion criteria. Each reconstructed block serves as a reference to predict subsequent blocks. Differences are noted and encoded as residuals. Competition between the various possible encoding mechanisms takes account of both the type of encoding used and the size of the bit-stream resulting from each type. A balance is achieved between the two considerations. Known mechanisms for Inter-frame coding using Inter-layer prediction are not fully satisfactory.

SUMMARY OF THE INVENTION

The present invention has been devised to address at least one of the foregoing concerns, in particular to improve Intra-frame coding or Inter-frame coding or both for scalable videos.

A method according to the invention for encoding a video sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, may comprise:

encoding a base layer made of base images;

encoding an enhancement layer made of enhancement images, including encoding at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame prediction only; and encoding at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame prediction;

According to a first aspect, the invention provides the above encoding method wherein encoding the enhancement original INTRA image comprises the steps of:

obtaining a residual enhancement image as a difference between the enhancement original INTRA image and a decoded version of the corresponding encoded base image in the base layer, the residual enhancement image comprising a plurality of blocks of pixels (in fact residual information corresponding to each original pixel), each block having a block type;

transforming pixel values for a block among said plurality of blocks into a set of coefficients each having a coefficient type, said block having a given block type;

determining an initial coefficient encoding merit for each coefficient type;

selecting coefficients based, for each coefficient, on the initial coefficient encoding merit for said coefficient type and on a predetermined block merit;

quantizing the selected coefficients into quantized symbols;

encoding the quantized symbols.

Thus, thanks to the use of the measure provided by the initial coefficient encoding merit, the selection of coefficients (e.g. DCT coefficients within the block) to be actually encoded is made simpler.

Optional features of the encoding method or of the encoding device are defined in the appended claims.

In one embodiment, a coefficient type is selected if the initial encoding merit for this coefficient type is greater than the predetermined block merit.

In one embodiment, the method comprised a prior step of determining the predetermined block merit based on a predetermined frame merit and on a number of blocks of the given block type per area unit.

In particular determining the predetermined frame merit comprises determining a frame merit and a distortion at the image level using a balancing parameter such that a video merit, computed based on said distortion and said frame merit, corresponds to a target video merit.

In particular, the step of determining a frame merit and a distortion at the image level is such that a product of the determined distortion at the image level and of the target video merit essentially equals a product of the balancing parameter and the determined frame merit.

In one embodiment the enhancement original INTRA image is a luminance image, the enhancement layer comprises at least one corresponding colour image comprising a plurality of colour blocks; and the method comprises steps of :

determining a colour frame merit;

determining, for each colour block of said plurality of colour blocks, a colour block merit for the concerned colour block based on the colour frame merit;

transforming, for each colour block of the plurality of blocks, pixel values for the concerned colour block into a set of coefficients each having a coefficient type;

selecting coefficient types based, for each coefficient, on an initial encoding merit for said coefficient type and on the colour block merit for the concerned colour block;

for each block of said plurality of colour blocks, selecting, for each selected coefficient type, a quantizer based on the colour block merit for the concerned colour block;

for each selected coefficient type, encoding coefficients having the concerned type using the selected quantizer for the concerned coefficient type.

In particular, determining the colour frame merit uses a balancing parameter.

In particular, determining the predetermined frame merit comprises determining a frame merit and a distortion at the image level such that a product of the determined distortion at the image level and of a target video merit essentially equals the determined frame merit and the step of determining the colour frame merit is such that a product of a corresponding distortion for the colour frame and of the target video merit essentially equals a product of the balancing parameter and the determined colour frame merit.

In one embodiment ,determining an initial coefficient encoding merit for a given coefficient type includes estimating a ratio between a distortion variation provided by encoding a coefficient having the given type and a rate increase resulting from encoding said coefficient.

In one embodiment, encoding the enhancement original INTRA image comprises the following steps:

determining, for each coefficient type and each block type, at least one parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type in the concerned block type;

determining the initial coefficient encoding merit for given coefficient type and block type based on the parameter for the given coefficient type and block type.

In particular, encoding the enhancement original INTRA image comprises, for each coefficient for which the initial coefficient encoding merit is greater than the predetermined block merit, selecting a quantizer depending on the parameter for the concerned coefficient type and block type and on the predetermined block merit.

Also, it may be provided that a parameter obtained for a previous enhancement INTRA image and representative of a probabilistic distribution of coefficients having the concerned coefficient type in the concerned block type in the previous enhancement INTRA image is reused as the at least one parameter representative of a probabilistic distribution of coefficients having the concerned

coefficient type in the concerned block type in the enhancement original INTRA image being encoded.

In a particular embodiment, the coefficient types respectively associated with the encoded selected coefficients form a first group of coefficient types; and

the method further comprises:

transmitting the encoded selected coefficients and parameters associated with coefficient types of the first group;

transforming pixel values for at least one block in a second enhancement original INTRA image of the enhancement layer into a set of second-image coefficients each having a coefficient type;

encoding only a subset of the set of second-image coefficients for said block in the second enhancement original INTRA image, wherein the coefficient types respectively associated with the encoded second-image coefficients form a second group of coefficient types;

transmitting the encoded second-image coefficients and parameters associated with coefficient types of the second group not included in the first group.

In particular, at least one parameter representative of the probabilistic distribution includes the standard deviation of the probabilistic distribution; and

the method further comprises the following steps:

for each coefficient type, computing a standard deviation for the probabilistic distribution of coefficients having the concerned coefficient type in the enhancement original INTRA image;

determining a number of bits necessary for representing the ratio between the maximum standard deviation, among the computed standard deviations associated with a coefficient type of the first group, and a predetermined value;

for each coefficient type of the first group, transmitting a word having a length equal to the determined number of bits and representing the standard deviation associated with the concerned coefficient type.

According to a feature, the parameters associated with coefficient types of the first group are transmitted in a first transport unit and wherein the parameters associated with coefficient types of the second group not in the first group are transmitted in a second transport unit, distinct from the first transport unit.

In one particular embodiment, the encoded first-image coefficients are transmitted in the first transport unit and wherein the encoded second-image coefficients are transmitted in the second transport unit.

In one particular embodiment, the first and second transport units are parameter transport units.

In one particular embodiment, the first transport unit carries a predetermined identifier and wherein the second transport unit carries said predetermined identifier.

According to another feature, the method comprises a step of estimating a proximity criterion between the enhancement original INTRA image being encoded and a third enhancement original INTRA image included in the enhancement layer,

the method further comprising the following steps if the proximity criterion is fulfilled:

transforming pixel values for at least one block in the third enhancement original INTRA image into a set of third-image coefficients each having a coefficient type;

encoding third-image coefficients for said block in the third enhancement original INTRA image, wherein the coefficient types respectively associated with the encoded third-image coefficients form a third group of coefficient types;

transmitting the encoded third-image coefficients, parameters associated with coefficient types of the third group not included in the first and second groups and a flag indicating previously received parameters are valid.

According to yet another feature, the method comprises the following steps if the proximity criterion is not fulfilled:

for each of a plurality of blocks in the third enhancement original INTRA image, transforming pixel values for the concerned block into a set of third-image coefficients each having a coefficient type;

for each coefficient type, computing at least one parameter representative of a probabilistic distribution of the third-image coefficients having said coefficient type;

for at least one block in the third enhancement original INTRA image, encoding third-image coefficients for said block;

transmitting the encoded third-image coefficients, parameters associated with coefficient types of transmitted third-image coefficients and a flag indicating previously received parameters are no longer valid.

According to yet another feature, estimating the proximity criterion includes estimating a difference between a distortion relating to the first enhancement original INTRA image and a distortion relating to the third enhancement original INTRA image.

According to a second aspect, the invention provides the above encoding method wherein encoding the enhancement original INTRA image comprising the steps of:

obtaining a residual enhancement image as a difference between the enhancement original INTRA image and a decoded version of the corresponding encoded base image in the base layer, the residual enhancement image comprising a plurality of blocks of pixels, each block having a block type;

performing an initial segmentation of the residual enhancement image into a set of initial blocks, thus determining, for each initial block, a block type associated with the concerned initial block;

determining, for each block type, an associated set of quantizers based on data corresponding to pixels of blocks having said block type;

selecting, among a plurality of possible segmentations defining an association between each block of this segmentation and an associated block type, the segmentation which minimizes an encoding cost estimated based on a measure of the rate necessary for encoding each block using the set of quantizers associated with the block type of the encoded block according to the concerned segmentation.

According to an embodiment, the encoding cost is computed using a predetermined frame merit and a number of blocks per area unit for the concerned block type.

According to an embodiment, the measure of the rate is computed based on the set of quantizers associated with the concerned block type and on parameters representative of probabilistic distributions of transformed coefficients of blocks having the concerned block type.

According to an embodiment, the encoding cost includes a cost for luminance, taking into account luminance distortion generated by encoding and decoding a luminance block using the set of quantizers associated with the concerned block type, and a cost for chrominance, taking into account chrominance distortion generated by encoding and decoding a chrominance block using the set of quantizers associated with the concerned block type.

According to an embodiment, the initial segmentation into blocks is based on block activity along several spatial orientations.

According to an embodiment, the selected segmentation is represented as a quad-tree having a plurality of levels, each associated with a block size, and leaves associated with blocks and having a value indicating either a label for the concerned block or a subdivision of the concerned block.

In particular, encoding the enhancement original INTRA image comprises a step of compressing the quad tree using an arithmetic entropy coding that uses, when coding the segmentation relating to a given block, conditional probabilities for the various possible leaf values depending on a state of a block in the base layer co-located with said given block.

According to an embodiment, the method comprises:

down-sampling video data having a first resolution to generate video data having a second resolution lower than said first resolution, and encoding the second resolution video data to obtain video data of the base layer having said second resolution;

decoding the base layer video data, up-sampling the decoded base layer video data to generate decoded video data having said first resolution, forming a difference between the generated decoded video data having said first resolution and said received video data having said first resolution to generate residual data,

compressing the residual data to generate video data of the enhancement layer, including determining an image segmentation into blocks for the enhancement layer, wherein the segmentation is represented as a quad-tree having a plurality of levels, each associated with a block size, and leaves associated with blocks and having a value indicating either a label for the concerned block or a subdivision of the concerned block;

arithmetic entropy coding the quad-tree using, when coding the segmentation relating to a given block, conditional probabilities for the various possible leaf values depending on a state of a block in the base layer co-located with said given block.

According to an embodiment, the method comprises:

decoding the encoded coefficients of the enhancement original INTRA image and decoding the corresponding encoded base image in the base layer, to obtain a rough decoded image corresponding to an original image of the sequence;

processing the rough decoded image through at least one adaptive post-filter adjustable depending on a parameter, wherein said parameter is derived based on pixel values and input to the adaptive post-filter.

In particular, the method may comprise determining the parameter input to the adaptive post-filter based on the predetermined frame merit.

According to a third aspect, the invention provides the above encoding method wherein encoding the enhancement original INTER image comprises the steps of:

selecting a prediction mode, from among a plurality of prediction modes, for predicting an enhancement block of the enhancement original INTER image, wherein the plurality of prediction modes includes at least one of:

a base mode prediction mode involving computation, from the base layer, of a base mode prediction image corresponding to the enhancement original INTER image, the base mode prediction image being composed of base mode blocks obtained using prediction information derived from prediction information of the base layer and wherein the enhancement block is derived from one or more base mode blocks of a spatially corresponding region of the base mode prediction image; and

a GRILP prediction mode including: obtaining a block predictor candidate for predicting the enhancement block within the enhancement original INTER image and an associated enhancement-layer residual block corresponding to said prediction; determining a block predictor in the base layer co-located with the determined block predictor candidate within the enhancement original INTER image; determining a base-layer residual block associated with the enhancement block in the base layer that is co-located with the enhancement block in the enhancement original INTER image, as the difference between the co-located enhancement block in the enhancement original INTER image and the determined block predictor in the base layer; determining, for the enhancement block of the enhancement original INTER image, a further residual block corresponding, at least partly, to the difference between the enhancement-layer residual block and the base-layer residual block;

obtaining a prediction block from the selected prediction mode and subtracting the prediction block from the enhancement block of the enhancement original INTER image to obtain a residual block;

transforming pixels values of the residual block to obtain transformed coefficients;

quantizing at least one of the transformed coefficients to obtain quantized symbols;

encoding the quantized symbols into encoded data.

According to an embodiment the plurality of prediction modes includes an inter difference mode in addition or in replacement of the GRILP mode including: performing a motion estimation on a current block of a current Enhancement Layer (EL) image to obtain a motion vector designating a reference block in a EL reference image; computing a difference image between the EL reference image and an upsampled version of the image of the base layer temporally co-located with the EL reference image; performing a motion compensation on the block of the obtained difference image pointed by the motion vector determined during the motion estimation step to obtain a residual block; adding the obtained residual block to the reference block to obtain a final block predictor used to predict the current block.

According to an embodiment, the plurality of prediction modes includes the following prediction modes:

a motion compensated temporal prediction mode within the enhancement layer;

an intra base layer prediction mode where the prediction block is taken from a base block co-located with the enhancement block in an up-sampled decoded version of the corresponding base image;

the base mode prediction mode, wherein each base mode block of the base mode prediction image derives from the co-located base block in the corresponding base image when the co-located base block is intra coded or derives, when the co-located base block is inter coded into a base residual using prediction information, from the block in the enhancement layer that is obtained by applying an up-sampled version of a motion vector of the prediction information onto the base mode block and from an up-sampled decoded version of the base residual;

the GRILP prediction mode and/or the inter difference prediction mode; and

a difference INTRA coding mode.

According to an embodiment, in the GRILP prediction mode, when an image of residual data used for the encoding of the base layer is available, determining the base-layer residual block in the base layer comprises:

determining the overlap in the image of residual data between the determined block predictor and the block predictor used in the encoding of the block co-located with the enhancement block in the base layer; and

using the part in the image of residual data corresponding to this overlap if any to compute a part of said further residual block of the enhancement original INTER image, wherein the samples of said further residual block of the enhancement original

INTER image corresponding to this overlap each corresponds to a difference between a sample of the enhancement-layer residual block and a corresponding sample of the base-layer residual block.

According to an embodiment, in the GRILP prediction mode, the determination of a predictor of the enhancement block is made using a cost function adapted to take into account the prediction of the enhancement-layer residual block to determine a rate distortion cost.

According to an embodiment, the method comprises de-blocking filtering the base mode prediction image before it is used to provide prediction blocks.

In particular, the de-blocking filtering is applied to the boundaries of the base mode blocks of the base mode prediction image.

According to a feature, the method further comprises deriving the organisation of transform units of base blocks in the base layer towards the enhancement layer wherein the de-blocking filtering is applied to the boundaries of the transform units derived from the base layer.

According to an embodiment, the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained; and

encoding the enhancement original INTER image further comprises encoding the motion information using a motion information predictor taken from a set including motion information, if any, associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

In particular, other motion information of the set is derived from the motion information by adding respective spatial offsets.

According to a feature, the set includes motion information, if any, associated with blocks spatially neighbouring the enhancement block in the enhancement original INTER image, and if no motion information exist for a given neighbouring block, motion information, if any, associated with the base block spatially corresponding to the given neighbouring block in the corresponding base image.

According to an embodiment, the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained;

encoding the enhancement original INTER image further comprises encoding the motion information using a motion information predictor taken from a set including more motion information predictors than another set usable for predicting motion information associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

In particular, the set of motion information predictors for the enhancement original INTER image includes at least one motion information predictor generated based on motion information from the base layer.

According to an embodiment, the set of vector predictors comprises at least one temporal motion information predictor and at least one spatial motion information predictor, the at least one temporal motion information predictor being positioned before the at least one spatial motion information predictor.

According to an embodiment, in case of spatial scalability between the base layer and the enhancement layer with the base layer having a lower spatial resolution than the enhancement layer; and

prediction information, such as a motion vector, is derived or up-sampled for a processing block of size 2Nx2N in the enhancement original INTER image from the base layer of lower spatial resolution, the derivation or up-sampling comprises:

determining whether or not the region of the base layer, spatially corresponding to the processing block, is wholly located within one elementary prediction unit of the base layer; and

in the case where the region of the base layer spatially corresponding to the processing block is fully located within one elementary prediction unit of the base layer, deriving prediction information for that processing block from the base layer prediction information of the said one elementary prediction unit;

otherwise in the case where the region of the base layer spatially corresponding to the processing block overlaps, at least partially, each of a plurality of elementary prediction units,

dividing the processing block into a plurality of sub-processing blocks , each of size NxN such that the region of the base layer spatially corresponding to each sub-processing block is wholly located within one elementary prediction unit of the base layer; and

deriving the prediction information for each sub-processing block from the base layer prediction information of the spatially corresponding elementary prediction unit.

In one particular embodiment, in the case where the corresponding elementary prediction unit of the base layer is Intra-coded then the processing block is predicted from the elementary prediction unit reconstructed and resampled to the enhancement layer resolution.

In one particular embodiment, in the case where the corresponding elementary prediction unit is Inter-coded then the processing block is temporally predicted using motion information derived from the said corresponding elementary prediction unit of the base layer.

In one particular embodiment, the processing block is temporally predicted further using a decoded temporal residual of the corresponding elementary prediction unit of the base layer, said temporal residual being computed between base layer images, as a function of the motion information of the elementary prediction unit.

In one particular embodiment, the spatial scaling between an image of the enhancement layer and a corresponding image of the base layer is a non-integer ratio. In particular, the non-integer ratio is 1.5.

In one embodiment, the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the method comprises:

obtaining a first set of quantization offsets to be applied to a group of enhancement images, a different quantization offset being obtained for at least two enhancement INTER images belonging to the same temporal depth; and

determining a quantization parameter used to encode the enhancement INTER images based on the obtained first set of quantization parameters.

In one embodiment, the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the method comprises:

obtaining a first set of quantization offsets to be applied to a group of enhancement images, the same quantization offset being obtained for at least two enhancement INTER images belonging to different temporal depths; and

determining a quantization parameter used to encode the enhancement INTER images based on the obtained first set of quantization parameters.

In one particular embodiment, among the at least two enhancement INTER images that belong to the same temporal depth, a first offset is obtained for a first enhancement INTER image having a reference image of a first quality and a second

offset, larger than said first offset, is obtained for a second enhancement INTER image having a reference image of a second quality lower than said first quality.

In one particular embodiment, the quantization offset obtained for an enhancement INTER image takes into account:

the number of enhancement images using the enhancement INTER image as a reference image for temporal prediction;

the temporal distance of the enhancement INTER image to its reference image having the lowest quantization offset; and

the value of the quantization parameter applied to this reference image.

In one particular embodiment, the quantization offset obtained for an enhancement INTER image is equal or larger than the quantization offset of its reference image having the lowest quantization offset.

In one particular embodiment, the method further comprises encoding, using a second set of different quantization offsets, a group of base images in the base layer that temporally coincides with the group of enhancement images ;

wherein the second set is obtained based on a temporal depth each base image belongs to.In one embodiment, encoding data representing the enhancement original INTER image further comprises encoding, in the bit-stream, quad-trees representing a segmentation of respective portions of the enhancement original INTER image and coding modes associated with blocks in the segmentation;

wherein the coding mode associated with a given block is encoded through

a first coding mode syntax element that indicates whether the coding mode associated with the given block is based on temporal/Inter prediction or not,

a second coding mode syntax element that indicates whether a prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information is used or not for encoding the block if the first coding mode syntax element refers to temporal/Inter prediction, or indicates whether the coding mode associated with the given block is a conventional Intra prediction or based on Inter-layer prediction if the first coding mode syntax element refers to non temporal/Inter prediction, and

in the case in which the coding mode associated with the given block is based on Inter-layer prediction, a third coding mode syntax element that

indicates whether the coding mode associated with the given block is the intra base layer mode or the base mode prediction mode.

In a embodiment the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the GRILP prediction sub-mode.

In an embodiment the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the inter difference prediction sub-mode.

In an embodiment a fourth coding mode syntax element indicates whether the inter difference block is used or not for encoding the block if the second coding mode syntax element indicates that the GRILP mode is not used or whether the GRILP mode is used or not for encoding the block if the second coding mode syntax element indicates that the inter difference mode is not used.

In an embodiment at least one high level syntax element indicates which of the coding mode syntax elements are present or not in the bit-stream.

In an embodiment when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding mode syntax element following the removed coding mode syntax element replaces the removed coding mode syntax element and the coding order of the remaining coding mode syntax elements is kept.

In an embodiment when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding order of the remaining coding mode syntax elements is modified.

In an embodiment the modification of the coding order of the remaining coding mode syntax elements takes into account the probability of occurrence of coding modes represented by remaining coding mode syntax elements.

In one embodiment,

encoding the enhancement original INTRA image comprises selecting quantizers from the predetermined block merit to quantize the selected coefficients, the predetermined block merit deriving from a frame merit;

encoding the enhancement original INTER image comprises selecting quantizers from a quantization parameter to quantize the transformed coefficients; and

the frame merit and the quantization parameter are computed from a user-specified quality parameter and are linked together with a balancing parameter.

In one embodiment, the method is implemented by a computer, and data, such as image samples, in the base layer are processed using 8-bit words and data, such as image samples, in the enhancement layer are processed using 10-bit words.

In one particular embodiment, data, such as image samples, from the base layer are up-scaled to 10-bit words by multiplying each value by 4, when this data from the base layer is processed for the encoding of the enhancement layer.

A method according to the invention for decoding a scalable video bit-stream, may comprise:

decoding a base layer made of base images;

decoding an enhancement layer made of enhancement images, including decoding data representing at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame decoding; and decoding data representing at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame decoding.

According to a fourth aspect, the invention provides the above decoding method wherein decoding data representing at least one block of pixels in the enhancement original INTRA image, comprises the steps of:

receiving said data and parameters each representative of a probabilistic distribution of a coefficient type;

decoding said data into symbols;

selecting coefficient types for which a coefficient encoding merit prior to encoding, estimated based on the parameter associated with the concerned coefficient type, is greater than a predetermined block merit;

for selected coefficient types, dequantizing symbols into dequantized coefficients having a coefficient type among the selected coefficient types;

transforming dequantized coefficients into pixel values in the spatial domain for said block.

In one embodiment, the method comprises a prior step of determining the predetermined block merit based on a predetermined frame merit and on a number of blocks of a block type of the block per area unit.

In one particular embodiment, determining the predetermined frame merit comprises determining a frame merit and a distortion at the image level using a

balancing parameter such that a video merit, computed based on said distortion and said frame merit, corresponds to a target video merit.

In one particular embodiment, the step of determining a frame merit and a distortion at the image level is such that a product of the determined distortion at the image level and of the target video merit essentially equals a product of the balancing parameter and the determined frame merit.

In one embodiment, the predetermined frame merit is decoded from the bit-stream.

In one embodiment, the enhancement original INTRA image is a luminance image, the enhancement layer comprises at least one corresponding colour image comprising a plurality of colour blocks; and the method comprises steps of :

determining a colour frame merit;

decoding data associated with a colour block among said plurality of colour blocks into a set of symbols each corresponding to a coefficient type, said block having a particular block type;

determining a colour block merit based on the colour frame merit and on a number of blocks of the particular block type per area unit;

selecting coefficient types based, for each coefficient type, on a coefficient encoding merit prior to encoding, for said coefficient type, and on the colour block merit;

for selected coefficient types, dequantizing symbols into dequantized coefficients having a coefficient type among the selected coefficient types;

transforming dequantized coefficients into pixel values in the spatial domain for said colour block.

In one particular embodiment, determining the colour frame merit uses a balancing parameter.

In one particular embodiment, determining the predetermined frame merit comprises determining a frame merit and a distortion at the image level such that a product of the determined distortion at the image level and of a target video merit essentially equals the determined frame merit and the step of determining the colour frame merit is such that a product of a corresponding distortion for the colour frame and of the target video merit essentially equals a product of the balancing parameter and the determined colour frame merit.

In one embodiment, the coefficient encoding merit prior to encoding for a given coefficient type estimates a ratio between a distortion variation provided by

encoding a coefficient having the given type and a rate increase resulting from encoding said coefficient.

In one embodiment, decoding data representing at least one block in the enhancement original INTRA image comprises, for each coefficient for which the coefficient encoding merit prior to encoding is greater than the predetermined block merit, selecting a quantizer depending on the received parameter associated with the concerned coefficient type and on the predetermined block merit, wherein dequantizing symbols is performed using the selected quantizer.

In one particular embodiment, decoding data representing the enhancement original INTRA image comprises determining the coefficient encoding merit prior to encoding for given coefficient type and block type based on the received parameters for the given coefficient type and block type.

In one particular embodiment, a parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type previously received for a previous enhancement INTRA image is reused as the at least one parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type in the enhancement original INTRA image being decoded.

In one particular embodiment, the selected coefficient types of the enhancement original INTRA image being decoded belong to a first group; and

the method further comprises the following steps:

receiving encoded coefficients relating to a second enhancement original INTRA image of the enhancement layer and having coefficient types in a second group;

receiving parameters associated with coefficient types of the second group not included in the first group;

decoding the received coefficients relating to the second enhancement original INTRA image, wherein decoding a received coefficient having a given coefficient type in the first group includes a step of dequantizing using a dequantizer selected based on the previously received parameter associated with the given coefficient type;

transforming the decoded coefficients into pixel values for the second enhancement original INTRA image.

In one particular embodiment, the parameters associated with coefficient types of the first group are received in a first transport unit and wherein the parameters

associated with coefficient types of the second group not in the first group are received in a second transport unit, distinct from the first transport unit.

In one particular embodiment, the information supplied to the decoder for said second image does not include information about the reused parameter(s).

In one particular embodiment, such a parametric probabilistic model is obtained for each type of encoded DCT coefficient in said first image.

In one particular embodiment, parameters of the first-image parametric probabilistic model obtained for at least one said DCT coefficient type are reused for said second image.

In one particular embodiment, the method comprises a step of receiving encoded coefficients relating to a third enhancement original INTRA image of the enhancement layer and a flag indicating whether previously received parameters are valid,

the method comprising the following steps if the received flag indicate that the previously received parameters are valid:

receiving parameters associated with coefficient types of a third group not included in the first and second groups;

decoding the received coefficients relating to the third enhancement original INTRA image, wherein decoding a received coefficient having a given coefficient type in the first or second group includes a step of dequantizing using a dequantizer selected based on the previously received parameter associated with the given coefficient type;

transforming the decoded coefficients into pixel values for the third enhancement original INTRA image;

In one particular embodiment, the method comprises the following steps if the received flag indicate that the previously received parameters are no longer valid:

receiving encoded coefficients relating to the third enhancement original INTRA image and having coefficient types in a first group;

receiving new parameters associated with coefficient types of encoded coefficients relating to the third enhancement original INTRA image;

decoding the received coefficients relating to the third enhancement original INTRA image, wherein decoding a received coefficient having a given coefficient type includes a step of dequantizing using a dequantizer selected based on the received new parameter associated with the given coefficient type;

transforming the decoded coefficients into pixel values for the third enhancement original INTRA image.

In one embodiment, the method further comprises decoding, from the bit-stream, a quad-tree representing a segmentation of the enhancement original INTRA image said plurality of blocks of pixels, each block having a block type, the quad tree having a plurality of levels, each associated with a block size, and leaves associated with blocks and having a value indicating either a label for the concerned block or a subdivision of the concerned block.

In one particular embodiment, decoding the quad tree uses an arithmetic entropy decoding that uses, when decoding the segmentation relating to a given block, conditional probabilities for the various possible leaf values depending on a state of a block in the base layer co-located with said given block.

In one embodiment, the method comprises:

receiving video data of the base layer, video data of the enhancement layer, a table of conditional probabilities and a coded quad-tree representing, by leaf values, an image segmentation into blocks for the enhancement original INTRA image;

decoding video data of the base layer to generate decoded base layer video data having a second resolution, lower than a first resolution, and up-sampling the decoded base layer video data to generate up-sampled video data having the first resolution;

for at least one block represented in the quad-tree, determining the probabilities respectively associated with the possible leaf values based on the received table and depending on a state of a block in the base layer co-located with said block;

decoding the coded quad-tree to obtain the segmentation, including arithmetic entropy decoding the leaf value associated with said block using the determined probabilities;

decoding, using the obtained segmentation, video data of the enhancement layer to generate residual data having the first resolution;

forming a sum of the up-sampled video data and the residual data to generate enhanced video data.

In one embodiment, the method comprises determining the parameter input to the adaptive post-filter based on the predetermined frame merit.

According to a fifth aspect, the invention provides the above decoding method wherein decoding data representing the enhancement original INTER image

comprising a plurality of blocks of pixels, each block having a block type, comprises the steps of:

decoding prediction mode information from the bit-stream for at least one enhancement block of the enhancement original INTER image to obtain a prediction mode having been selected from among a plurality of prediction modes, wherein the plurality of prediction modes includes at least one of:

a base mode prediction mode involving computation, from the base layer, of a base mode prediction image corresponding to the enhancement original INTER image, the base mode prediction image being composed of base mode blocks obtained using prediction information derived from prediction information of the base layer and wherein the enhancement block is derived from one or more base mode blocks of a spatially corresponding region of the base mode prediction image; and

a GRILP prediction mode including: obtaining from the bit-stream the location of a block predictor of the enhancement block within the enhancement original INTER image to be decoded and a residual block comprising difference information between enhancement image residual information and base layer residual information; determining a block predictor in the base layer co-located with the block predictor in the enhancement original INTER image; determining a base-layer residual block corresponding to the difference between the block of the base layer co-located with the enhancement block to be decoded and the determined block predictor in the base layer; reconstructing an enhancement-layer residual block using the determined base-layer residual block and said residual block obtained from the bit stream; reconstructing the enhancement block using the block predictor and the enhancement-layer residual block;

obtaining a prediction block from the selected prediction mode and adding the prediction block to a decoded enhancement residual block to obtain the enhancement block, said enhancement residual block comprising quantized symbols, the decoding of the enhancement original INTER image comprising inverse quantizing these quantized symbols to obtain transformed coefficients.

According to an embodiment the plurality of prediction modes includes an inter difference mode in addition or in replacement of the GRILP mode including for a block to decode: obtaining a motion vector designating a reference block in a EL reference image; computing a difference image between the EL reference image and

an upsampled version of the image of the base layer temporally co-located with the EL reference image; performing a motion compensation on the block of the obtained difference image pointed by the motion vector determined during the motion estimation step to obtain a residual block; adding the obtained residual block to the reference block to obtain a final block predictor used to decode the current block.

In one embodiment, the plurality of prediction modes includes the following prediction modes:

a motion compensated temporal prediction mode within the enhancement layer;

an intra base layer prediction mode where the prediction block is taken from a base block co-located with the enhancement block in an up-sampled decoded version of the corresponding base image;

the base mode prediction mode, wherein each base mode block of the base mode prediction image derives from the co-located base block in the corresponding base image when the co-located base block is intra coded or derives, when the co-located base block is inter coded into a base residual using prediction information, from the block in the enhancement layer that is obtained by applying an up-sampled version of a motion vector of the prediction information onto the base mode block and from an up-sampled decoded version of the base residual;

the GRILP prediction mode and/or the inter difference prediction mode; and

a difference INTRA coding mode.

In one embodiment, in the GRILP prediction mode, when an image of residual data of the base layer is available, determining the base-layer block residual in the base layer comprises:

determining the overlap in the image of residual data between the obtained block predictor and the block predictor used in the encoding of the block co-located with the enhancement block in the base layer;

using the part in the image of residual data corresponding to this overlap if any to reconstruct a part of the enhancement-layer residual block, wherein the samples of the enhancement-layer residual block corresponding to this overlap each involves an addition of a sample of the obtained residual block and a corresponding sample of the base-layer residual block.

In one embodiment, the method comprises de-blocking filtering the base mode prediction image before it is used to provide prediction blocks.

In one particular embodiment, the de-blocking filtering is applied to the boundaries of the base mode blocks of the base mode prediction image.

In one particular embodiment, the method further comprises deriving the organisation of transform units of base blocks in the base layer towards the enhancement layer wherein the de-blocking filtering is applied to the boundaries of the transform units derived from the base layer.

In one embodiment, the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained; and

decoding the enhancement original INTER image further comprises decoding the motion information using a motion information predictor taken from a set including motion information, if any, associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

In one particular embodiment, other motion information of the set is derived from the motion information by adding respective spatial offsets.

In one particular embodiment, the set includes motion information, if any, associated with blocks spatially neighbouring the enhancement block in the enhancement original INTER image, and if no motion information exist for a given neighbouring block, motion information, if any, associated with the base block spatially corresponding to the given neighbouring block in the corresponding base image.

In one embodiment, the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained;

decoding the enhancement original INTER image further comprises decoding the motion information using a motion information predictor taken from a set including more motion information predictors than another set usable for predicting motion information associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

In one particular embodiment, the set of motion information predictors for the enhancement original INTER image includes at least one motion information predictor generated based on motion information from the base layer.

In one embodiment, the set of vector predictors comprises at least one temporal motion information predictor and at least one spatial motion information

predictor, the at least one temporal motion information predictor being positioned before the at least one spatial motion information predictor.

In one embodiment, in case of spatial scalability between the base layer and the enhancement layer with the base layer having a lower spatial resolution than the enhancement layer; and

prediction information, such as a motion vector, is derived or up-sampled for a processing block of size 2Nx2N in the enhancement original INTER image from the base layer of lower spatial resolution, the derivation or up-sampling comprises:

determining whether or not the region of the base layer, spatially corresponding to the processing block, is wholly located within one elementary prediction unit of the base layer; and

in the case where the region of the base layer spatially corresponding to the processing block is fully located within one elementary prediction unit of the base layer, deriving prediction information for that processing block from the base layer prediction information of the said one elementary prediction unit;

otherwise in the case where the region of the base layer spatially corresponding to the processing block overlaps, at least partially, each of a plurality of elementary prediction units,

dividing the processing block into a plurality of sub-processing blocks , each of size NxN such that the region of the base layer spatially corresponding to each sub-processing block is wholly located within one elementary prediction unit of the base layer; and

deriving the prediction information for each sub-processing block from the base layer prediction information of the spatially corresponding elementary prediction unit.

In one particular embodiment, in the case where the corresponding elementary prediction unit of the base layer is Intra-coded then the processing block is predicted from the elementary prediction unit reconstructed and resampled to the enhancement layer resolution.

In one particular embodiment, in the case where the corresponding elementary prediction unit is Inter-coded then the processing block is temporally predicted using motion information derived from the said corresponding elementary prediction unit of the base layer.

In one particular embodiment, the processing block is temporally predicted further using a decoded temporal residual of the corresponding elementary prediction

unit of the base layer, said temporal residual being computed between base layer images, as a function of the motion information of the elementary prediction unit.

In one particular embodiment, the spatial scaling between an image of the enhancement layer and a corresponding image of the base layer is a non-integer ratio. In particular, the non-integer ratio is 1.5.

In one embodiment, the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the method comprises:

obtaining a first set of quantization offsets to be applied to a group of enhancement images, a different quantization offset being obtained for at least two enhancement INTER images belonging to the same temporal depth; and

determining a quantization parameter used for inverse quantization to decode the enhancement INTER images based on the obtained first set of quantization parameters.

In one embodiment, the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the method comprises:

obtaining a first set of quantization offsets to be applied to a group of enhancement images, the same quantization offset being obtained for at least two enhancement INTER images belonging to different temporal depths; and

determining a quantization parameter used for inverse quantization to decode the enhancement INTER images based on the obtained first set of quantization parameters.

In one particular embodiment, among the at least two enhancement INTER images that belong to the same temporal depth, a first offset is obtained for a first enhancement INTER image having a reference image of a first quality and a second offset, larger than said first offset, is obtained for a second enhancement INTER image having a reference image of a second quality lower than said first quality.

In one particular embodiment, the quantization offset obtained for an enhancement INTER image takes into account:

the number of enhancement images using the enhancement INTER image as a reference image for temporal prediction;

the temporal distance of the enhancement INTER image to its reference image having the lowest quantization offset; and

the value of the quantization parameter applied to this reference image.

In one particular embodiment, the quantization offset obtained for an enhancement INTER image is equal or larger than the quantization offset of its reference image having the lowest quantization offset.

In one particular embodiment, the method further comprises encoding, using a second set of different quantization offsets, a group of base images in the base layer that temporally coincides with the group of enhancement images ;

wherein the second set is obtained based on a temporal depth each base image belongs to.

In one embodiment, decoding data representing the enhancement original INTER image further comprises decoding,

from the bit-stream, quad-trees representing a segmentation of respective portions of the enhancement original INTER image and coding modes associated with blocks in the segmentation;

wherein decoding the quad-tree comprises decoding, from a received code associated with a block in the segmentation,

a first coding mode syntax element that indicates whether the coding mode associated with the block is based on temporal/Inter prediction or not,

a second coding mode syntax element that indicates whether a prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information is used or not for encoding the block is activated or not if the first coding mode syntax element refers to temporal/Inter prediction, or indicates whether the coding mode associated with the block is a conventional Intra prediction or based on Inter-layer prediction if the first coding mode syntax element refers to non temporal/Inter prediction, and

in the case in which the coding mode associated with the given block is based on Inter-layer prediction, a third coding mode syntax element that indicates whether the coding mode associated with the given block is the intra base layer mode or the base mode prediction mode.

In an embodiment the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the GRILP prediction sub-mode.

In an embodiment the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the inter difference prediction sub-mode.

In an embodiment a fourth coding mode syntax element indicates whether the inter difference block was used or not for encoding the block if the second coding mode syntax element indicates that the GRILP mode was not used or whether the GRILP mode was used or not for encoding the block if the second coding mode syntax element indicates that the inter difference mode was not used.

In an embodiment at least one high level syntax element indicates which of the coding mode syntax elements are present or not in the bit-stream.

In an embodiment when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding mode syntax element following the removed coding mode syntax element replaces the removed coding mode syntax element and the coding order of the remaining coding mode syntax elements is kept.

In an embodiment when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding order of the remaining coding mode syntax elements is modified.

In an embodiment the modification of the coding order of the remaining coding mode syntax elements takes into account the probability of occurrence of coding modes represented by remaining coding mode syntax elements.

In one embodiment,

decoding the enhancement original INTRA image comprises selecting quantizers from the predetermined block merit to dequantize symbols of the selected coefficient types, the predetermined block merit deriving from a frame merit;

decoding the enhancement original INTER image comprises selecting quantizers from a quantization parameter to inverse quantize the quantized symbols; and

the frame merit and the quantization parameter are computed from a received quality parameter and are linked together with a balancing parameter.

In one embodiment, the decoding method is implemented by a computer, and data, such as image samples, in the base layer are processed using 8-bit words and data, such as image samples, in the enhancement layer are processed using 10-bit words.

In one particular embodiment, data, such as image samples, from the base layer are up-scaled to 10-bit words by multiplying each value by 4, when this date from the base layer is processed for the decoding of the enhancement layer.

A video encoder according to the invention for encoding a video sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, may comprises:

a base layer encoding module for encoding a base layer made of base images;

an enhancement layer encoding module for encoding an enhancement layer made of enhancement images, including an Intra encoding module for encoding at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame prediction only; and an Inter encoding module for encoding at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame prediction.

According to a sixth aspect, the invention provides the above video encoder wherein the Intra encoding module comprises:

a module for obtaining a residual enhancement image as a difference between the enhancement original INTRA image and a decoded version of the corresponding encoded base image in the base layer, the residual enhancement image comprising a plurality of blocks of pixels, each block having a block type;

a transforming module for transforming pixel values for a block among said plurality of blocks into a set of coefficients each having a coefficient type, said block having a given block type;

a merit determining module for determining an initial coefficient encoding merit for each coefficient type;

a coefficient selector for selecting coefficients based, for each coefficient, on the initial coefficient encoding merit for said coefficient type and on a predetermined block merit;

a quantizing module for quantizing the selected coefficients into quantized symbols;

an encoding module for encoding the quantized symbols.

According to a seventh aspect, the invention provides the above video encoder wherein the Intra encoding module comprises:

a module for obtaining a residual enhancement image as a difference between the enhancement original INTRA image and a decoded version of the corresponding encoded base image in the base layer, the residual enhancement image comprising a plurality of blocks of pixels, each block having a block type;

a module for performing an initial segmentation of the residual enhancement image into a set of initial blocks, thus determining, for each initial block, a block type associated with the concerned initial block;

a module for determining, for each block type, an associated set of quantizers based on data corresponding to pixels of blocks having said block type;

a module for selecting, among a plurality of possible segmentations defining an association between each block of this segmentation and an associated block type, the segmentation which minimizes an encoding cost estimated based on a measure of the rate necessary for encoding each block using the set of quantizers associated with the block type of the encoded block according to the concerned segmentation.

According to an eigth aspect, the invention provides the above video encoder wherein the Inter encoding module comprises:

a module for selecting a prediction mode, from among a plurality of prediction modes, for predicting an enhancement block of the enhancement original INTER image, wherein the plurality of prediction modes includes at least one of:

a base mode prediction mode involving computation, from the base layer, of a base mode prediction image corresponding to the enhancement original INTER image, the base mode prediction image being composed of base mode blocks obtained using prediction information derived from prediction information of the base layer and wherein the enhancement block is derived from one or more base mode blocks of a spatially corresponding region of the base mode prediction image; and

a GRILP prediction mode including: obtaining a block predictor candidate for predicting the enhancement block within the enhancement original INTER image and an associated enhancement-layer residual block corresponding to said prediction; determining a block predictor in the base layer co-located with the determined block predictor candidate within the enhancement original INTER image; determining a base-layer residual block associated with the enhancement block in the base layer that is co-located with the enhancement block in the enhancement original INTER image, as the

difference between the co-located enhancement block in the enhancement original INTER image and the determined block predictor in the base layer; determining, for the enhancement block of the enhancement original INTER image, a further residual block corresponding, at least partly, to the difference between the enhancement-layer residual block and the base-layer residual block;

a module for obtaining a prediction block from the selected prediction mode and subtracting the prediction block from the enhancement block of the enhancement original INTER image to obtain a residual block;

a module for transforming pixels values of the residual block to obtain transformed coefficients;

a module for quantizing at least one of the transformed coefficients to obtain quantized symbols;

a module for encoding the quantized symbols into encoded data.

According to an embodiment the plurality of prediction modes includes an inter difference mode in addition or in replacement of the GRILP mode including: performing a motion estimation on a current block of a current Enhancement Layer (EL) image to obtain a motion vector designating a reference block in a EL reference image; computing a difference image between the EL reference image and an upsampled version of the image of the base layer temporally co-located with the EL reference image; performing a motion compensation on the block of the obtained difference image pointed by the motion vector determined during the motion estimation step to obtain a residual block; adding the obtained residual block to the reference block to obtain a final block predictor used to predict the current block.

According to an embodiment, the plurality of prediction modes includes the following prediction modes:

a motion compensated temporal prediction mode within the enhancement layer;

an intra base layer prediction mode where the prediction block is taken from a base block co-located with the enhancement block in an up-sampled decoded version of the corresponding base image;

the base mode prediction mode, wherein each base mode block of the base mode prediction image derives from the co-located base block in the corresponding base image when the co-located base block is intra coded or derives, when the co-located base block is inter coded into a base residual using prediction

information, from the block in the enhancement layer that is obtained by applying an up-sampled version of a motion vector of the prediction information onto the base mode block and from an up-sampled decoded version of the base residual;

the GRILP prediction mode and/or the inter difference prediction mode; and

a difference INTRA coding mode.

A video decoder according to the invention for decoding a scalable video bit-stream, may comprise:

a base layer decoding module decoding a base layer made of base images;

an enhancement layer decoding module decoding an enhancement layer made of enhancement images, including an Intra decoding module for decoding data representing at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame decoding; and an Inter decoding module for decoding data representing at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame decoding.

According to a ninth aspect, the invention provides the above video encoder wherein the Intra decoding module for decoding data representing at least one block of pixels in the enhancement original INTRA image, comprises :

a module for receiving said data and parameters each representative of a probabilistic distribution of a coefficient type;

a module for decoding said data into symbols;

a module for selecting coefficient types for which a coefficient encoding merit prior to encoding, estimated based on the parameter associated with the concerned coefficient type, is greater than a predetermined block merit;

a module for dequantizing, for selected coefficient types, symbols into dequantized coefficients having a coefficient type among the selected coefficient types;

a module for transforming dequantized coefficients into pixel values in the spatial domain for said block.

According to a tenth aspect, the invention provides the above video encoder wherein the Inter decoding module for decoding data representing the enhancement original INTER image comprising a plurality of blocks of pixels, each block having a block type, comprises:

a module for decoding prediction mode information from the bit-stream for at least one enhancement block of the enhancement original INTER image to obtain a

prediction mode having been selected from among a plurality of prediction modes, wherein the plurality of prediction modes includes at least one of:

a base mode prediction mode involving computation, from the base layer, of a base mode prediction image corresponding to the enhancement original INTER image, the base mode prediction image being composed of base mode blocks obtained using prediction information derived from prediction information of the base layer and wherein the enhancement block is derived from one or more base mode blocks of a spatially corresponding region of the base mode prediction image; and

a GRILP prediction mode including: obtaining from the bit-stream the location of a block predictor of the enhancement block within the enhancement original INTER image to be decoded and a residual block comprising difference information between enhancement image residual information and base layer residual information; determining a block predictor in the base layer co-located with the block predictor in the enhancement original INTER image; determining a base-layer residual block corresponding to the difference between the block of the base layer co-located with the enhancement block to be decoded and the determined block predictor in the base layer; reconstructing an enhancement-layer residual block using the determined base-layer residual block and said residual block obtained from the bit stream; reconstructing the enhancement block using the block predictor and the enhancement-layer residual block;

a module for obtaining a prediction block from the selected prediction mode and adding the prediction block to a decoded enhancement residual block to obtain the enhancement block, said enhancement residual block comprising quantized symbols, the decoding of the enhancement original INTER image comprising inverse quantizing these quantized symbols to obtain transformed coefficients.

According to an embodiment, the plurality of prediction modes includes an inter difference mode in addition or in replacement of the GRILP mode including for a block to decode: obtaining a motion vector designating a reference block in a EL reference image; computing a difference image between the EL reference image and an upsampled version of the image of the base layer temporally co-located with the EL reference image; performing a motion compensation on the block of the obtained difference image pointed by the motion vector determined during the motion estimation

step to obtain a residual block; adding the obtained residual block to the reference block to obtain a final block predictor used to decode the current block.

According to an embodiment, the plurality of prediction modes includes the following prediction modes:

a motion compensated temporal prediction mode within the enhancement layer;

an intra base layer prediction mode where the prediction block is taken from a base block co-located with the enhancement block in an up-sampled decoded version of the corresponding base image;

the base mode prediction mode, wherein each base mode block of the base mode prediction image derives from the co-located base block in the corresponding base image when the co-located base block is intra coded or derives, when the co-located base block is inter coded into a base residual using prediction information, from the block in the enhancement layer that is obtained by applying an up-sampled version of a motion vector of the prediction information onto the base mode block and from an up-sampled decoded version of the base residual;

the GRILP prediction mode and/or the inter difference prediction mode; and

a difference INTRA coding mode.

The video encoder and decoder may comprise optional features as defined in the enclosed claims 1-32-61.

The optional features proposed above in connection with the encoding and decoding methods may also apply to the video encoder and decoder respectively just mentioned.

The invention also provides an encoding device for encoding an image substantially as herein described with reference to, and as shown in, Figure 7; Figures 7 and 28; Figures 7, 28 and 42; Figures 7, 28, 42 and at least one from Figures 33, 34, 35, 36A, 38, 39 and 44; Figure 9; Figures 9 and 11; or Figures 9, 11 and at least one from Figures 21, 21A, 21B, 22, 24 and 25 of the accompanying drawings.

The invention also provides a decoding device for decoding a scalable video bit-stream substantially as herein described with reference to, and as shown in, Figure 8; Figures 8 and 29; Figures 8, 29 and 43; Figures 8, 29, 43 and at least one from Figures 33, 34, 35, 36A, 38, 39 and 44; Figure 10; Figures 10 and 12; or Figures 10, 12 and at least one from Figures 21, 21A, 21B, 24A and 25A of the accompanying drawings.

The invention also provides an encoding method for encoding an image substantially as herein described with reference to, and as shown in, Figure 7; Figures 7 and 28; Figures 7, 28 and 42; Figures 7, 28, 42 and at least one from Figures 33, 34, 35, 36A, 38, 39 and 44; Figure 9; Figures 9 and 11; or Figures 9, 11 and at least one from Figures 21, 21A, 21B, 22, 24 and 25 of the accompanying drawings.

The invention also provides a decoding method for decoding a scalable video bit-stream substantially as herein described with reference to, and as shown in, Figure 8; Figures 8 and 29; Figures 8, 29 and 43; Figures 8, 29, 43 and at least one from Figures 33, 34, 35, 36A, 38, 39 and 44; Figure 10; Figures 10 and 12; or Figures 10, 12 and at least one from Figures 21, 21A, 21B, 24A and 25Aof the accompanying drawings

At least parts of the methods according to the invention may be computer implemented. Accordingly, the present invention may take the form of an entirely hardware embodiment, an entirely software embodiment (including firmware, resident software, micro-code, etc.) or an embodiment combining software and hardware aspects which may all generally be referred to herein as a "circuit", "module" or "system". Furthermore, the present invention may take the form of a computer program product embodied in any tangible medium of expression having computer usable program code embodied in the medium.

Since the present invention can be implemented in software, the present invention can be embodied as computer readable code for provision to a programmable apparatus on any suitable carrier medium, for example a tangible carrier medium or a transient carrier medium. A tangible carrier medium may comprise a storage medium such as a floppy disk, a CD-ROM, a hard disk drive, a magnetic tape device or a solid state memory device or the like. A transient carrier medium may include a signal such as an electrical signal, an electronic signal, an optical signal, an acoustic signal, a magnetic signal or an electromagnetic signal, e.g. a microwave or RF signal.

BRIEF DESCRIPTION OF THE DRAWINGS

Other particularities and advantages of the invention will also emerge from the following description, illustrated by the accompanying drawings, in which:

- **Figure 1A** schematically illustrates a data communication system in which one or more embodiments of the invention may be implemented;

- **Figure 1B** illustrates an example of a device for encoding or decoding images, capable of implementing one or more embodiments of the present invention;
- **Figure 2** illustrates all-INTRA coding structure for scalable video coding (SVC);
- **Figure 3** illustrates a low-delay temporal coding structure according to the HEVC standard;
- **Figure 4** illustrates a random access temporal coding structure according to the HEVC standard;
- **Figure 5** illustrates a standard video encoder, compliant with the HEVC standard for video compression;
- **Figure 5A** schematically illustrates elementary prediction units and prediction unit concepts specified in the HEVC standard;
- **Figure 6** illustrates a block diagram of a decoder, compliant with standard HEVC or H.264/AVC and reciprocal to the encoder of **Figure 5**;
- **Figure 7** illustrates a block diagram of a scalable video encoder according to embodiments of the invention, compliant with the HEVC standard in the compression of the base layer;
- **Figure 8** illustrates a block diagram of a scalable decoder according to embodiments of the invention, compliant with standard HEVC or H.264/AVC in the decoding of the base layer, and reciprocal to the encoder of **Figure 7**;
- **Figure 9** schematically illustrates encoding sub-part handling enhancement INTRA images in the scalable video encoder architecture of **Figure 7**;
- **Figure 10** schematically illustrates decoding sub-part handling enhancement INTRA images in the scalable video decoder architecture of **Figure 8**, and reciprocal to the encoding features of **Figure 9**;
- **Figure 11** illustrates the encoding process associated with the residuals of an enhancement layer according to at least one embodiment;
- **Figure 12** illustrates the decoding process consistent with the encoding process of **Figure 11** according to at least one embodiment;
- **Figure 13** shows an exemplary embodiment of a process for determining optimal quantizers according to embodiments of the invention at the block level;
- **Figure 14** illustrates an example of a quantizer based on Voronoi cells;
- **Figure 15** shows the correspondence between data in the spatial domain (pixels) and data in the frequency domain;
- **Figure 16** illustrates an exemplary distribution over two quanta;

- **Figure 17** shows exemplary rate-distortion curves, each curve corresponding to a specific number of quanta;
- **Figure 18** shows the rate-distortion curve obtained by taking the upper envelope of the curves of **Figure 17**;
- **Figure 19** depicts several rate-distortion curves obtained for various possible parameters of the DCT coefficient distribution;
- **Figure 20** shows a merit-distortion curve for a DCT coefficient;
- **Figure 21** shows an exemplary embodiment of a process for determining optimal quantizers according to embodiments of the invention at the image level;
- **Figure 21A** shows a process for determining luminance frame merit for INTRA images and final quality parameter for INTER images, from a user-specified quality parameter;
- **Figure 21B** shows a process for determining optimal quantizers according to embodiments of the invention at the level of a video sequence;
- **Figure 22** shows an encoding process of residual enhancement INTRA image according to embodiments of the invention;
- **Figure 23** illustrates a bottom-to-top algorithm used in the frame of the encoding process of **Figure 22**;
- **Figure 24** shows an exemplary method for encoding parameters representing the statistical distribution of DCT coefficients;
- **Figure 24A** shows a corresponding method for decoding parameters;
- **Figure 24B** shows a possible way of distributing encoded coefficient and parameters in distinct NAL units;
- **Figure 25** shows the adaptive post-filtering applied at the encoder;
- **Figure 25A** shows the post-filtering applied at the decoder;
- **Figure 26A** illustrates the quantization offsets typically used for a GOP of size 8 in the prior art;
- **Figures 26B** to **26F** give examples of quantization schemes according to various embodiments of the invention;
- **Figures 27** to **27C** are trees illustrating syntaxes for encoding a coding mode tree according to embodiments of the invention;
- **Figure 28** schematically illustrates encoding sub-part handling enhancement INTER images in the scalable video encoder architecture of **Figure 7**;

- **Figure 29** schematically illustrates decoding sub-part handling enhancement INTRA images in the scalable video decoder architecture of **Figure 8**, and reciprocal to the encoding features of **Figure 28**;
- **Figure 30** schematically illustrates prediction information up-sampling according to an embodiment of the invention in the case of a non-integer scaling ratio between base and enhancement layers;
- **Figure 31A** schematically illustrates prediction modes in embodiments of the scalable architectures of **Figures 28** and **29**;
- **Figure 31B** schematically illustrates inter-layer derivation of prediction information for 4x4 enhancement layer blocks in accordance with an embodiment of the invention;
- **Figure 32** schematically illustrates derivation of prediction units of the enhancement layer in accordance with an embodiment of the invention;
- **Figure 33** is a flowchart illustrating steps of a method of deriving prediction information in accordance with an embodiment of the invention;
- **Figure 34** is a flowchart illustrating steps of a method of deriving prediction information in accordance with an embodiment of the invention;
- **Figure 35** schematically illustrates the construction of a Base Mode prediction image according to an embodiment of the invention;
- **Figure 36** schematically illustrates processing of a base mode prediction image in accordance with an embodiment of the invention;
- **Figure 36A** is flow chart illustrating the de-blocking filtering of the base mode prediction image;
- **Figure 36B** schematically illustrates a method of deriving a transform tree from a base layer to an enhancement layer;
- **Figures 36C** and **36D** schematically illustrate transform tree interlayer derivation in the case of dyadic spatial scalability;
- **Figure 37** illustrates the residual prediction in the GRILP mode in an embodiment of the invention;
- **Figure 38** illustrates the method used for GRILP residual prediction in an embodiment of the invention;
- **Figure 39** illustrates the method used for GRILP decoding in an embodiment of the invention;
- **Figure 40** illustrates an alternative embodiment of GRILP mode in the context of single loop encoding;

- **Figure 41** illustrates an alternative embodiment of GRILP mode in the context of intra coding;
- **Figure 42** is an overall flow chart of an algorithm according to an embodiment of the invention used to encode an INTER image;
- **Figure 43** is an overall flow chart of an algorithm according to the invention used to decode an INTER image, complementary to the encoding algorithm of **Figure 42**;
- **Figure 44** shows a schematic of the AMVP predictor set derivation for an enhancement image of a scalable codec of the HEVC type according to a particular embodiment;
- **Figure 45** illustrates spatial and temporal blocks that can be used to generate motion vector predictors in AMVP and Merge modes of scalable HEVC coding and decoding systems according to a particular embodiment;
- **Figure 46** shows a schematic of the derivation process of motion vectors for an enhancement image of a scalable codec of the HEVC type, according to a particular embodiment, for the Merge modes;
- **Figure 47** shows an example of spatial positions of the neighboring blocks of the current block in the enhancement image and their co-located blocks in the base image;
- **Figure 48A to 48G** illustrate alternative coding mode trees to the coding mode tree of **Figure 27**.

DETAILED DESCRIPTION OF EMBODIMENTS OF THE INVENTION

Figure 1A illustrates a data communication system in which one or more embodiments of the invention may be implemented. The data communication system comprises a sending device, in this case a server 1, which is operable to transmit data packets of a data stream to a receiving device, in this case a client terminal 2, via a data communication network 3. The data communication network 3 may be a Wide Area Network (WAN) or a Local Area Network (LAN). Such a network may be for example a wireless network (Wifi / 802.11a or b or g or n), an Ethernet network, an Internet network or a mixed network composed of several different networks. In a particular embodiment of the invention the data communication system may be, for example, a digital television broadcast system in which the server 1 sends the same data content to multiple clients.

The data stream 4 provided by the server 1 may be composed of multimedia data representing video and audio data. Audio and video data streams may, in some embodiments, be captured by the server 1 using a microphone and a camera respectively. In some embodiments data streams may be stored on the server 1 or received by the server 1 from another data provider. The video and audio streams are coded by an encoder of the server 1 in particular for them to be compressed for transmission.

In order to obtain a better ratio of the quality of transmitted data to quantity of transmitted data, the compression of the video data may be of motion compensation type, for example in accordance with the HEVC type format or H.264/AVC type format and including features of the invention as described below.

A decoder of the client 2 decodes the reconstructed data stream received by the network 3. The reconstructed images may be displayed by a display device and received audio data may be reproduced by a loud speaker. Reflecting the encoding, the decoding also includes features of the invention as described below.

Figure 1B shows a device 10, in which one or more embodiments of the invention may be implemented, illustrated arranged in cooperation with a digital camera 5, a microphone 6 (shown via a card input/output 11), a telecommunications network 3 and a disc 7, comprising a communication bus 12 to which are connected:

- a central processing CPU 13, for example provided in the form of a microprocessor
- a read only memory (ROM) 14 comprising a program 14A whose execution enables the methods according to an embodiment of the invention. This memory 14 may be a flash memory or EEPROM;
- a random access memory (RAM) 16 which, after powering up of the device 10, contains the executable code of the program 14A necessary for the implementation of an embodiment of the invention. This RAM memory 16, being random access type, provides fast access compared to ROM 14. In addition the RAM 16 stores the various images and the various blocks of pixels as the processing is carried out on the video sequences (transform, quantization, storage of reference images etc.);
- a screen 18 for displaying data, in particular video and/or serving as a graphical interface with the user, who may thus interact with the programs according to an embodiment of the invention, using a keyboard 19 or any other means e.g. a mouse (not shown) or pointing device (not shown);

- a hard disc 15 or a storage memory, such as a memory of compact flash type, able to contain the programs of an embodiment of the invention as well as data used or produced on implementation of an embodiment of the invention;

- an optional disc drive 17, or another reader for a removable data carrier, adapted to receive a disc 7 and to read/write thereon data processed, or to be processed, in accordance with an embodiment of the invention and;

- a communication interface 9 connected to a telecommunications network 34

- connection to a digital camera 5

The communication bus 12 permits communication and interoperability between the different elements included in the device 10 or connected to it. The representation of the communication bus 12 given here is not limiting. In particular, the CPU 13 may communicate instructions to any element of the device 10 directly or by means of another element of the device 10.

The disc 7 can be replaced by any information carrier such as a compact disc (CD-ROM), either writable or rewritable, a ZIP disc or a memory card. Generally, an information storage means, which can be read by a micro-computer or microprocessor, which may optionally be integrated in the device 10 for processing a video sequence, is adapted to store one or more programs whose execution permits the implementation of the method according to an embodiment of the invention.

The executable code enabling the coding device to implement an embodiment of the invention may be stored in ROM 14, on the hard disc 15 or on a removable digital medium such as a disc 7.

The CPU 13 controls and directs the execution of the instructions or portions of software code of the program or programs of an embodiment of the invention, the instructions or portions of software code being stored in one of the aforementioned storage means. On powering up of the device 10, the program or programs stored in non-volatile memory, e.g. hard disc 15 or ROM 14, are transferred into the RAM 16, which then contains the executable code of the program or programs of an embodiment of the invention, as well as registers for storing the variables and parameters necessary for implementation of an embodiment of the invention.

It should be noted that the device implementing an embodiment of the invention, or incorporating it, may be implemented in the form of a programmed apparatus. For example, such a device may then contain the code of the computer program or programs in a fixed form in an application specific integrated circuit (ASIC).

The device 10 described here and, particularly, the CPU 13, may implement all or part of the processing operations described below.

Figure 2 illustrates the structure of a scalable video stream 20, when all images or frames are encoded in INTRA mode. As shown, an all-INTRA coding structure consists of a series of images which are encoded independently from each other. This makes it possible to decode each image by its own.

The base layer 21 of the scalable video stream 20 is illustrated at the bottom of the figure. In this base layer, each image is INTRA coded and is usually referred to as an "I" image. INTRA coding of an image involves predicting a macroblock or block or coding unit according to HEVC language from its directly neighbouring blocks within the same image.

For example, the base layer may be made of high definition (HD) frames.

A spatial enhancement layer 22 is encoded on top of the base layer 21. It is illustrated at the top of **Figure 2**. This spatial enhancement layer 22 introduces some spatial refinement information over the base layer. In other words, the decoding of this spatial layer leads to a decoded video sequence that has usually a higher spatial resolution than the base layer. The higher spatial resolution adds to the quality of the reproduced images.

Another type of scalability is the SNR scalability where the enhancement layer has the same resolution as the base layer but provides an improved image quality (improvement of the Signal to Noise Ratio: SNR). In this case again, the enhancement layer adds to the quality of the reproduced images.

In the figure only one enhancement layer is shown, but several enhancement layers could be added providing different resolution and quality improvements.

As illustrated in the figure, each enhancement image, denoted an 'EI' image, is INTRA coded. An enhancement INTRA image is encoded independently from other enhancement images. It is coded in a predictive way, by predicting it only from the temporally coincident image in the base layer. This involves inter-layer prediction.

The enhancement layer may be made of ultra-high definition (UHD) images. UHD is typically four times (4k2k pixels) the definition of an HD video which is the current standard definition video. Other resolution for the enhancement layer may be the very ultra high definition, which is sixteen times that definition (i.e. 8k4k pixels). In the case of SNR scalability, the enhancement layer has the same resolution as the base layer: HD in this example.

Known down-sampling mechanisms are known to obtain HD base layer image from an original sequence of UHD images.

Figures 3 and **4** illustrate video coding structures that involves both INTRA frames (I) and INTER frames ("B" in the Figures), in so-called "low delay" and "random access" configurations, respectively. These are the two coding structures comprised in the common test conditions in the HEVC standardization process.

Figure 3 shows the low-delay temporal coding structure 30. In this configuration, an input image frame is predicted from several already coded images. Therefore, only forward temporal prediction, as indicated by arrows 31, is allowed, which ensures the low delay property. The low delay property means that on the decoder side, the decoder is able to display a decoded image straight away once this image is in a decoded format, as represented by arrow 32 (POC index is the index of the images in the video sequence). Note: the input video sequence is shown as comprised of a base layer 33 and an enhancement layer 34, which are each further comprised of a first INTRA image I and subsequent INTER images B.

In addition to temporal prediction, inter-layer prediction between the base 33 and enhancement layer 34 is also illustrated in **Figure 3** and referenced by arrows, including arrow 35. Indeed, the scalable video coding of the enhancement layer 34 aims to exploit the redundancy that exists between the coded base layer 33 and the enhancement layer 34, in order to provide good coding efficiency in the enhancement layer 34.

Figure 4 illustrates the random access temporal coding structure 40 e.g. as defined in the HEVC standard. The input sequence is broken down into groups of pictures or images, here indicated by arrows GOP. The random access property means that several access points are enabled in the compressed video stream, i.e. the decoder can start decoding the sequence at an image which is not necessarily the first image in the sequence. This takes the form of periodic INTRA-frame coding in the stream as illustrated by **Figure 4**.

In addition to INTRA images, the random access coding structure allows INTER prediction, both forward 41 and backward 42 (in relation to the display order as represented by arrow 43) predictions can be effected. This is achieved by the use of B images, as illustrated. The random access configuration also provides temporal scalability feature, which takes the form of the hierarchical B images, B0 to B3 as illustrated, the organization of which is shown in the Figure.

As for the low delay coding structure of **Figure 3**, additional prediction tools are used in the coding of enhancement images: inter-layer prediction tools.

As shown in the Figures, each enhancement image has a temporally corresponding base image in the base layer. This is the most common situation for scalable video sequences. However, different time sampling of the images between the base layer and the enhancement layer may exist, in which case the teachings of the invention as described herein can still apply. Indeed, missing images in a layer compared to another layer may be generated through interpolation from neighbouring images of the same layer.

Before describing scalable video coder and decoder according to embodiments of the invention, standard video encoding device and decoding device are first described with reference to **Figures 5** and **6**.

Figure 5 illustrates a standard video encoding device, of a generic type, conforming to the HEVC or H.264/AVC video compression system. A block diagram 50 of a standard HEVC or H.264/AVC encoder is shown.

The input to this non-scalable encoder consists in the original sequence of frame images 51 to compress. The encoder successively performs the following steps to encode a standard video bit-stream.

A first image to be encoded (compressed) is divided into pixel blocks, called coding unit in the HEVC standard. The first image is thus split into blocks or macroblocks 52.

Figure 5A depicts the coding units and prediction unit concepts specified in the HEVC standard. These concepts are sometimes referred to by the word "block" or "macroblock" below. A coding unit of an HEVC image corresponds to a square block of that image, and can have a size in a pixel range from 8x8 to 64x64. A coding unit which has the highest size authorized for the considered image is also called a Largest Coding Unit (LCU) or CTB (coded tree block) 510. As described below, for each coding unit of an enhancement INTER image, the encoder decides how to partition it into one or several prediction units (PU) 520. Each prediction unit can have a square or rectangular shape and is given a prediction mode (INTRA or INTER) and some prediction information. With respect to INTRA prediction, the associated prediction parameters consist in the angular direction used in the spatial prediction of the considered prediction unit, associated with corresponding spatial residual data. In case of INTER prediction, the prediction information comprises the reference image indices and the motion vector(s) used to predict the considered prediction unit, and the

associated temporal residual texture data. Illustrations 5A-A to 5A-H show some of the possible arrangements of partitioning which are available.

Back to **Figure 5**, depending on whether the first image is an INTRA image I or an INTER image B, coding through motion estimation/prediction 53/55 is respectively non-activate (INTRA-frame coding) or active (INTER-frame coding). The INTRA prediction is always active.

Each block of an INTRA image undergoes INTRA prediction 56 to determine the spatial neighbouring block (prediction block) that would provide the best performance to predict the current block. Then latter is then encoded in INTRA mode using reference to the prediction block.

Each block of an INTER image first undergoes a motion estimation operation 53, which comprises a search, among reference images stored in a dedicated memory buffer 54, for reference blocks that would provide a good prediction of the current block. This motion estimation step provides one or more reference image indexes which contain the found reference blocks, as well as the corresponding motion vectors. A motion compensation step 55 then applies the estimated motion vectors on the found reference blocks and uses it to obtain a residual block that will be coded later on. Moreover, an Intra prediction step 56 determines the spatial prediction mode that would provide the best performance to predict the current block and encode it in INTRA mode.

Afterwards, a coding mode selection mechanism 57 chooses the coding mode, among the spatial and temporal predictions, which provides the best rate distortion trade-off in the coding of the current block of the INTER image.

Whatever the image considered, be INTRA or INTER, the difference between the current block 52 (in its original version) and the prediction block obtained through Intra prediction or motion compensation (not shown) is calculated. This provides the (temporal or spatial) residual to compress. The residual block then undergoes a transform (DCT) and a quantization 58. Entropy coding 59 of the so-quantized coefficients QTC (and associated motion data MD) is performed. The compressed texture data associated to the coded current block 999 is sent for output.

Finally, the current block is reconstructed by scaling and inverse transform 58'. This comprises inverse quantization and inverse transform, followed by a sum between the inverse transformed residual and the prediction block of the current block. Once the current image is reconstructed and post-processed 102, it is stored in a

memory buffer 54 (the DPB, Decoded Picture Buffer) so that it is available for use as a reference image to predict any subsequent image to be encoded.

Finally, a last entropy coding step is given the coding mode and, in case of an Inter block, the motion data, as well as the quantized DCT coefficients previously calculated. This entropy coder encodes each of these data into their binary form and encapsulates the so-encoded block into a container called NAL unit (Network Abstract Layer). A NAL unit contains all encoded coding units (i.e. blocks) from a given slice. A coded HEVC bit-stream consists in a series of NAL units.

In order to reduce the cost of encoding motion information, a motion vector may be encoded in terms of a difference between the motion vector and a motion vector predictor, typically selected from a set of vector predictors including spatial motion vectors (one or more motion vectors of the blocks surrounding the block to encode) and temporal motion vectors (, known as Advanced Motion Vector Prediction (AMVP) in HEVC..

A motion vector competition (MVCOMP) consists in determining from among the set of motion vector predictors or candidates (a candidate being a particular type of predictor for a particular prediction mode) which motion vector predictor or candidate minimizes the encoding cost, typically a rate-distortion cost, of the residual motion vector (difference between the motion vector predictor and the current block motion vector).

According to the current HEVC design, three modes can be used for temporal prediction (Inter prediction): Inter mode, Merge mode, and Merge Skip mode. A set of motion vector predictors containing at most two predictors is used for the Inter mode and at most five predictors is used for the Merge Skip mode and the Merge mode. The main difference between these modes is the data signaling in the bit-stream.

In the Inter mode all data are explicitly signaled. This means that the texture residual is coded and inserted into the bit-stream (the texture residual is the difference between the current block and the Inter prediction block). For the motion information, all data are coded. Thus, the direction type is coded (uni or bi-directional). The list index (L0 or L1 list), if needed, is also coded and inserted into the bit-stream. The related reference image indexes are explicitly coded and inserted into the bit-stream. The motion vector value is predicted by the selected motion vector predictor. The motion vector residual for each component is then coded and inserted into the bit-stream followed by the predictor index.

In the Merge mode, the texture residual and the predictor index are coded and inserted into the bit-stream. A motion vector residual, direction type, list or reference image index are not coded. These motion parameters are derived from the predictor index. Thus, the predictor, referred to as candidate, is the predictor of all data of the motion information.

In the Merge Skip mode no information is transmitted to the decoder side except for the "mode" and the predictor index. In this mode the processing is similar to the Merge mode except that no texture residual is coded or transmitted. The pixel values of a Merge Skip block are the pixel values of the block predictor.

Figure 6 provides a block diagram of a standard HEVC or H.264/AVC decoding system 60. This decoding process of a H.264 bit-stream 61 starts by the entropy decoding 62 of each block (array of pixels) of each coded image in the bit-stream. This entropy decoding provides the coding mode, the motion data (reference image indexes, motion vectors of Inter coded macroblocks) and residual data. This residual data consists in quantized and transformed DCT coefficients. Next, these quantized DCT coefficients undergo inverse quantization (scaling) and inverse transform operations 63.

The decoded residual is then added to the temporal 64 or Intra 65 prediction macroblock of current macroblock, to provide the reconstructed macroblock. The choice 69 between INTRA or INTER prediction depends on the prediction mode information which is provided by the entropy decoding step. It is to be noted that encoded Intra-frames comprise only Intra predicted macroblocks and no Inter predicted macroblock.

The reconstructed macroblock finally undergoes one or more in-loop post-filtering processes, e.g. deblocking 66, which aim at reducing the blocking artefact inherent to any block-based video codec, and improve the quality of the decoded image.

The full post-filtered image is then stored in the Decoded Picture Buffer (DPB), represented by the frame memory 67, which stores images that will serve as references to predict future images to decode. The decoded images 68 are also ready to be displayed on screen.

A scalable video coder according to the invention and a corresponding scalable video decoder are now described with reference to **Figures 7** to **47**.

Figure 7 illustrates a block diagram of a scalable video encoder, which comprises a straightforward extension of the standard video coder of **Figure 5**, towards

a scalable video coder. This video encoder may comprise a number of subparts or stages, illustrated here are two subparts or stages A7 and B7 producing data corresponding to a base layer 73 and data corresponding to one enhancement layer 74. Additional subparts A7 may be contemplated in case other enhancement layers are defined in the scalable coding scheme. Each of the subparts A7 and B7 follows the principles of the standard video encoder 50, with the steps of transformation, quantization and entropy coding being applied in two separate paths, one corresponding to each layer.

The first stage B7 aims at encoding the H.264/AVC or HEVC compliant base layer of the output scalable stream, and hence is identical to the encoder of **Figure 5**. Next, the second stage A7 illustrates the coding of an enhancement layer on top of the base layer. This enhancement layer brings a refinement of the spatial resolution to the (down-sampled 77) base layer.

As illustrated in **Figure 7**, the coding scheme of this enhancement layer is similar to that of the base layer, except that for each block or coding unit of a current INTER image 51 being compressed or coded, additional prediction modes can be chosen by the coding mode selection module 75. These are described below with reference to **Figures 26 to 47**.

In addition, INTRA-frame coding is improved compared to standard HEVC. This is described below with reference to **Figures 9** to **25**.

The new coding modes correspond to the inter-layer prediction 76. Inter-layer prediction 76 consists in re-using data coded in a layer lower than current refinement or enhancement layer (e.g. base layer), as prediction data of the current coding unit.

The lower layer used is called the reference layer for the inter-layer prediction of the current enhancement layer. In case the reference layer contains a image that temporally coincides with the current image to encode, then it is called the base image of the current image. As described below, the co-located block (at same spatial position) of the current coding unit that has been coded in the reference layer can be used to provide data in view of building or selecting a prediction unit or block to predict the current coding unit. More precisely, the prediction data that can be used from the co-located block includes the coding mode, the block partition or break-down, the motion data (if present) and the texture data (temporal residual or reconstructed block) of that co-located block. In case of a spatial enhancement layer, some up-sampling 78 operations of the texture and prediction data are performed.

Figure 8 presents a block diagram of a scalable video decoder 80 which would apply on a scalable bit-stream made of two scalability layers, e.g. comprising a base layer and an enhancement layer, for example the bit-stream generated by the scalable video encoder of **Figure 7**. This decoding process is thus the reciprocal processing of the scalable coding process of the same Figure. The scalable bit-stream being decoded 81, as shown in **Figure 8** is made of one base layer and one spatial enhancement layer on top of the base layer, which are demultiplexed 82 into their respective layers.

The first stage of **Figure 8** concerns the base layer decoding process B8. As previously explained for the non-scalable case, this decoding process starts by entropy decoding 62 each coding unit or block of each coded image in the base layer. This entropy decoding 62 provides the coding mode, the motion data (reference image indexes, motion vectors of Inter coded macroblocks) and residual data. This residual data consists of quantized and transformed DCT coefficients. Next, these quantized DCT coefficients undergo inverse quantization and inverse transform operations 63. Motion compensation 64 or Intra prediction 65 data can be added 8C.

Deblocking 66 is effected. The so-reconstructed residual data is then stored in the frame buffer 67.

Next, the decoded motion and temporal residual for Inter blocks, and the reconstructed blocks are stored into a frame buffer in the first stage B8 of the scalable decoder of **Figure 8**. Such frames contain the data that can be used as reference data to predict an upper scalability layer.

Next, the second stage A8 of **Figure 8** performs the decoding of a spatial enhancement layer A8 on top of the base layer decoded by the first stage. This spatial enhancement layer decoding involves the entropy decoding of the second layer 81, which provides the coding modes, motion information as well as the transformed and quantized residual information of blocks of the second layer, and other parameters as described below (e.g. channel parameters for INTRA-coded images).

Next step consists in predicting blocks in the enhancement image. The choice 87 between different types of block prediction modes (those suggest above with reference to the encoder of **Figure 7**) depends on the prediction mode obtained from the entropy decoding step 62.

Note that the blocks of INTRA-coded images are all Intra predicted, while the blocks of INTER-coded images are predicted through either Intra prediction or Inter prediction, among the available prediction coding modes. Details on the Intra frame

coding and on the several inter-layer prediction modes are provided below, from which prediction blocks are obtained.

The result of the entropy decoding 62 undergoes inverse quantization and inverse transform 86, and then is added 8D to the obtained prediction block.

The obtained block is post-processed 66 to produce the decoded enhancement image that can be displayed.

As it transpires from above, different coding/decoding mechanisms are implemented in this embodiment of the invention when coding Intra frames and when coding Inter frames.

Below, INTRA-frame encoding features and corresponding decoding features according to embodiments of the invention are first described with reference to **Figures 9** to **25**. Then, INTER-frame encoding features and corresponding decoding features are described with reference to **Figures 26** to **47**.

These features are described with reference to merely the same example of codec. However, other embodiments of the invention may include parts only of these features, in which case HEVC features may replace non-implemented features. For illustrative purposes, these optional features to implement comprise but are not limited to: Intra frame encoding; use of merits to select coefficients to encode; implementation of iterative segmentation of a residual enhancement image; use of spatially oriented activity during initial segmentation; prediction of channel parameters from one image to the other; use of balancing parameters between luminance and chrominance components when determining frame merits; use of conditional probabilities from base layer when encoding the quad tree representing a segmentation of a residual enhancement image; post-filtering parameter for Intra frame decoding that is function of coded content; coding of the parameters representing the distribution of the DCT coefficients; distribution of the encoded coefficients in distinct NAL units; balancing the rate in the video by determining merit for Intra image and quality parameter for Inter images; Inter frame encoding; Inter layer prediction; Intra base layer prediction; use of base mode prediction; implementation of a particular up-sampling scheme for up-sampling prediction information from base layer to enhancement layer with a ratio of 1.5 between the layer resolutions; use of generalized residual inter-layer prediction; inter-layer motion information prediction and re-arrangement of predictors within the set of motion information predictors; use of a different QP offset scheme between the base layer and the enhancement layer; use of a bit depth different between base layer

computation and enhancement layer computation; syntax for describing the coding modes used.

Figure 9 illustrates a particular type of scalable video encoder architecture 90. The described encoding features handles enhancement INTRA images according to a particular coding way, below referred to as a low complexity coding (LCC) mechanism.

In the example, the disclosed encoder is dedicated to the encoding of a spatial or SNR (signal to noise) enhancement layer on top of a standard coded base layer. The base layer is compliant with the HEVC or H.264/AVC video compression standard. In other embodiments, the base layer may implement all or part of the coding mechanisms for INTER images, in particular LCC, described in relation with the enhancement layer.

The overall architecture of the encoder 90 involving LCC is now described. The input full resolution original image 91 is down-sampled 90A to the base layer resolution level 92 and is encoded 90B with HEVC. This produces a base layer bit-stream 94. The input full resolution original image 91 is now represented by a base layer which is essentially at a lower resolution than the original. Then the base layer image 93 is reconstructed 90C to produce a decoded base layer image 95 and up-sampled 90D to the top layer resolution in case of spatial scalability to produce an image 96. Thus information from only one (base) layer of the original image 91 is now available. This constitutes a decrease in image data available and a lower quality image.

The up-sampled decoded base layer image 96 is then subtracted 90E, in the pixel domain, from the enhancement image corresponding to the full resolution original image 91 to get a residual enhancement image X 97.

The information contained in X is the error or pixel difference due to the base layer encoding/internal decoding (e.g. quantization and post-processing) and the up-sampling. It is also known as a "residual".

Briefly, the residual enhancement image 97 is now subjected to the encoding process 90F which comprises transformation, quantization and entropy operations. This is the above-mentioned LCC mechanism. The processing is performed sequentially on macroblocks or “coding units” using a DCT (Direct Cosine Transform) function, to produce a DCT profile over the global image area. Quantization is performed by fitting with GGD (Generalised Gaussian Distribution) functions the

values taken by DCT coefficient, per DCT channel. Use of such functions allows flexibility in the quantization step, with a smaller step being available for more central regions of the curve. An optimal centroid position per quantization step may also be applied to optimize the quantization process. Entropy coding is then applied (e.g. using arithmetic coding) for the quantized data. The result is the coded enhancement layer 98 associated in the coding with the original image 91. The coded enhancement layer is also converted and added to the enhancement layer bit-stream 99 with its associated parameters 99' (99 prime).

For down sampling, H.264/SVC down-sampling filters are used and for up sampling, the DCTIF interpolation filters of quarter-pixel motion compensation in HEVC are used.

Exemplary 8-tap interpolation filters for luma component and exemplary 4-tap interpolation filters for chroma components are reproduced below, where phase ½ is used to obtain an additional up-sampled pixel in case of dyadic scalability and phases 1/3 and 2/3 are used to obtain two additional up-sampled pixels (in replacement of a central pixel before up-sampling) in case of spatial scalability with ratio equal to 1.5.

Phase	Luma Filter coefficients	Chroma Filter Coefficients
0	{ 0, 0, 0, 64, 0, 0, 0, 0}	{ 0, 64, 0, 0}
1/3	{-1, 4, -11, 52, 26, -8, 3, -1}	{-5, 50, 22, -3}
½	{-1, 4, -11, 40, 40, -11, 4, -1}	{-4, 36, 36, -4}
2/3	{-1, 3, -8, 26, 52, -11, 4, -1}	{-3, 22, 50, -5}

Table 1: phases and filter coefficients used in the texture up-sampling process

The residual enhancement image is encoded using DCT and quantization, which will be further elucided with reference to **Figure 11**. The resulting coded enhancement layer 98 consists of coded residual data as well as some parameters used to model DCT channels of the residual enhancement image. It is recalled that the process described here belongs to the INTRA-frame coding process.

As visible on **Figure 9**, the encoded DCT image is also decoded and inverse transformed 90G to obtain the decoded residual image in the pixel domain (also computed at the decoder). This decoded residual image is summed 90H with the up-sampled decoded base layer image in order to obtain the rough enhanced version of the image.

Adaptive post filtering is then applied to this rough decoded image such that the post-filtered decoded image is as close as possible to the original image (raw video). In practice, the filters are for instance selected to minimize a rate-distortion cost.

Parameters of the applied post-filters (for instance a deblocking filter, a sample adaptive offset filter and an adaptive loop filter, as described in more detail below with reference to **Figure 25**) are thus adjusted to obtain a post-filtered decoded image as close as possible to the raw video and the post-filtering parameters thus determined are sent to the decoder in a dedicated bit stream 99".

It may be noted that the resulting image (post-filtered decodes image) is a reference image to be used in the encoding loop of systems using temporal prediction as it is the representation eventually used at the decoder as explained below.

Figure 10 illustrates a scalable video decoder 100 associated with the type of scalable video encoder architecture 90 shown in **Figure 9**. The described decoding features handles enhancement INTRA images according to the decoding part of the LCC mechanism.

The inputs to the decoder 100 are equivalent to the base layer bit-stream 94 and the enhancement layer bit-stream 99, with its associated parameters 99' (99 prime). The input bit-stream to that decoder comprises the HEVC-coded base layer 93, enhancement residual coded data 98, and parameters 99' of the DCT channels in the residual enhancement image.

First, the base layer is decoded 100A, which provides a reconstructed base image 101. The reconstructed base image 101 is up-sampled 100B to the enhancement layer resolution to produce an up-sampled decoded base image 102. Then, the enhancement layer 98 is decoded using a residual data decoding process 100C further described in association with **Figure 12**. This process is invoked, which provides successive de-quantized DCT blocks 103. These DCT blocks are then inverse transformed and added 100D to their co-located up-sampled block from the up-sampled decoded base image 102. The so-reconstructed enhancement image 104 finally undergoes HEVC post-filtering processes 100E, i.e. de-blocking filter, sample adaptive offset (SAO) and/or Adaptive Loop Filter (ALF), based on received post-filtering parameters 99". A filtered reconstructed image 105 of full resolution is produced and can be displayed.

Figure 11 illustrates the coding process 110 associated with the residuals of an enhancement layer, an example of which is image 97 shown in **Figure 9**. The coding process comprises transformation by DCT function, quantization and entropy

coding. This process applies on a set of blocks or coding units, such as a complete residual image or a slice as defined in HEVC.

The input 97 to the encoder consists of a set of DCT blocks forming the residual enhancement layer. Several DCT transform sizes are supported in the transform process: 32, 16, 8 and 4. The transform size is flexible and is decided 110A according to the characteristics of the input data. The input residual image 97 is first divided into 32x32 macroblocks. The transform size is decided for each macroblock as a function of its activity level in the pixel domain as described below. Then the transform is applied 110B, which provides an image of DCT blocks 111 according to an initial segmentation. The transforms used are the 4x4, 8x8, 16x16 and 32x32 DCT, as defined in the HEVC standard.

The next coding step comprises computing, by channel modelling 110C, a statistical model of each DCT channel 112. A DCT channel consists of the set of values taken by samples from all image blocks at same DCT coefficient position, for a given block type. Indeed, a variety of block types can be implemented as described below to segment the image accordingly and provide better encoding.

DCT coefficients for each block type are modelled by a Generalized Gaussian Distribution (GGD) as described below. For such a distribution, each DCT channel is assigned a quantizer. This non-uniform scalar quantizer 113 is defined by a set of quantization intervals and associated de-quantized sample values. A pool of such quantizers 114 is available on both the encoder and on the decoder side. Various quantizers are pre-computed off-line, through the Chou-Lookabaugh-Gray rate distortion optimization process described below.

The selection of the rate distortion optimal quantizer for a given DCT channel proceeds as follows. Given input coding parameters, a distortion target 115 is determined for the DCT channel under consideration. To do so, a distortion target allocation among various DCT channels, and among various block sizes, is performed. The distortion allocation ensures that each DCT channel of each block size should be encoded at level that corresponds to identical rate distortion slope among all coded DCT channels. This rate distortion slope depends on an input quality parameter, given by the user through use of merits as described below.

Once the distortion target 115 is obtained for each DCT channel, the right quantizer 113 to use is chosen 110D. As the rate distortion curve associated to each pre-computed quantizer is known (tabulated), this merely consists in choosing the quantizer that provides minimal bitrate for given distortion target.

Then DCT coefficients are quantized 110E to produce quantized DCT X_Q values 116, and entropy coded 110F to produce a set of values $H(X_Q)$ 117. As described below, an encoding cost competition process makes it possible to select the best segmentation of the residual enhancement image (in practice of each 64x64 large coding units or LCUs of the image) into blocks or coding units.

The entropy coder used consists of a simple, non-contextual, non-adaptive arithmetic coder. The arithmetic coding employs, for each DCT channel, a set of fixed probabilities, respectively associated to each pre-computed quantization interval. Therefore, these probabilities are entirely calculated off-line, together with the rate distortion optimal quantizers. Probability values are never updated during the encoding or decoding processes, and are fixed for the whole image being processed. In particular, this ensures the spatial random access feature, and also makes the decoding process highly parallelizable.

As a result of the proposed enhancement INTRA image coding scheme, the enhancement layer bit-stream is made of the following syntax elements for each INTRA image:

- parameters of each coded DCT channel model 99' (99 prime). Two parameters are needed to fully specify a generalized Gaussian distribution. Therefore, two parameters are sent for each encoded DCT channel. These are sent only once for each image. An embodiment of the invention provides for prediction or reuse of such parameters from one residual enhancement INTRA image to the other, as described below with reference to **Figures 24**.

- chosen block types 118 are arithmetic encoded 110F. Generally, the block types segmenting the image are stored as a quad-tree ("block type quad-tree") which is then encoded. The probabilities used for their arithmetic coding are computed during the transform sizes selection, are quantized and fixed-length coded into the output bit-stream. These probabilities may be fixed for the whole frame or slice. In an embodiment described below, these probabilities are function of probabilities on block types in the corresponding base layer.

- coded residual data 99 results from the entropy coding of quantized DCT coefficients.

Note that the above syntax elements represent the content of coded slice data in the scalable extension of HEVC. The NAL unit container of HEVC can be used to encapsulate a slice that is coded according to the coding scheme of **Figure 11**.

Figure 12 depicts the enhancement INTRA image decoding process 120 which corresponds to the encoding process illustrated in **Figure 11**. The input to the decoder consists in the enhancement layer bit-stream 99 (coded residual data and coded block type quad-tree) and the parametric model of DCT channels 99' (99 prime), for the input residual enhancement image 97.

First, following a process similar to that effected in the encoder, the decoder determines the distortion target 115 of each DCT channel, given the parametric model of each coded DCT channel 99' (99 prime). Then, the choice of optimal quantizers (or quantifiers) 110D for each DCT channel is performed exactly in the same way as on the encoder side. Given the chosen quantizers 113, and thus probabilities of all quantized DCT symbols, the arithmetic decoder is able to decode the input coded residual data 99 using the decoded block type quad-tree to know the association between each block and corresponding DCT channel. This provides successive quantized DCT blocks, which are then inverse quantized 120A and inverse transformed 120B. The transform size of each DCT block is obtained from the decoded block types.

The encoding of the residual enhancement image X for an enhancement INTRA image is now described. As explained in more details below, it is proposed to determine an initial segmentation of the image to be encoded, then to change this segmentation in order to optimize an encoding cost and to use the optimizing segmentation for encoding.

The main steps of this optimized encoding process are now described one by one, before a presentation of the whole process is given with reference to **Figure 22**.

Conventionally, the residual enhancement image is to be transformed, using for example a DCT transform, to obtain an image of transformed block coefficients, for example an image made of a plurality of DCT blocks, each comprising DCT coefficients.

As an example, the residual enhancement image may be divided by the initial segmentation just mentioned into blocks B_k, each having a particular block type. Several block types may be considered, owing in particular to various possible sizes for the block. Other parameters than size may be used to distinguish between block types.

In particular, as there may be a big disparity of activity (or energy) between blocks with the same size, a segmentation of an image by using only block size is not fine enough to obtain an optimal performance of classification of parts of the image.

This is why it is proposed to add a label to the block size in order to distinguish various levels and/or characteristics of a block activity.

It is proposed for instance to use only square blocks, here blocks of dimensions 32x32, 16x16 and 8x8, and the following block types for luminance residual images, each block type being defined by a size and a label (corresponding to an index of energy for instance, but possibly also to other parameters as explained below):

- 32x32 label 1;
- 32x32 label 2;
- *etc.*
- 32x32 label N_{32};
- 16x16 label 1 (*e.g.* low);
- 16x16 label 2;
- *etc.*;
- 16x16 label N_{16};
- 8x8 label 1 (*e.g.* low);
- 8x8 label 2;
- *etc.*;
- 8x8 label N_8 (e.g. high).

In addition, a further block type may be introduced for each block size, with a label "*skip*" meaning that the corresponding block of data is not encoded and that corresponding residual pixels, or equivalently DCT coefficients, are considered to have a null value (value zero). It is however proposed here not to use these types with skip-label in the initial segmentation, but to introduce them during the segmentation optimisation process, as described below.

There are thus N_{32}+1 block types of size 32x32, N_{16}+1 block types of size 16x16 and N_8+1 block types of size 8x8. The choice of the parameters N_{32}, N_{16}, N_8 depends on the residual image content and, as a general rule, high quality coding requires more block types than low quality coding.

For the initial segmentation, the choice of the block size is performed here by computing the L_2 integral I of a morphological gradient (measuring residual activity, *e.g.* residual morphological activity) on each 32x32 block, before applying the DCT transform. (Such a morphological gradient corresponds to the difference between a dilatation and an erosion of the luminance residual image, as explained for instance in "*Image Analysis and Mathematical Morphology*", Vol. 1, by Jean Serra, Academic Press, February 11, 1984.) If the integral computed for a block is higher than a

predetermined threshold, the concerned block is divided into four smaller, here 16x16-, blocks; this process is applied on each obtained 16x16 block to decide whether or not it is divided into 8x8 blocks (top-down algorithm).

Once the block size of a given block is decided, the block type of this block is determined based on the morphological integral computed for this block, for instance here by comparing the morphological integral I with thresholds defining three or more bands of residual activity (*i.e.* three or more indices of energy or three or more labels as exemplified above) for each possible size (for example: bottom, low or normal residual activity for 16x16-blocks and low, normal, high residual activity for 8x8-blocks).

It may be noted that the morphological gradient is used in the present example to measure the residual activity but that other measures of the residual activity may be used, instead or in combination, such as local energy or Laplace's operator.

In a possible embodiment, the decision to attribute a given label to a particular block (once its size is determined as above) may be based not only on the magnitude of the integral I, but also on the ratio of vertical activity vs. horizontal activity, *e.g.* thanks to the ratio I_h/I_v, where I_h is the L_2 integral of the horizontal morphological gradient and I_v is the L_2 integral of the vertical morphological gradient. In other words, the initial segmentation is based on block activity along several spatial orientations

For instance, the concerned block will be attributed a label (*i.e.* a block type) depending on whether the ratio I_h/I_v is below 0.5 (corresponding to a block with residual activity oriented in the vertical direction), between 0.5 and 2 (corresponding to a block with non-oriented residual activity) and above 2 (corresponding to a block with residual activity oriented in the horizontal direction).

It is proposed here that chrominance blocks each have a block type inferred from the block type of the corresponding luminance block in the image. For instance chrominance block types can be inferred by dividing in each direction the size of luminance block types by a factor depending on the resolution ratio between the luminance and the chrominance.

The same segmentation is thus used for the three image components, namely the chrominance components U and V, and the luminance component Y.

Practically, a block type NxN label L for a macroblock underlies the following inference for each image component:

- block type NxN label L_Y for the block NxN in Y;

- block type N/2xN/2 label L_U for the block N/2xN/2 in U;
- block type N/2xN/ label L_V for the block N/2xN/2 in V.

A subscript of the component name has been added to the label because, as we will see later, the coding depends also on the image component. For instance, the coding of NxN2 label L_Y is not the same as the coding of N/2xN/2 label L_U since associated quantizers may differ. Similarly, the coding of N/2xN/2 label L_U differs from the coding of N/2xN/2 label L_V.

In an example where use is made of 4:2:0 videos, where chrominance (U and V) images are down-sampled by a factor two both vertically and horizontally compared to the corresponding luminance image, blocks in chrominance images have a size (among 16x16, 8x8 and 4x4) and a label both inferred from the size and label of the corresponding block in the luminance image.

In addition, it is proposed here as just explained to define the block type in function of its size and an index of the energy, also possibly considering orientation of the residual activity. Other characteristics can also be considered such as for example the encoding mode used for the co-located or "spatially corresponding" block of the base layer, referred below as to the "*base coding mode*". Typically, Intra blocks of the base layer 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 (*i.e.* Skipped blocks).

Figure 13 shows an exemplary process for determining optimal quantizers (based on a given segmentation, *e.g.* the initial segmentation or a modified segmentation during the optimising process) focusing on steps performed at the block level.

Once a segmentation is determined, including the definition of a block type associated to each block (steps S2), a DCT transform is then applied to each of the concerned blocks (step S4) in order to obtain a corresponding block of DCT coefficients.

Within a block, the DCT coefficients are associated with an index i (*e.g.* i = 1 to 64), following an ordering used for successive handling when encoding, for example.

Blocks are grouped into macroblocks MB_k. A very common case for so-called 4:2:0 YUV video streams is a macroblock made of 4 blocks of luminance Y, 1 block of chrominance U and 1 block of chrominance V. Here too, other configurations may be considered.

To simplify the explanations, only the coding of the luminance component is described here with reference to **Figure 13**. However, the same approach can be used for coding the chrominance components. In addition, it will be further explained with reference to **Figures 21A** and **21B** how to process luminance and chrominance in relation with each other.

Starting from the image X in the frequency domain (i.e. made of DCT blocks having block types), a probabilistic distribution P of each DCT coefficient is determined using a parametric probabilistic model at step S6. This is referenced 110C in **Figure 11**.

Since, in the present example, the image X is a residual image, i.e. information is about a noise residual, it is efficiently modelled by Generalized Gaussian Distributions (GGD) having a zero mean: $\mathrm{DCT}(X) \approx GGD(\alpha, \beta)$,

where α,β are two parameters to be determined and the GGD follows the following two-parameter distribution: $GGD(\alpha, \beta, x) := \frac{\beta}{2\alpha\Gamma(1/\beta)} \exp(-|x/\alpha|^{\beta})$,

and where Γ is the well-known Gamma function: $\Gamma(z) = \int_0^{\infty} t^{z-1} e^{-t} dt$.

The DCT coefficients cannot be all modelled by the same parameters and, practically, the two parameters α, β depend on:

- the video content. This means that the parameters must be computed for each image or for every group of n images for instance;
- the index i of the DCT coefficient within a DCT block B_k. Indeed, each DCT coefficient has its own behaviour. A DCT channel is thus defined for the DCT coefficients co-located (i.e. having the same index) within a plurality of DCT blocks (possibly all the blocks of the image). A DCT channel can therefore be identified by the corresponding coefficient index *i* for a given block type *k*. For illustrative purposes, if the residual enhancement image X is divided into 8x8 pixel blocks, the modelling 110C has to determine the parameters of 64 DCT channels for each base coding mode.
- the block type defined above. The content of the image, and then the statistics of the DCT coefficients, may be strongly related to the block type because, as explained above, the block type is selected in function of the image content, for instance to use large blocks for parts of the image containing little information.

In addition, since the luminance component Y and the chrominance components U and V have dramatically different source contents, they must be

encoded in different DCT channels. For example, if it is decided to encode the luminance component Y on one channel and to encode jointly the chrominance components UV on another channel, 64 channels are needed for the luminance of a block type of size 8x8 and 16 channels are needed for the joint UV chrominance (made of 4x4 blocks) in a case of a 4:2:0 video where the chrominance is down-sampled by a factor two in each direction compared to the luminance. Alternatively, one may choose to encode U and V separately and 64 channels are needed for Y, 16 for U and 16 for V.

At least 64 pairs of parameters for each block type may appear as a substantial amount of data to transmit to the decoder (parameter bit-stream 99'). However, experience proves that this is quite negligible compared to the volume of data needed to encode the residuals of Ultra High Definition (4k2k or more) videos. As a consequence, one may understand that such a technique is preferably implemented on large videos, rather than on very small videos because the parametric data would take too much volume in the encoded bit-stream.

In embodiments as described below with reference to **Figures 24**, some channel parameters are reused from one residual enhancement INTRA image to the other, thus drastically reducing the amount of such data to transmit.

For sake of simplicity of explanation, a set of DCT blocks corresponding to the same block type is now considered. The process will be reiterated for each block type.

To obtain the two parameters α_i, β_i defining the probabilistic distribution P_i for a DCT channel i, the Generalized Gaussian Distribution model is fitted onto the DCT block coefficients of the DCT channel, i.e. the DCT coefficients co-located within the DCT blocks of the same block type. Since this fitting is based on the values of the DCT coefficients, the probabilistic distribution is a statistical distribution of the DCT coefficients within a considered channel i.

For example, the fitting may be simply and robustly obtained using the moment of order k of the absolute value of a GGD:

$$M_k^{\alpha_i,\beta_i} := E\left(\left|GGD(\alpha_i,\beta_i)\right|^k\right)_{(k\in R_+)}$$

$$= \int_{-\infty}^{\infty} |x|^k \, GGD(\alpha_i,\beta_i,x)\,dx = \frac{{\alpha_i}^k \Gamma((1+k)/\beta_i)}{\Gamma(1/\beta_i)} \cdot$$

Determining the moments of order 1 and of order 2 from the DCT coefficients of channel i makes it possible to directly obtain the value of parameter β_i:

$$\frac{M_2}{(M_1)^2} = \frac{\Gamma(1/\beta_i)\Gamma(3/\beta_i)}{\Gamma(2/\beta_i)^2}$$

The value of the parameter β_i can thus be estimated by computing the above ratio of the two first and second moments, and then the inverse of the above function of β_i.

Practically, this inverse function may be tabulated in memory of the encoder instead of computing Gamma functions in real time, which is costly.

The second parameter α_i may then be determined from the first parameter β_i and the second moment, using the equation: $M_2 = \sigma^2 = \alpha_i^2 \Gamma(3/\beta_i)/\Gamma(1/\beta_i)$.

The two parameters α_i, β_i being determined for the DCT coefficients i, the probabilistic distribution P_i of each DCT coefficient i is defined by

$$P_i(x) = GGD(\alpha_i, \beta_i, x) = \frac{\beta_i}{2\alpha_i \Gamma(1/\beta_i)} \exp(-|x/\alpha_i|^{\beta_i}).$$

Referring to **Figure 11**, a quantization 110E of the DCT coefficients is to be performed in order to obtain quantized symbols or values. As explained below, it is proposed here to first determine a quantizer per DCT channel so as to optimize a rate-distortion criterion.

Figure 14 illustrates an exemplary Voronoi cell based quantizer.

A quantizer is made of M Voronoi cells distributed along the values of the DCT coefficients. Each cell corresponds to an interval $[t_m, t_{m+1}]$, called quantum Q_m.

Each cell has a centroid c_m, as shown in the Figure.

The intervals are used for quantization: a DCT coefficient comprised in the interval $[t_m, t_{m+1}]$ is quantized to a symbol a_m associated with that interval.

For their part, the centroids are used for de-quantization: a symbol a_m associated with an interval is de-quantized into the centroid value c_m of that interval.

The quality of a video or still image may be measured by the so-called Peak-Signal-to-Noise-Ratio or PSNR, which is dependent upon a measure of the L2-norm of the error of encoding in the pixel domain, *i.e.* the sum over the pixels of the squared difference between the original pixel value and the decoded pixel value. It may be recalled in this respect that the PSNR may be expressed in dB as: $10.\log_{10}(\frac{MAX^2}{MSE})$, where MAX is the maximal pixel value (in the spatial domain) and

MSE is the mean squared error (*i.e.* the above sum divided by the number of pixels concerned).

However, as noted above, most of video codecs compress the data in the DCT-transformed domain in which the energy of the signal is much better compacted.

The direct link between the PSNR and the error on DCT coefficients is now explained.

For a residual block, we note ψ_n its inverse DCT (or IDCT) pixel base in the pixel domain as shown on **Figure 15**. If one uses the so-called IDCT III for the inverse transform, this base is orthonormal: $\|\psi_n\| = 1$.

On the other hand, in the DCT domain, the unity coefficient values form a base φ_n which is orthogonal. One writes the DCT transform of the pixel block X as follows: $X_{DCT} = \sum_n d^n \varphi_n$,

where d^n 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:

$$X = \mathrm{IDCT}(X_{DCT}) = \mathrm{IDCT}\sum_n d^n \varphi_n = \sum_n d^n \mathrm{IDCT}(\varphi_n) = \sum_n d^n \psi_n .$$

If the value of the de-quantized coefficient d^n after decoding is denoted d_Q^n, one sees that (by linearity) the pixel error block is given by : $\varepsilon_X = \sum_n (d^n - d_Q^n)\psi_n$

The mean L_2-norm error on all blocks, is thus:

$$E\left(\|\varepsilon_X\|_2^2\right) = E\left(\sum_n \left|d^n - d_Q^n\right|^2\right) = \sum_n E\left(\left|d^n - d_Q^n\right|^2\right) = \sum_n D_n^2$$

where D_n^2 is the mean quadratic error of quantization on the n-th DCT coefficient, or squared distortion for this type of coefficient. The distortion is thus a measure of the distance between the original coefficient (here the coefficient before quantization) and the decoded coefficient (here the dequantized coefficient).

It is thus proposed below to control the video quality by controlling the sum of the quadratic errors on the DCT coefficients. In particular, this control is preferable compared to the individual control of each of the DCT coefficient, which is *a priori* a sub-optimal control.

In the embodiment described here, it is proposed to determine (*i.e.* to select in step 110D of **Figure 11**) a set of quantizers (to be used each for a corresponding DCT channel), the use of which results in a mean quadratic error having a target value D_t^2 while minimising the rate obtained. This corresponds to step S16 in **Figure 13**.

In view of the above correspondence between PSNR and the mean quadratic error D_n^2 on DCT coefficients, these constraints can be written as follows:

$$\text{minimize } R = \sum_n R_n(D_n) \quad \text{s.t.} \quad \sum_n D_n^2 = D_t^2 \text{ (A)}$$

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

It may be noted that the above minimization problem (A) may only be fulfilled by optimal quantizers which are solution of the problem

$$\text{minimize } R_n(D_n) \quad \text{s.t.} \quad E\left(\left|d^n - d_Q^n\right|^2\right) = D_n^2 \text{ (B).}$$

This statement is simply proven by the fact that, assuming a first quantizer would not be optimal following (B) but would fulfil (A), then a second quantizer with less rate but the same distortion can be constructed (or obtained). So, if one uses this second quantizer, the total rate R has been diminished without changing the total distortion $\sum_n D_n^2$; this is in contradiction with the first quantizer being a minimal solution of the problem (A).

As a consequence, the rate-distortion minimization problem (A) can be split into two consecutive sub-problems without losing the optimality of the solution:

- first, determining optimal quantizers and their associated rate-distortion curves $R_n(D_n)$ following the problem (B), which will be done in the present case for GGD channels as explained below;

- second, by using optimal quantizers, the problem (A) is changed into the problem (A_opt):

$$\text{minimize } R = \sum_n R_n(D_n) \quad \text{s.t.} \quad \sum_n D_n^2 = D_t^2 \text{ and } R_n(D_n) \text{ is optimal} \quad \text{(A_opt).}$$

Based on this analysis, it is proposed as further explained below:

- to compute off-line (step S8 in **Figure 13**) optimal quantizers adapted to possible probabilistic distributions of each DCT channel (thus resulting in the pool 114

of quantizers of **Figure 11**). The same pool 114 is generally used for all the block types occurring in the image (or in the video);

- to select (step S16) one of these pre-computed optimal quantizers for each DCT channel (*i.e.* each type of DCT coefficient) such that using the set of selected quantizers results in a global distortion corresponding to the target distortion D_t^2 with a minimal rate (*i.e.* a set of quantizers which solves the problem A_opt).

It is now described a possible embodiment for the first step S8 of computing optimal quantizers for possible probabilistic distributions, here Generalised Gaussian Distributions.

It is proposed to change the previous complex formulation of problem (B) into the so-called Lagrange formulation of the problem: for a given parameter $\lambda > 0$, we determine the quantization in order to minimize a cost function such as $D^2 + \lambda R$. We thus get an optimal rate-distortion couple (D_λ, R_λ). In case of a rate control (*i.e.* rate minimisation) for a given target distortion Δ_t, the optimal parameter $\lambda > 0$ is determined by $\lambda_{\Delta_t} = \underset{\lambda,\, D_\lambda \le \Delta_t}{\arg\min} R_\lambda$ (*i.e.* the value of λ for which the rate is minimum while fulfilling the constraint on distortion) and the associated minimum rate is $R_{\Delta_t} = R_{\lambda_{\Delta_t}}$.

As a consequence, by solving the problem in its Lagrange formulation, for instance following the method proposed below, it is possible to plot a rate distortion curve associating a resulting minimum rate to each distortion value ($\Delta_t \mapsto R_{\Delta_t}$) which may be computed off-line as well as the associated quantization, *i.e.* quantizer, making it possible to obtain this rate-disortion pair.

It is precisely proposed here to formulate problem (B) into a continuum of problems (B_lambda) having the following Lagrange formulation

$$\text{minimize } D_n^2 + \lambda R_n(D_n) \quad \text{s.t.} \quad E\left(\left|x - d_m\right|^2\right) = D_n^2 \quad \text{(B_lambda)}.$$

The well-known Chou-Lookabaugh-Gray algorithm is a good practical way to perform the required minimisation. It may be used with any distortion distance d ; we describe here a simplified version of the algorithm for the L^2-distance. This is an iterative process from any given starting guessed quantization.

As noted above, this algorithm is performed here for each of a plurality of possible probabilistic distributions (in order to obtain the pre-computed optimal quantizers for the possible distributions to be encountered in practice), and for a

plurality of possible numbers M of quanta. It is described below when applied for a given probabistic distribution P and a given number M of quanta.

In this respect, as the parameter alpha α (or equivalently the standard deviation σ of the Generalized Gaussian Definition) can be moved out of the distortion parameter D_n^2 because it is a homothetic parameter, only optimal quantizers with unity standard deviation $\sigma = 1$ need to be determined in the pool of quantizers.

Taking advantage of this remark, in the proposed embodiment, the GGD representing a given DCT channel will be normalized before quantization (*i.e.* homothetically transformed into a unity standard deviation GGD), and will be de-normalized after de-quantization. Of course, this is possible because the parameters (in particular here the parameter α or equivalently the standard deviation σ) of the concerned GGD model are sent to the decoder in the video bit-stream 99'.

Before describing the algorithm itself, the following should be noted.

The position of the centroids c_m is such that they minimize the distortion δ_m^2 inside a quantum, in particular one must verify that $\partial_{c_m} \delta_m^2 = 0$ (as the derivative is zero at a minimum).

As the distortion δ_m of the quantization, on the quantum Q_m, is the mean error $E(d(x;c_m))$ for a given distortion function or distance d, the distortion on one quantum when using the L^2-distance is given by $\delta_m^2 = \int_{Q_m} |x - c_m|^2 P(x)\mathrm{d}x$ and the nullification of the derivative thus gives: $c_m = \int_{Q_m} xP(x)\mathrm{d}x / P_m$, where P_m is the probability of x to be in the quantum Q_m and is simply the following integral $P_m = \int_{Q_m} P(x)\mathrm{d}x$.

Turning now to minimisation of the cost function $C = D^2 + \lambda R$, and considering that the rate reaches the entropy of the quantized data: $R = -\sum_{m=1}^{M} P_m \log_2 P_m$, the nullification of the derivatives of the cost function for an optimal solution can be written as:

$$0 = \partial_{t_{m+1}} C = \partial_{t_{m+1}} \left[\Delta_m^2 - \lambda P_m \ln P_m + \Delta_{m+1}^2 - \lambda P_{m+1} \ln P_{m+1} \right]$$

Let us set $\overline{P} = P(t_{m+1})$ the value of the probability distribution at the point t_{m+1}. From simple variational considerations, see **Figure 16**, we get

$\partial_{t_{m+1}} P_m = \overline{P}$ and $\partial_{t_{m+1}} P_{m+1} = -\overline{P}$.

Then, a bit of calculation leads to

$$\begin{aligned}\partial_{t_{m+1}} \Delta_m^2 &= \partial_{t_{m+1}} \int_m^{t_{m+1}} |x - c_m|^2 P(x)\mathrm{d}x \\ &= \overline{P}|t_{m+1} - c_m|^2 + \int_m^{t_{m+1}} \partial_{t_{m+1}} |x - c_m|^2 P(x)\mathrm{d}x \\ &= \overline{P}|t_{m+1} - c_m|^2 - 2\partial_{t_{m+1}} c_m \int_m^{t_{m+1}} (x - c_m) P(x)\mathrm{d}x \\ &= \overline{P}|t_{m+1} - c_m|^2\end{aligned}$$

as well as

$$\partial_{t_{m+1}} \Delta_{m+1}^2 = -\overline{P}|t_{m+1} - c_{m+1}|^2 .$$

As the derivative of the cost is now explicitly calculated, its cancellation gives: $0 = \overline{P}|t_{m+1} - d_m|^2 - \lambda\overline{P}\ln P_m - \lambda P_m \frac{\overline{P}}{P_m} - \overline{P}|t_{m+1} - d_{m+1}|^2 + \lambda\overline{P}\ln P_{m+1} + \lambda P_m \frac{\overline{P}}{P_m}$,

which leads to a useful relation between the quantum boundaries t_m, t_{m+1} and the centroids c_m : $t_{m+1} = \frac{c_m + c_{m+1}}{2} - \lambda \frac{\ln P_{m+1} - \ln P_m}{2(c_{m+1} - c_m)}$.

Thanks to these formulae, the Chou-Lookabaugh-Gray algorithm can be implemented by the following iterative process:

1. Start with arbitrary quanta Q_m defined by a plurality of limits t_m
2. Compute the probabilities P_m by the formula $P_m = \int_{Q_m} P(x)\mathrm{d}x$
3. Compute the centroids c_m by the formula $c_m = \int_{Q_m} xP(x)\mathrm{d}x / P_m$
4. Compute the limits t_m of new quanta by the formula

$$t_{m+1} = \frac{c_m + c_{m+1}}{2} - \lambda \frac{\ln P_{m+1} - \ln P_m}{2(c_{m+1} - c_m)}$$

5. Compute the cost $C = D^2 + \lambda R$ by the formula $C = \sum_{m=1}^{M} \Delta_m^2 - \lambda P_m \ln P_m$
6. Loop to 2. until convergence of the cost C

When the cost C has converged (i.e. C varies by an amount less than a minimum threshold, from one iteration to the next one), the current values of limits t_m and centroids c_m define a quantization, *i.e.* a quantizer, with M quanta, which solves

the problem (B_lambda), *i.e.* minimises the cost function for a given value λ, and has an associated rate value R_λ and an distortion value D_λ.

Such a process is implemented for many values of the Lagrange parameter λ (for instance 100 values comprised between 0 and 50). It may be noted that for λ equal to 0, there is no rate constraint, which corresponds to the so-called Lloyd quantizer.

In order to obtain optimal quantizers for a given parameter β of the corresponding GGD, the problems (B_lambda) are to be solved for various odd (by symmetry) values of the number M of quanta and for the many values of the parameter λ. A rate-distortion diagram for the optimal quantizers with varying M is thus obtained, as shown on **Figure 17**.

It turns out that, for a given distortion, there is an optimal number M of needed quanta for the quantization associated to an optimal parameter λ. In brief, one may say that optimal quantizers of the general problem (B) are those associated to a point of the upper envelope of the rate-distortion curves making this diagram, each point being associated with a number of quanta (*i.e.* the number of quanta of the quantizer leading to this point of the rate-distortion curve). This upper envelope is illustrated on **Figure 18**. At this stage, we have now lost the dependency on λ of the optimal quantizers: for a given rate (or a given distortion) corresponds only one optimal quantizer whose number of quanta M is fixed.

Based on observations that the GGD modelling provides a value of β almost always between 0.5 and 2 in practice, and that only a few discrete values are enough for the precision of encoding, it is proposed here to tabulate β every 0.1 in the interval between 0.2 and 2.5. In another embodiment, β is tabulated every 0.2 in the interval [0.6, 2.0]; thus, there are eight values for β, each of which may thus be encoded on three bits through a table shared between the encoder and the decoder, when β is added to bit-stream 99'.

Considering these values of β (*i.e.* here for each of the 24 values of β taken in consideration between 0.2 and 2.5 or for each of the 8 values of β taken in consideration between 0.6 and 2.0), rate-distortion curves, depending on β, are obtained (step S10) as shown on **Figure 19**. It is of course possible to obtain according to the same process rate-distortion curves for a larger number of possible values of β.

Each curve may in practice be stored in the encoder (the same at the decoder) in a table containing, for a plurality of points on the curve, the rate and distortion (coordinates) of the point concerned, as well as features defining the associated quantizer (here the number of quanta and the values of limits t_m and centroids c_m for the various quanta). For instance, a few hundreds of quantizers may be stored for each β up to a maximum rate, *e.g.* of 5 bits per DCT coefficient, thus forming the pool 114 of quantizers mentioned in **Figure 11**. It may be noted that a maximum rate of 5 bits per coefficient in the enhancement layer makes it possible to obtain good quality in the decoded image. Generally speaking, it is proposed to use a maximum rate per DCT coefficient equal or less than 10 bits, for which value near lossless coding is provided.

Before turning to the selection of quantizers (step S16), for the various DCT channels and among these optimal quantizers stored in association with their corresponding rate and distortion when applied to the concerned distribution (GGD with a specific parameter β), it is proposed here to select which part of the DCT channels are to be encoded. Indeed, in a less optimal solution, every DCT channel is encoded.

Based on the observation that the rate decreases monotonously as a function of the distortion induced by the quantizer, precisely in each case in the manner shown by the curves just mentioned, it is possible to write the relationship between rate and distortion as follows: $R_n = f_n(-\ln(D_n / \sigma_n))$,

where σ_n is the normalization factor of the DCT coefficient, *i.e.* the GGD model associated to the DCT coefficient has σ_n for standard deviation, and where $f_n' \geq 0$ in view of the monotonicity just mentioned.

In particular, without encoding (equivalently zero rate) leads to a quadratic distortion of value σ_n^2 and we deduce that $0 = f_n(0)$.

Finally, one observes that the curves are convex for parameters β lower than two: $\beta \leq 2 \Rightarrow f_n'' \geq 0$.

It is proposed here to consider the merit of encoding a DCT coefficient. More encoding basically results in more rate R_n (in other words, the corresponding cost) and less distortion D_n^2 (in other words the resulting gain or advantage).

Thus, when dedicating a further bit to the encoding of the video (rate increase), it should be determined on which DCT coefficient this extra rate is the most

efficient. In view of the analysis above, an estimation of the merit M of encoding may be obtained by computing the ratio of the benefit on distortion to the cost of encoding:

$$M_n := \left| \frac{\Delta D_n^{\,2}}{\Delta R_n} \right|.$$

Considering the distortion decreases by an amount ε, then a first order development of distortion and rates gives

$$(D-\varepsilon)^2 = D^2 - 2\varepsilon D + o(\varepsilon)$$

and

$$\begin{aligned} R(D-\varepsilon) &= f_n(-\ln((D-\varepsilon)/\sigma)) \\ &= f_n(-\ln(D/\sigma) - \ln(1-\varepsilon/D)) \\ &= f_n(-\ln(D/\sigma) + \varepsilon/D + o(\varepsilon)) \\ &= f_n(-\ln(D/\sigma)) + \varepsilon f_n'(-\ln(D/\sigma))/D \end{aligned}$$

As a consequence, the ratio of the first order variations provides an explicit formula for the merit of encoding: $M_n(D_n) = \dfrac{2D_n^2}{f_n'(-\ln(D_n/\sigma_n))}$.

If the initial coefficient encoding merit or "initial merit" M_n^0 is defined as the merit of encoding at zero rate, *i.e.* before any encoding, this initial merit M_n^0 can thus be expressed as follows using the preceding formula: $M_n^0 := M_n(\sigma_n) = \dfrac{2\sigma_n^2}{f_n'(0)}$ (because as noted above no encoding leads to a quadratic distortion of value σ_n^2). That is determining an initial coefficient encoding merit for a given coefficient type includes estimating a ratio between a distortion variation provided by encoding a coefficient having the given type and a rate increase resulting from encoding said coefficient.

It is thus possible, starting from the pre-computed and stored rate-distortion curves, to determine the function f_n associated with a given DCT channel and to compute the initial merit M_n^0 of encoding the corresponding DCT coefficient (the value $f_n'(0)$ being determined by approximation thanks to the stored coordinates of rate-distortion curves).

It may further be noted that, for β lower than two (which is in practice almost always true), the convexity of the rate distortion curves teaches us that the merit is an increasing function of the distortion.

In particular, the initial merit is thus an upper bound of the merit: $M_n(D_n) \leq M_n^0$.

It will now be shown that, when satisfying the optimisation criteria defined above, all encoded DCT coefficients in the block have the same merit after encoding. Furthermore, this does not only apply to one block only, but as long as the various functions f_n used in each DCT channel are the unchanged, *i.e.* in particular for all blocks in a given block type. Hence the common merit value for encoded DCT coefficients will now be referred to as the merit of the block type or "block merit".

The above property of equal merit after encoding may be shown for instance using the Karush-Kuhn-Tucker (KKT) necessary conditions of optimality. In this goal, the quality constraint $\sum_n D_n^2 = D_t^2$ can be rewritten as $h = 0$ with $h(D_1, D_2, \ldots) := \sum_n D_n^2 - D_t^2$.

The distortion of each DCT coefficient is upper bounded by the distortion without coding: $D_n \leq \sigma_n$, and the domain of definition of the problem is thus a multi-dimensional box $\Omega = \{(D_1, D_2, \ldots); D_n \leq \sigma_n\} = \{(D_1, D_2, \ldots); g_n \leq 0\}$, defined by the functions $g_n(D_n) := D_n - \sigma_n$.

Thus, the problem can be restated as follows:

$\text{minimize } R(D_1, D_2, \ldots)$ s.t. $h = 0, g_n \leq 0$ (A_opt').

Such an optimization problem under inequality constrains can effectively be solved using so-called Karush-Kuhn-Tucker (KKT) necessary conditions of optimality.

In this goal, the relevant KKT function Λ is defined as follows:

$$\Lambda(D_1, D_2, \ldots, \lambda, \mu_1, \mu_2, \ldots) := R - \lambda h - \sum_n \mu_n g_n .$$

The KKT necessary conditions of minimization are

- stationarity: $d\Lambda = 0$,
- equality: $h = 0$,
- inequality: $g_n \leq 0$,
- dual feasibility: $\mu_n \geq 0$,

- saturation: $\mu_n g_n = 0$.

It may be noted that the parameter λ in the KKT function above is unrelated to the parameter λ used above in the Lagrange formulation of the optimization problem meant to determine optimal quantizers.

If $g_n = 0$, the n-th condition is said to be saturated. In the present case, it indicates that the n-th DCT coefficient is not encoded.

By using the specific formulation $R_n = f_n(-\ln(D_n/\sigma_n))$ of the rate depending on the distortion discussed above, the stationarity condition gives:

$$0 = \partial_{D_n}\Lambda = \partial_{D_n}R_n - \lambda\partial_{D_n}h - \mu_n\partial_{D_n}g_n = -f_n'/D_n - 2\lambda D_n - \mu_n,$$

i.e. $2\lambda D_n^2 = -\mu_n D_n - f_n'$.

By summing on n and taking benefit of the equality condition, this leads to

$$2\lambda D_t^2 = -\sum_n \mu_n D_n - \sum_n f_n'. \quad (*)$$

In order to take into account the possible encoding of part of the coefficients only as proposed above, the various possible indices n are distributed into two subsets:

- the set $I^0 = \{n; \mu_n = 0\}$ of non-saturated DCT coefficients (*i.e.* of encoded DCT coefficients) for which we have $\mu_n D_n = 0$ and $D_n^2 = -f_n'/2\lambda$, and

- the set $I^+ = \{n; \mu_n > 0\}$ of saturated DCT coefficients (i.e. of DCT coefficients not encoded) for which we have $\mu_n D_n = -f_n' - 2\lambda\sigma_n^2$.

From (*), we deduce

$$2\lambda D_t^2 = -\sum_{I^+} \mu_n D_n - \sum_n f_n' = \sum_{I^+} f_n' + 2\lambda\sum_{I^+}\sigma_n^2 - \sum_n f_n'$$

and by gathering the λ's

$$2\lambda\left(D_t^2 - \sum_{I^+}\sigma_n^2\right) = \sum_{I^0} f_n'.$$

As a consequence, for a non-saturated coefficient $(n \in I^0)$, *i.e.* a coefficient to be encoded, we obtain:

$$D_n^2 = \left(D_t^2 - \sum_{I^+}\sigma_n^2\right) f_n'(-\ln(D_n/\sigma_n)) / \sum_{m\in I^0} f_m'(-\ln(D_m/\sigma_m)).$$

This formula for the distortion makes it possible to rewrite the above formula giving the merit $M_n(D_n)$ as follows for non-saturated coefficients:

$$M_n(D_n) = 2.\left(D_t^2 - \sum_{I^+} \sigma_n^2 \right) / \sum_{m \in I^0} f_m'(-\ln(D_m / \sigma_m))$$

Clearly, the right side of the equality does not depend on the DCT channel n concerned. Thus, for a block type k, for any DCT channel n for which coefficients are encoded, the merit associated with said channel after encoding is the same: $M_n = m_k$.

Another proof of the property of common merit after encoding is the following: supposing that there are two encoded DCT coefficients with two different merits M1 < M2, if an infinitesimal amount of rate from coefficient 1 is put on coefficient 2 (which is possible because coefficient 1 is one of the encoded coefficients and this does not change the total rate), the distortion gain on coefficient 2 would then be strictly bigger than the distortion loss on coefficient 1 (because M1 < M2). This would thus provide a better distortion with the same rate, which is in contradiction with the optimality of the initial condition with two different merits.

As a conclusion, if the two coefficients 1 and 2 are encoded and if their respective merits M1 and M2 are such that M1 < M2, then the solution is not optimal.

Furthermore, all non-coded coefficients have a merit smaller than the merit of the block type (*i.e.* the merit of coded coefficients after encoding).

In view of the property of equal merits of encoded coefficients when optimisation is satisfied, it is proposed here to encode only coefficients for which the initial encoding merit $M_n^0 = \frac{2\sigma_n^2}{f_n'(0)}$ is greater than a predetermined target block merit m_k.

The DCT coefficients with an initial encoding merit M_n^0 lower than the predetermined target block merit m_k are not encoded. In other words, all non-encoded coefficients (i.e. I^+) have a merit smaller than the merit of the block type.

In practice, for each coefficient type and each block type, at least one parameter (β) representative of a probabilistic distribution of coefficients having the concerned coefficient type in the concerned block type is determined; and the initial

coefficient encoding merit for given coefficient type and block type is determined based on the parameter for the given coefficient type and block type.

Then, for each coefficient to be encoded, a quantizer is selected to obtain the target block merit as the merit of the coefficient after encoding: first, the corresponding distortion, which is thus such that $M_n(D_n) = \frac{2D_n^2}{f_n'(-\ln(D_n/\sigma_n))} = m_k$, can be found by dichotomy using stored rate-distortion curves (step S14); the quantizer associated (see steps S8 and S10 above) with the distortion found is then selected (step S16).

Figure 20 illustrates such a stored merit-distortion curve for coefficient *n*.

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 D_n^2 such that $M_n(D_n) = M_{block}$.

Knowing the target block merit m_k for the block type *k*, it is possible to deduce a target distortion D_n from that curve associated with a particular DCT coefficient.

Then the parameter β of the DCT channel model for the considered DCT coefficient (obtained from the considered set of statistics) makes it possible to select one of the curves of **Figure 19**, for example the curve of **Figure 18**.

The target distortion D_n in that curve thus provides a unique optimal quantizer for DCT coefficient *n*, having M quanta Q_m.

That means that for each coefficient for which the initial coefficient encoding merit is greater than the predetermined block merit, a quantizer is selected depending on the coefficient probabilistic distribution parameter for the concerned coefficient type and block type and on the target block merit.

This is done for each DCT coefficient to encode (i.e. belonging to I^0) thus defining the quantizers for a block type.

Since, similar operations are performed for the other block types, the quantizers for all the block types can thus be fully selected.

This may be done for the luminance image Y and for the chrominance images U,V. In particular, in this process, the computation of DCT coefficients and GGD statistics is performed for the luminance image Y and for the chrominance images U,V (both using the same initial segmentation associating a block type to each block of the segmentation).

Figure 21 shows the process for determining optimal quantizers implemented in the present example at the level of the residual image, which includes in particular determining the target block merit for the various block types.

First, the image is segmented at step S30 into a plurality of blocks each having a given block type k, for instance in accordance with the process described above based on residual activity, or as a result of a change in the segmentation as explained below.

A parameter k designating the block type currently considered is then initialised at step S32.

The target block merit m_k for the block type k currently considered is the computed at step S34 based on a predetermined frame merit m^F and on a number of blocks v_k of the given block type per area unit, here according to the formula:

$$m_k = v_k . m^F$$

The frame merit m^F (m^Y below) for the luminance image Y is deduced from a user-specified QP parameter, as described below with reference to **Figure 21A**. Then the frame merits m^U, m^V for the chrominance images U, V are also derived from the user-specified QP parameter Y as explained below. Note that all the frame merits are derived from a video merit that is directly linked to the user-specified QP parameter

For instance, one may choose the area unit as being the area of a 16x16 block, *i.e.* 256 pixels. In this case, $v_k = 1$ for block types of size 16x16, $v_k = 4$ for block types of size 8x8 *etc*. One also understands that the method is not limited to square blocks; for instance $v_k = 2$ for block types of size 16x8.

This type of computation makes it possible to obtain a balanced encoding between block types, *i.e.* here a common merit of encoding per pixel (equal to the frame merit m^F) for all block types.

This is because the variation of the pixel distortion $\Delta\delta^2_{P,k}$ for the block type k is the sum $\sum_{codedn} \Delta D^2_{n,k}$ of the distortion variations provided by the various encoded DCT coefficients, and can thus be rewritten as follows thanks to the (common) block merit: $\Delta\delta^2_{P,k} = m_k . \sum_{codedn} \Delta R_{n,k} = m_k . \Delta R_k$ (where ΔR_k is the rate variation for a block of

type k). Thus, the merit of encoding per pixel is: $\frac{\Delta\delta_{P,k}^2}{\Delta U_k} = \frac{m_k.\Delta R_k}{v_k.\Delta R_k} = m^F$ (where U_k is the rate per area unit for the block type concerned) and has a common value over the various block types.

Optimal quantizers are then determined for the block type k currently considered by the process described above with reference to **Figure 13** using the data in blocks having the current block type k when computing parameters of the probabilistic distribution (GGD statistics) and using the block merit m_k just determined as the target block merit in step S14 of **Figure 13**.

The next block type is then considered by incrementing k (step S38), checking whether all block types have been considered (step S40) and looping to step S34 if all block types have not been considered.

If all block types have been considered, the whole residual image has been processed (step S42), which ends the encoding process at the image level presented here.

Figure 21A describes a process for deriving the frame merit m^Y for luminance component from a user-specified quality parameter. More precisely, this Figure illustrates a balancing of coding between INTRA images and INTER images, thus providing a final quality parameter QP_{final} for INTER coding and a Luma frame merit $M_{F,Y}$ for INTRA coding.

The process begins at step S50 where a user specifies merits λ_{video} for the video to encode, in particular a video merit $\lambda_{video}[layerId]$ for each layer composing the scalable video. Indeed a Luma frame merit $M_{F,Y}$ will be generated for a given layer (base or enhancement), meaning that different frame merits are obtained for different layers.

Step S52 consists in obtaining the index *layerId* of the layer to which the current image to encode belongs. Generally, base layer is indexed 0, while the enhancement layers are incrementally indexed from 1.

Step S52 is followed by step S54 where a video quality parameter QP_{video} is computed for the current layer *layerId* from the user-specified merits λ_{video}, as follows

$$QP_{video} = 3log_2(\lambda_{video}[layerId]) + 12$$

Next, at step S56, the position *PicIdx* of the current image within a GOP (see **Figure 3** or **4**) is determined. Generally, an INTRA image is given a position equal

to 0. Positions of the INTER images are 1 to 8 or to 16 depending on the considered coding structure.

Next, at step S58, a QP_{offset} for the current image in the considered layer is set to 0 for INTRA image. Note that this parameter QP_{offset} is used for INTER image only according to the formula shown on the Figure and described later with reference to **Figures 26A** to **26F**.

Next, at step S60, a quality parameter for the image is computed: $QP = QP_{video} + QP_{offset}$

It is followed by step S62 where a Lagrange parameter λ_{final} is computed as illustrated on the Figure. This is a usual step as known in the prior art, e.g. in HEVC, version HM-6.1.

Next, the step S64 makes it possible to handle differently INTRA images and INTER images.

If the current image is an INTRA image, the frame merit m^Y for luminance component is computed at step S66 according to the following formula:

$$m^Y = M_{F,Y} = \lambda_{I_scale} \times \lambda_{final}$$

where λ_{I_scale} represents a scaling factor used to balance the coding between enhancement INTRA and INTER images. This scaling factor may be fixed or user-specified and may depend on a spatial scalability ratio between base and enhancement layers.

If the current image is an INTER image, a different computation is performed at step S68 to obtain a final quality parameter QP_{final} (described later)

The process then ends at step S70.

Following the calculation of m^Y, the frame merits m^U, m^V for the chrominance components U, V of the residual enhancement INTRA image are derived. This is now described with reference to **Figure 21B**.

Figure 21B shows a process for determining optimal quantizers, which includes in particular determining the frame merits for each of chrominance components U,V for each image of the video sequence from the user-specified quality parameter. This Figure also provides an alternative way to compute the frame merit for the luminance component Y.

It is proposed in the present embodiment to consider the following video quality function:

$Q(R_Y,R_U,R_V)=PSNR_Y+\theta_U.PSNR_U+\theta_V.PSNR_V$,

where R* is the rate for the component * of an image, $PSNR_*$ is the PSNR for the component * of an image and θ_U, θ_V are balancing parameters provided by the user in order to select the acceptable degree of distortion in the concerned chrominance component (U or V) relative to the degree of distortion in the luminance component.

In order to unify the explanations in the various components, use is made below of $\theta_Y=1$ and the video quality function considered here can thus be rewritten as:

$Q(R_Y,R_U,R_V)= \theta_Y.PSNR_Y+\theta_U.PSNR_U+\theta_V.PSNR_V$.

As already noted, the PSNR is the logarithm of the frame distortion: $PSNR_* = \ln(D_*^2)$ (D_*^2 being the frame distortion for the image of the component *) and it can thus be written at the first order that $\Delta PSNR_* = \frac{\Delta D_*^2}{D_*^2}$.

As the merit m^F of encoding per pixel is the same whatever the block in an image, the relationship between distortion and rate thus remains valid at the image level (by summing over the frame the distortions of the one hand and the rates on the other hand, each corresponding distortion and rate defining a constant ratio m^F) and it can be written that: $\Delta D_*^2 = m^*.\Delta R_*$.

The variation of the video quality Q defined above depending on the attribution of the rate R_* to a given component * can thus be estimated to: $\frac{\partial Q}{\partial R_*} = \frac{\theta_*.m^*}{D_*^2}$.

It is proposed in the process below to encode the residual data such that no component is favoured compared to another one (taking into account the video quality function Q), *i.e.* such that $\frac{\partial Q}{\partial R_Y} = \frac{\partial Q}{\partial R_U} = \frac{\partial Q}{\partial R_V}$. As described below, the encoding process will thus be designed to obtain a value μ^{VIDEO} for this common merit, which value defines the video merit and is selectable by the user (μ^{VIDEO} is directly linked to λ_{video} by a constant factor representing the value of the area unit). In view of the above formulation for $\frac{\partial Q}{\partial R_*}$, the process below is thus designed such that:

$$\mu^{VIDEO} = \frac{\theta_Y.m^Y}{D_Y^2} = \frac{\theta_U.m^U}{D_U^2} = \frac{\theta_V.m^V}{D_V^2},$$

i.e. to obtain, for each of the three components, a frame merit m^* such that the function $e(m^*) = \mu^{VIDEO}.D_*^2(m^*) - \theta_*.m^*$ is null (the distortion at the frame/image level D_*^2 being here noted $D_*^2(m^*)$ in order to explicit the fact that it depends on the frame merit m^*).

The process shown in **Figure 21B** applies to a particular component, denoted * below, of a specific image and is to be applied to each of the two chrominance components U, V (to Y in the alternative implementation) at each iteration of the optimisation process described below with reference to **Figure 22**.

The process of **Figure 21B** applies to an image which is segmented into blocks according to a current segmentation (which can be either an initial segmentation as defined above or a segmentation produced at any step by the optimization process described below with reference to **Figure 22**).

A DCT transform is applied (step S80) to each block thus defined in the concerned image.

Parameters representative of the statistical distribution of coefficients (here α_i, β_i as explained above) are then computed (step S82) for each block type, each time for the various coefficient types. As noted above, this applies to a given component * only. In one embodiment illustrated below with reference to **Figures 24**, some parameters for some enhancement INTRA images are obtained from enhancement INTRA images previous processed and encoded.

Before entering a loop implemented to determine the frame merit m^*, a lower bound m_L^* and an upper bound m_U^* for the frame merit are initialized at step S84 at predetermined values. The lower bound m_L^* and the upper bound m_U^* define an interval, which includes the sought frame merit and which will be reduced in size (divided by two) at each step of the dichotomy process. At initialization step S84, the lower bound m_L^* may be chosen as strictly positive but small, corresponding to a nearly lossless encoding, while the upper bound m_U^* is chosen for instance greater than all initial encoding merits (over all DCT channels and all block types).

A temporary luminance frame merit m^* is computed (step S86) as equal to $\frac{m_L^* + m_U^*}{2}$ (*i.e.* in the middle of the interval).

A block merit is then computed at step S88 for each of the various block types, as explained above with reference to **Figure 21** (see in particular step S34) according to the formula: $m_k = \nu_k . m^*$. Block merits are computed based on the temporary frame merit defined above. The next steps are thus based on this temporary value which is thus a tentative value for the frame merit for the concerned component *.

For each block type k in the frame, the distortions $D^2_{n,k,*}$ after encoding of the various DCT channels n are then determined at step S88 in accordance with what was described with reference to **Figure 13**, in particular step S14, based on the block merit m_k just computed and on optimal rate-distortion curves determined beforehand at step S89, in the same manner as in step S10 of **Figure 13**.

The frame distortion for the luminance frame D^2_* can then be determined at step S92 by summing over the block types thanks to the formula:

$$D^2_* = \sum_k \rho_k . \delta^2_{P,k,*} = \sum_k \rho_k . (\sum_n D^2_{n,k,*})$$

, where ρ_k is the density of a block type in the frame, *i.e.* the ratio between the total area for blocks having the concerned block type k and the total area of the frame.

It is then checked at step S94 whether the interval defined by the lower bound m^*_L and the upper bound m^*_U have reached a predetermined required accuracy α, i.e. whether $m^*_U - m^*_L < \alpha$.

If this is not the case, the dichotomy process will be continued by selecting one of the first half of the interval and the second half of the interval as the new interval to be considered, depending on the sign of $e(m^*)$, *i.e.* here the sign of $\mu^{VIDEO}.D^2_*(m^*) - \theta_*.m^*$,which will thus converge towards zero as required to fulfill the criterion defined above. It may be noted that the selected video merit μ^{VIDEO} (see selection step S81) and, in the case of chrominance frames U, V, the selected balancing parameter θ_* (*i.e.* θ_U or θ_V) are introduced at this stage in the process for determining the frame merit m^*.

The lower bound m^*_L and the upper bound m^*_U are adapted consistently with the selected interval (step S98) and the process loops at step S86.

If the required accuracy is reached, the process continues at step S96 where quantizers are selected in a pool of quantizers predetermined at step S87 and associated with points of the optimal rate-distortion curves already used (see explanations relating to step S8 in **Figure 13**), based on the distortions values $D^2_{n,k,*}$ obtained during the last iteration of the dichotomy process (step S90 described above).

These selected quantizers may be used for encoding coefficients in an encoding process or in the frame of a segmentation optimization method as described below (see step S104 in particular).

The process just described for determining optimal quantizers uses a function $e(m^*)$ resulting in an encoded image having a given video merit (denoted μ^{VIDEO} above), with the possible influence of balancing parameters θ_*.

As a possible variation, it is possible to use a different function $e(m^*)$, which will result in the encoded image fulfilling a different criterion. For instance, if it is sought to obtain a target distortion D_t^2, the function $e(m^*) = D_*^2(m^*) - D_t^2$ could be used instead.

In a similar manner, if it is sought to control the rate of an image (for a given component) to a target rate R_t, the function $e(m^*) = R_*(m^*) - R_t$ could be used. In this case, step S90 would include determining the rate for encoding each of the various channels (also considering each of the various blocks of the current segmentation) using the rate-distortion curves (S89) and step S92 would include summing the determined rates to obtain the rate R_* for the frame.

In addition, although the process of **Figure 21B** has been described in the context of a video sequence with three colour components, it also applies in the context of a video sequence with a single colour component, *e.g.* luminance, in which case no balancing parameter is used ($\theta_* = 1$, which is by the way the case for the luminance component in the example just described where θ_Y was defined as equal to 1).

In this embodiment, the luminance frame merit and the colour frame merits are determined using a balancing parameter between respective distortions at the image level and frame merits.

Figure 22 shows an exemplary embodiment of an encoding process for residual enhancement INTRA image. As briefly mentioned above, the process is an optimization process using the processes described above, in particular with reference to **Figure 21B**.

This process applies here to a video sequence comprising a luminance component Y and two luminance components U,V.

The process starts at step S100 with determining an initial segmentation for the luminance image Y based on the content of the blocks of the image, *e.g.* in accordance with the initial segmentation method described above using a measure of residual activity. As already explained, this segmentation defines a block type for each block obtained by the segmentation, which block type refers not only to the size of the block but also to other possible parameters, such as a label derived for instance from the measure of residual activity.

It is possible in addition to force this initial segmentation to provide at least one block for each possible block type (except possibly for the block types having a skip-label), for instance by forcing some blocks to have the block types not encountered by use of the segmentation method based on residual activity, whatever the content of these blocks. As will be understood from the following description, forcing the presence of each and every possible block type in the segmentation makes it possible to obtain statistics and optimal quantizers for each and every block type and thus to enlarge the field of the optimization process.

The process then enters a loop (optimization loop).

At step S102, DCT coefficients are computed for blocks defined in the current segmentation (which is the initial segmentation the first time step S102 is implemented) and, for each block type, parameters (GGD statistics) representing the probabilistic distributions of the various DCT channels are computed or obtained from previous enhancement INTRA image (see below **Figures 24**). This is done in conformity with steps S4 and S6 of **Figure 13** described above.

The computation of DCT coefficients and GGD statistics is performed for the luminance image Y and for chrominance images U,V (each time using the same current segmentation associating a block type to each block of the segmentation).

Frame merits (m^* above), block merits m_k (for each block type) and optimal quantizers for the various block types and DCT channels can thus be determined at step S104 thanks to the process of **Figure 21B**.

These elements can then be used at step S106 in an encoding cost competition between possible segmentations, each defining a block type for each block of the segmentation. It may be noted that block types with a skip label, i.e. corresponding to non-encoded blocks, may easily be introduced at this stage (when they are not considered at the time of determining the initial segmentation) as their distortion equals the distortion of the block in the base layer and their rate is null.

This approach thus corresponds to performing an initial segmentation of the obtained residual enhancement frame into a set of initial blocks, thus determining, for each initial block, a block type associated with the concerned initial block; determining, for each block type, an associated set of quantizers based on data corresponding to pixels of blocks having said block type; selecting, among a plurality of possible segmentations defining an association between each block of this segmentation and an associated block type, the segmentation which minimizes an encoding cost estimated based on a measure of the rate necessary for encoding each block using the set of

quantizers associated with the block type of the encoded block according to the concerned segmentation.

It is proposed here to use a Lagrangian cost of the type $\frac{D}{\lambda}+R$, or in an equivalent manner $\frac{D^2}{\lambda}+R$, (as an encoding cost in the encoding cost competition) computed from the bit rate needed for encoding by using the quantizers of the concerned (competing) block type and the distortion after quantization and dequantization by using the quantizers of the concerned (competing) block type. As a possible variation, the encoding cost may be estimated differently, such as for instance using only the bit rate just mentioned (*i.e.* not taking into account the distortion parameter).

The Lagrangian cost generated by encoding blocks having a particular block type will be estimated as follows.

The cost of encoding for the luminance is $\frac{\delta^2_{P,k,Y}}{\lambda}+R_{k,Y}$ where $\delta^2_{P,k,Y}$ is the pixel distortion for the block type k introduced above and $R_{k,Y}$ is the associated rate.

It is known that, as rate and distortion values are constrained on a given rate-distortion curve, Lagrange's parameter can be written as follows: $\lambda=-\frac{\partial\delta^2_{P,k,Y}}{\partial R_{k,Y}}$ and thus approximated as follows: $\lambda\approx-\frac{\Delta\delta^2_{P,k,Y}}{\Delta R_{k,Y}}=m_k=v^Y_k.m^Y$ (where v^Y_k is the number of blocks of the given block type per area unit in the luminance image).

It is thus proposed to estimate the luminance cost as follows:

$C_{k,Y}=\frac{\delta^2_{P,k,Y}}{v^Y_k.m^Y}+R_{k,Y}+R_{k,QT}$, where $R_{k,QT}$ is the bit rate associated to the parsing of the generalized quad-tree (representing the segmentation; the "block type quad-tree" as mentioned above) to mark the type of the concerned block in the bit stream. A possible manner to encode the block type quad tree in the bit stream is described below. This bit rate $R_{k,QT}$ is computed at step S105.

When using the embodiment of **Figure 21B** where distinct frame merits m^U, m^V are determined respectively for the U component and for the V component, the estimation of the Lagrangian cost presented above applies in a similar manner in the

case of colour components U,V, except that no rate is dedicated to a chrominance (block type) quad-tree as it is considered here that the segmentation for chrominance images follows the segmentation for the luminance image. The Lagrangian cost for chrominance components can be estimated as follows:

$$C_{k,U} = \frac{\delta^2_{P,k,U}}{\nu^U_k . m^U} + R_{k,U} \quad \text{and} \quad C_{k,V} = \frac{\delta^2_{P,k,V}}{\nu^V_k . m^V} + R_{k,V}.$$

The combined cost, taking into account luminance and chrominance, can thus be estimated by the following formula:

$$C_{k,YUV} = \frac{\delta^2_{P,k,Y}}{\nu^Y_k . m^Y} + \frac{\delta^2_{P,k,U}}{\nu^U_k . m^U} + \frac{\delta^2_{P,k,V}}{\nu^V_k . m^V} + R_{k,Y} + R_{k,U} + R_{k,V} + R_{k,QT}.$$

This formula thus makes it possible to compute the Lagrangian cost in the competition between possible segmentations mentioned above and described in more details below, in the frame of the embodiment of **Figure 21B**.

Here, each considered cost $C_{k,Y}$, $C_{k,UV}$, $C_{k,U}$ or $C_{k,V}$ is computed using a predetermined frame merit (m) and a number (ν) of blocks per area unit for the concerned block type.

Also, the combined encoding cost $C_{k,YUV}$ includes a cost for luminance, taking into account luminance distortion generated by encoding and decoding a luminance block using the set of quantizers associated with the concerned block type, and a cost for chrominance, taking into account chrominance distortion generated by encoding and decoding a chrominance block using the set of quantizers associated with the concerned block type.

Also, the distortions $\delta^2_{P,k,Y}$, $\delta^2_{P,k,U}$ and $\delta^2_{P,k,V}$ are computed in practice by applying the quantizers selected at step S104 for the concerned block type, then by applying the associated dequantization and finally by comparing the result with the original residual. This last step can *e.g.* be done in the DCT transform domain because the IDCT is a L2 isometry and total distortion in the DCT domain is the same as the total pixel distortion, as already explained above.

Bit-rates $R_{k,Y}$, $R_{k,U}$ and $R_{k,V}$ can be evaluated without performing the entropy encoding of the quantized coefficients. This is because one knows the rate cost of each quantum of the quantizers; this rate is simply computed from the probability of falling into this quantum and the probability is provided by the GGD channel modeling associated with the concerned block type. In other words, the

measure of each rate may be computed based on the set of quantizers associated with the concerned block type k and on parameters representative of probabilistic distributions of transformed coefficients of blocks having the concerned block type.

Lastly, the size (more precisely the area) of a block impacts the cost formula through the geometrical parameters ν_k^Y, ν_k^U and ν_k^V.

For instance, in the case of a 16x16-pixel unit area and a 4:2:0 YUV colour format, the number of blocks per area unit for 16x16 blocks is $\nu_k^Y = 1$ for luminance blocks and $\nu_k^U = \nu_k^V = \nu_k^{UV} = 2$ for chrominance blocks. 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 number of blocks per area unit for 8x8 blocks is $\nu_k^Y = 4$ for luminance blocks and $\nu_k^{UV} = 8$ for chrominance blocks.

In the case considered here were possible block sizes are 32x32, 16x16 and 8x8, the competition between possible segmentations performed at step S106 (already mentioned above) seeks to determine for each 32x32 area or LCU both:

- the segmentation of this area into 32x32, 16x16 or 8x8 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 (based on the additivity of costs), one can dramatically decrease the number of configurations to deal with.

As shown in **Figure 23** (left part), a 16x16 block is segmented into four 8x8 blocks. By using 8x8 cost competition (where the cost for each 8x8 block is computed based on the above formula for each possible block types of size 8x8, including for the block type having a skip label, for which the rate is null), the most competitive type (*i.e.* the type with the smallest cost) can be selected for each 8x8 block. Then, the cost $C_{16,best8*8}$ associated with the 8x8 (best) segmentation is just the addition of the four underlying best 8x8 costs.

The bottom-to-top process can be used by comparing this best cost $C_{16,best8*8}$ using 8x8 blocks for the 16x16 block to costs computed for block types of size 16x16.

Figure 23 is based on the assumption (for clarity of presentation) that there are two possible 16x16 block types. Three costs are then to be compared:

- the best 8x8 cost $C_{16,best8*8}$ deduced from cost additivity;
- the 16x16 cost $C_{16,type1}$ using 16x16 block type 1,
- the 16x16 cost $C_{16,type2}$ using 16x16 block type 2.

The smallest cost among these 3 costs decides the segmentation and the types of the 16x16 block.

The bottom-to-top process is continued at a larger scale (in the present case where 32x32 blocks are to be considered); it may be noted that the process could have started at a lower scale (considering first 4x4 blocks). In this respect, the bottom-to-top competition is not limited to two different sizes, not even to square blocks.

By doing so for each 32x32 block of the image, it is thus possible to define a new segmentation, defining a block type for each block of the segmentation (step S108).

Then, if the segmentation does not evolve anymore (*i.e.* if the new segmentation is the same as the previous segmentation) or if a predetermined number of iterations has been reached, the process quits the loop and step S110 (described below) is proceeded with. Else, the process loops to step S102 where DCT coefficients and GGD statistics will be computed based on the new segmentation.

It may be noted in this respect that the loop is needed because, after the first iteration, the statistics are not consistent anymore with the new segmentation (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.

The block type competition helps improving the compression performance of about 10%.

At step S110, DCT coefficients are computed for the blocks defined in the (optimized) segmentation resulting from the optimization process (loop just described), *i.e.* the new segmentation obtained at the last iteration of step S108 and, for each block type defined in this segmentation, parameters (GGD statistics) representing the probabilistic distributions of the various DCT channels are computed. As noted above, this is done in conformity with steps S4 and S6 of **Figure 13** described above.

Frame merits (m^* above), block merits m_k (for each block type) and optimal quantizers for the various block types and DCT channels can thus be

determined at step S112 thanks to the process of **Figure 21B**, using GGD statistics provided at step S110 and based on the optimized segmentation.

The DCT coefficients of the blocks of the images (which coefficients where computed at step S110) are then quantized at step S114 using the selected quantizers.

The quantized coefficients are then entropy encoded at step S116 by any known coding technique like VLC coding or arithmetic coding. Context adaptive coding (CAVLC or CABAC) may also be used.

For example, the quantized coefficients coded by an entropy encoder following the statistical distribution of the corresponding DCT channels. For instance, the entropy coding may be performed by any known coding technique like a context-free arithmetic coding. Indeed, no context is needed simply because the probability of occurrence of each quantum is known *a priori* thanks to the knowledge of the GDD. These probabilities of occurrence may be computed off-line and stored associated with each quantizer.

Context-free coding also allows a straightforward design of the codec with the so-called "random spatial access" feature, desired at the Intra frame of the video sequence.

An enhancement layer bit-stream to be transmitted for the considered residual enhancement image can thus be computed based on encoded coefficients. The bit stream also includes parameters α_i, β_i representative of the statistical distribution of coefficients computed or obtained at step S110, as well as a representation of the segmentation (block type quad tree) determined by the optimization process described above.

In one embodiment, it is proposed to reuse statistics (i.e. parameters α or σ , β) from one enhancement INTRA image to the other.

Figure 24 shows a method for encoding parameters representing the statistical distribution of DCT coefficients (parameters β and σ) in an embodiment where these parameters are not computed for every enhancement INTRA image, but only for some particular images called "*restat*" frames.

In this situation, parameters representative of a probabilistic distribution of coefficients having a given coefficient type in a given block type in a first enhancement INTRA image are reused as parameters representative of a probabilistic distribution of coefficients having the given coefficient type in the given block type in a new enhancement INTRA image to encode. From them, corresponding optimal quantizers

are obtained for quantizing (dequantizing in the decoding) the coefficients having said coefficient type in said block type.

A new enhancement INTRA image f to be encoded is considered in **Figure 24** (step S200).

A proximity criterion between this new image f and the latest "*restat*" frame f_{restat} (*i.e.* the latest image for which parameters representing the statistical distribution of DCT coefficients were computed) is first estimated (step S202).

The proximity criterion is for instance based on a number of images separating the new image and the latest restat frame and/or based on a difference in distortion at the image level between these two images.

In this respect, in order to determine the distortion at the image level for the new image, it is possible to apply, before a decision is made at step 202, the process of **Figure 21B** (see step S92), either including computation of new GGD statistics (step S82) or not. Depending on the decision made based on this distortion at the image level, the new statistics (if they were computed) may be discarded.

If the proximity criterion is fulfilled (for instance if the new image and the latest restat frame are separated by less than a predetermined number of images, or if the distortion at the image level has evolved by less than a predetermined percentage since the last restat frame), it is considered that the statistics computed for the latest restat frame can still be used and a process comparable to the process of **Figure 21B** is thus applied based on the statistics (parameters β and σ) computed for the latest restat frame (step S204), thus without any statistics computation (*i.e.* without performing step S82 in **Figure 21B** or S102 in **Figure 22**). In addition, a flag (see *e.g.* the flag restat_flag in **Figure 24B**) indicating that the image is not a restat frame (*i.e.* that the image is a non restat frame) is set in a header associated with the image.

If the proximity criterion is not fulfilled (for instance if the new image and the latest restat frame are separated by a predetermined number of images – or more, or if the distortion at the image level has evolved by more than a predetermined percentage since the last restat frame), it is considered that the statistics computed for the latest restat frame can no longer be used and a process comparable to the process of **Figure 21B** is thus applied (step S206), including the computation of parameters β, σ (*i.e.* step S82 in **Figure 21B**). In addition, a flag (see *e.g.* the flag restat_flag in **Figure 24B**) indicating that the image is a restat frame is set in the header associated with the image.

It may thus be noted that parameters computed based on a restat frame are kept (*i.e.* stored in memory) so as to be used during the encoding of non restat frames (step S204), and discarded only when new parameters are computed in connection with a further (generally the following) restat frame.

Before explaining the way parameters β, σ are encoded in both cases, we first give the choices made for this parameter encoding in the present example.

To ease the explanation, it is proposed to use the characteristic function $\chi_{n,k,*}$ ($\chi_{n,k,*} \in \{0,1\}$) specifying whether a given DCT channel n (for blocks of block type k and component *) is coded or not (see explanations about the theorem of equal merits and step S14 above). Its value is 1 if the associated DCT channel n is encoded, *i.e.* $D_{n,k,*} < \sigma_{n,k,*}$, and 0 otherwise. As further explained below, if a channel is not encoded, there is no need (it would be a waste of bit-rate) to send the associated statistics (*i.e.* parameters β, σ).

A consequence, the following rule will be followed in the detailed description given below:

- if $\chi_{n,k,*} = 1$, send associated statistics $\sigma_{n,k,*}$ and $\beta_{n,k,*}$

- if $\chi_{n,k,*} = 0$, do not send associated statistics.

In the present embodiment, the parameter $\beta_{n,k,*}$ is tabulated on 8 values as follows : $\beta_{n,k,*} \in \{0.6, 0.8, 1.0, 1.2, 1.4, 1.6, 1.8, 2.0\}$ and is encoded over 3 bits through a look-up table.

The parameter $\sigma_{n,k,*}$ (which is positive) is quantized up to a fixed (predetermined) precision $\sigma_{0,*}$ such that $\sigma_{n,k,*} = \hat{\sigma}_{n,k,*}.\sigma_{0,*}$, where $\hat{\sigma}_{n,k,*}$ is thus a non-null integer to be encoded in the bit stream. It is proposed here to use a precision $\sigma_{0,Y}$ for the luminance component distinct from the precisions $\sigma_{0,U}$, $\sigma_{0,V}$ used for the chrominance components.

The number of bits $N^{\sigma}_{k,*}$ needed to encode the various integers $\hat{\sigma}_{n,k,*}$ depends on the block type k and the component * considered. It is set to be enough to encode the maximum value among these integers for channels to be encoded and $N^{\sigma}_{k,*}$ can thus be written:

$N^{\sigma}_{k,*} = INT(\log_2(\max_{n;\chi_{n,k,*}=1} \hat{\sigma}_{n,k,*})) + 1$, where INT is the integer truncation.

The number of encoded channels for a component and a block type is:

$N_{k,*}^{\chi} = \sum_{n} \chi_{n,k,*}$.

Based on these elements, we now describe the practical encoding of the statistics of a component * of a block type k.

When the current image f is a restat frame and new statistics (*i.e.* new parameters β, σ) have been computed (step S206 described above), it is proposed to encode in the bit stream to be transmitted only parameters relating to channels for which a DCT coefficient is encoded. The practical encoding process is thus as follows:

1. encode the number of encoded channel $N_{k,*}^{\chi}$ (*e.g.* on 10 bits for 32x32 blocks, 8 bits for 16x16 blocks, *etc.*) at step S208;
2. if $N_{k,*}^{\chi} = 0$, encoding is finished (result "*Yes*" at step S210) and a new block type k and/or component * is next processed (step S226);
3. encode $N_{k,*}^{\sigma}$, *e.g.* on 4 bits, at step S212 (when the process continues for the current block type and component, *i.e.* when $N_{k,*}^{\chi}$ is not null);
4. loop on DCT channels, *i.e.* loop on the indicia n (see steps S222 and S224)
   a. encode $\chi_{n,k,*}$ on 1 bit (step S214);
   b. if $\chi_{n,k,*} = 0$, set a register $X_{n,k,*}^{f+1}$ to zero (step S218) to keep track of fact that parameters for the concerned channel are not yet available at the decoder side;
   c. if $\chi_{n,k,*} = 1$, encode $\hat{\sigma}_{n,k,*}$ on $N_{k,*}^{\sigma}$ bits (*i.e.* as a word of $N_{k,*}^{\sigma}$ bits) and $\beta_{n,k,*}$, here on 3 bits (*i.e.* as a 3-bit word),, and set a register $X_{n,k,*}^{f+1}$ to one to keep track of fact that parameters for the concerned channel are available at the decoder side (step S220);
   d. until the number of encoded channels encountered reaches $N_{k,*}^{\chi}$ (which is checked at step S222).

It may be noted that the loop on n may follow a so-called zigzag scan of the DCT coefficients to allow a faster reaching of the bound $N_{k,*}^{\chi}$ and saving the rate of many potentially useless flags $\chi_{n,k,*}$.

When the process has looped through the channels until the number of encoded channels is reached, a new block type k and/or a new component * is considered (S226) and the encoding process loops at step S208.

When all the block types and components have been considered, the needed parameters are encoded and can be transmitted to the decoder.

It may be noted that the register $X_{n,k,*}^{f+1}$ recording whether or not parameters for a given channel have been encoded in the bit stream (and sent to the decoder) will be used when encoding parameters needed with respect to the following image f+1 (if a not a restat frame) to determine whether there is a need to send the concerned parameters (see below), hence the subscript f+1 on $X_{n,k,*}^{f+1}$.

For a restat frame f, we can thus write by convention: $X_{n,k,*}^{f} = 0$.

We now describe the case where no statistics have been computed for the current image f (step S204), *i.e.* the case of "*non-restat*" frames. For such images, use is made of all available statistics coming from the preceding images (i.e. each and every preceding image since the latest restat frame); some additional statistics may have to be encoded as explained below.

Said differently, for non restat frames, the available statistics come from:

- the latest restat frame; and
- the union of additional statistics provided by the images between the latest re-stat frame and the current image.

As already mentioned, the statistics, and thus in particular the additional statistics (to be sent for the current image f), are not computed on the current non restat frame. Actually, the statistics are computed only on restat frames. The additional statistics to be sent for the current image f are just the addition of channels which were not used in the preceding images but are now useful in the current non restat frame f; the statistics of these added channels are however those computed on the latest restat frame.

The use of non restat frame allows not only the saving of bit-rate thanks to a smaller rate for statistics in the bit stream.

The preceding explanations can be written as follows:

$$X_{n,k,*}^{f} = \max_{f_r \le f' < f} \chi_{n,k,*}^{f'}$$

where f_r is the index of the latest restat frame before the current image f and where the superscript f' has been added to the characteristic function $\chi_{n,k,*}$ to

signify that this characteristic function relates to the image f'. This relationship can be rewritten under an inducted form:

$$X^{f}_{n,k,*} = \max\left\{X^{f-1}_{n,k,*}, \chi^{f-1}_{n,k,*}\right\} \qquad \text{(for } f > f_r\text{)}$$

This is important because it means that it is possible to determine $X^{f}_{n,k,*}$ at the decoder side just by the decoding of the statistics (*i.e.* precisely by decoding of $\chi_{n,k,*}$ in the latest restat frame and subsequent images up to the current image) and thus without needing the effective computation of the $\chi_{n,k,*}$ on the decoder side.

The practical encoding of statistics for a non restat frame f is performed as follows:

1. encode $N^{\chi}_{k,*}$ (the number of additional needed statistics) at step S228;

2. if $N^{\chi}_{k,*} = 0$, encoding is finished (based on step S230, going to step S246 to consider a new block type and/or a new component);

3. encode $N^{\sigma}_{k,*}$ (step S232);

4. loop on DCT channels, *i.e.* loop on n (thanks to steps S242 and S244)

   a. if $X^{f}_{n,k,*} = 1$ (test at step S233), set $X^{f+1}_{n,k,*} = 1$ (step S237) and do nothing (as the concerned statistic is already available), *i.e.* loop directly to consider the next channel, if any (via step S242 and step S244);

   b. if $X^{f}_{n,k,*} = 0$,

      i. encode $\chi_{n,k,*}$ on 1 bit (step S234)

      ii. if $\chi_{n,k,*} = 1$ (test at step S235),

         - encode $\hat{\sigma}_{n,k,*}$ on $N^{\sigma}_{k,*}$ bits and $\beta_{n,k,*}$ on 3 bits (additional encoded channel) at step S240;
         - set $X^{f+1}_{n,k,*} = 1$ at step S237;

      iii. if $\chi_{n,k,*} = 0$,

         - set $X^{f+1}_{n,k,*} = 0$ at step S238 (channel still not available at the decoder);

   c. until the number of additional encoded channels reaches $N^{\chi}_{k,*}$ (positive outcome in test S242).

As already noted, the loop on n may follow a so-called zigzag scan of the DCT coefficients to allow a faster reaching of the bound $N_{k,*}^{\chi}$ and saving the rate of many potentially useless flags $\chi_{n,k,*}$.

When each and every channel has been processed, a new block type k and/or a new component * is considered and parameters encoding for this newly considered block type and component starts at step S228.

When all the block types and components have been considered, the needed additional parameters are encoded and can be transmitted to the decoder.

Figure 24A shows a method for decoding parameters representing the statistical distribution of DCT coefficients (parameters β and σ). The method is implemented at the decoder when receiving the parameters encoded and sent in accordance with what has just been described with reference to **Figure 24**.

A new image f to be decoded is considered in **Figure 24** (step S300).

The flag indicating whether or not this new image is a restat frame (*i.e.* whether the previously stored parameters are no longer valid or are still valid) is read in the header associated with the image (step S302).

If the flag indicates that the current image is not a restat frame, the statistics received for the latest restat frame and the subsequent images can still be used. Additional statistics, not yet received but needed for decoding the current image, are thus received and decoded in accordance with the process described below starting at step S328.

If the flag indicates that the current image is a restat frame, previously received statistics (here parameters β and σ) are discarded and new statistics are received and decoded in accordance with the process described below starting at step S308.

Similarly to what happens at the encoder side, parameters received with the restat frame and subsequent images are kept (*i.e.* stored in memory) so as to be used during the decoding of subsequent non restat frames, and discarded only when new parameters are received in connection with a further (generally the following) restat frame.

The decoding process for a restat frame is as follows, starting with a given coefficient type k and a given component *:

1. decode the number of encoded channel $N_{k,*}^{\chi}$ for the current coefficient type k and component * (sep S308);

2. if $N^{\chi}_{k,*}=0$, decoding for the current coefficient type and component is finished (result "*Yes*" at step S310) and a new block type k and/or component * is next processed (step S326);

3. if $N^{\chi}_{k,*}\neq 0$, decode $N^{\sigma}_{k,*}$, here on 4 bits, at step S312 and continue as follows;

4. loop on DCT channels, *i.e.* loop on the indicia n (see steps S322 and S324)

a. decode $\chi_{n,k,*}$ on 1 bit (step S314);

b. if $\chi_{n,k,*}=0$, set a register $X^{f+1}_{n,k,*}$ to zero (step S318) to keep track of fact that parameters for the concerned channel are not yet available ;

c. if $\chi_{n,k,*}=1$, decode $\hat{\sigma}_{n,k,*}$ on $N^{\sigma}_{k,*}$ bits and $\beta_{n,k,*}$, here on 3 bits, and set a register $X^{f+1}_{n,k,*}$ to one to keep track of fact that parameters for the concerned channel are now available (step S320);

d. until the number of encoded channels encountered reaches $N^{\chi}_{k,*}$ (which is checked at step S322).

As noted, the loop on n may follow a so-called zigzag scan of the DCT coefficients (in conformity to what was done at the encoder side).

When the process has looped through the channels until the number of encoded channels is reached, a new block type k and/or a new component * is considered (S326) and the decoding process loops at step S308.

When all the block types and components have been considered, the needed parameters have been decoded and can thus be used to perform the decoding of encoded coefficients (in particular to select the [de]quantizer to be used during the decoding of coefficients).

As already noted, the registers $X^{f+1}_{n,k,*}$ recording (at the decoder side) whether or not parameters for a given channel have been received in the bit stream, decoded and stored (and sent to the decoder) are used when decoding additional parameters received with respect to the following image f+1 (if a not a restat frame) to determine which parameters have already been received and thus which parameters are liable to be received, as further explained below.

The decoding process for a non restat frame is as follows, starting with a given coefficient type k and a given component *:

1. decode $N^{\chi}_{k,*}$ (the number of additional statistics to be received and decoded) at step S328;

2. if $N_{k,*}^{\chi}=0$, decoding is finished (based on step S330, going to step S346 to consider a new block type and/or a new component);

3. else, decode $N_{k,*}^{\sigma}$ (step S332);

4. loop on DCT channels, *i.e.* loop on n (thanks to steps S342 and S344)

a. if $X_{n,k,*}^{f}=1$ (test at step S233), set $X_{n,k,*}^{f+1}=1$ (step S237) and do nothing (as the concerned statistic is already available and is not included therefore in the bit stream relating to the current image), *i.e.* loop directly to consider the next channel, if any (via step S342 and step S344);

b. if $X_{n,k,*}^{f}=0$,

i. decode $\chi_{n,k,*}$ on 1 bit (step S334)

ii. if $\chi_{n,k,*}=1$ (test at step S335),

- decode $\hat{\sigma}_{n,k,*}$ on $N_{k,*}^{\sigma}$ bits and $\beta_{n,k,*}$ on 3 bits (additional encoded channel) at step S240;
- set $X_{n,k,*}^{f+1}=1$ at step S337 (to record the fact that the corresponding parameters have been received and decoded and are thus stored for use by subsequent images);

iii. if $\chi_{n,k,*}=0$,

- set $X_{n,k,*}^{f+1}=0$ at step S238 (statistics for the concerned channel still not available);

c. until the number of additional encoded channels reaches $N_{k,*}^{\chi}$ (positive outcome in test S342).

When each and every channel has been processed, a new block type k and/or a new component * is considered and parameters encoding for this newly considered block type and component starts at step S328.

As clear from what has just been described, the decoding of flags $\chi_{n,k,*}$ allows updating $X_{n,k,*}^{f+1}$ for the following image without starting the decoding of encoded coefficients defining the current image. This means that there is no dependence between images.

Figure 24B shows a possible way to use carriers for transmitting statistics which make it possible to decode a non re-stat frame without decoding the preceding images.

As shown in **Figure 24B**, the data relating to a particular image (*i.e.* the encoded coefficients for that image and parameters sent in connection with these encoded coefficients as per the process of **Figure 24**) are sent using two distinct NAL ("*Network Abstraction Layer*") units, namely a VCL ("*Video Coding Layer*") NAL unit I_i and a non-VCL NAL unit, here an APS ("*Adaptation Parameter Sets*") NAL unit APSi.

The APS NAL unit APSi associated with an image i contains:

- an identifier aps_id which is common to all APS NAL units transporting parameters computed for a given restat frame (*e.g.* in **Figure 24B**: aps_id=0 for images 0,...,7 and aps_id=1 for images 8 and 9);

- a flag restat_flag indicative of whether the associated image i is a restat frame (restat_flag=1 for a restat frame, restat_flag=0 for a non restat frame);

- parameters β and σ for DCT channels for which coefficients are encoded for the corresponding image i, except for parameters which are carried in an APS NAL unit relating to a previous image.

For a restat frame, these parameters are the parameters encoded according to steps S208 to S226 in **Figure 24**; for a non restat frame, these parameters are the additional parameters encoded according to steps S228 to S246 in **Figure 24**.

The VCL NAL unit associated with an image j contains:

- the identifier aps_id designating the APS NAL units transporting parameters computed for a given restat frame, which is thus also the identifier contained in the APS NAL unit APSi associated with image i;

- the video data, *i.e.* the encoded coefficients for encoded DCT channels.

As visible from **Figure 24B**, it is recommended to increment the identifier aps_id when encoding a restat frame so that the identifier aps_id can be used to ease the identification of APS NAL units which define a given set of statistics (*i.e.* parameters computed based on a given restat frame and successively transmitted).

Thus, when randomly accessing a particular image i, the corresponding VCL NAL unit I_i is accessed and the identifier aps_id is read; the decoder then reads and decodes each and every APS NAL unit having this identifier aps_id and corresponding to image i or a prior image. The decoder then has the necessary

parameters for decoding the coefficients contained in VCL NAL unit I_i and can thus proceed to this decoding.

Figure 24B represents one possible example of use of APS NAL units for carrying statistic but other solutions may be employed. For instance, only one APS NAL unit may be used for two (or more) different images when their statistics parameters are identical which avoids transmitting redundant information in APS NAL and finally save bitrate.

Table 1 below proposes a syntax for APS NAL units, modified to include the different statistical parameters regarding the INTRA picture. The main modifications are located in the aps_residual_param part comprising the *restat* flag and the aps_residual_stat part comprising the GGD model parameters for encoded DCT channels in each block type.

aps_rbsp( ) {	Descriptor
aps_id	ue(v)
aps_scaling_list_data_present_flag	u(1)
if(aps_scaling_list_data_present_flag)	
scaling_list_param( )	
aps_deblocking_filter_flag	u(1)
if(aps_deblocking_filter_flag) {	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
aps_sao_interleaving_flag	u(1)
if(!aps_sao_interleaving_flag) {	
aps_sample_adaptive_offset_flag	u(1)
if(aps_sample_adaptive_offset_flag)	
aps_sao_param( )	
}	
aps_adaptive_loop_filter_flag	u(1)
if(aps_adaptive_loop_filter_flag)	
alf_param( )	
aps_extension_flag	u(1)
if(aps_extension_flag)	
aps_residual_param( )	
while(more_rbsp_data( ))	
aps_extension_data_flag	u(1)
rbsp_trailing_bits( )	
}	
aps_residual_param( ) {	Descriptor
for(t = 0; t< number_type_32; t++) {	
block_type_32[t]	u(1)
texture_32 |= block_type_32[t]	
}	
for(t = 0; t< number_type_16; t++) {	
block_type_16[t]	u(1)
texture_16 |= block_type_16[t]	
}	
for(t = 0; t< number_type_8; t++) {	
block_type_8[t]	u(1)
texture_8 |= block_type_8[t]	
}	
re_stat_flag	u(1)

if (re_stat_flag == true) {	
for(b = 0; i<number_proba; b++) {	
for(t = 1; t< number_type_32; t++) {	
if (CU_type_32[t] == 1) {	
block_proba_32[b*(number_type_32+1)+t]	u(8)
}	
if (texture_32 ==1) {	
block_proba_32[b*(number_type_32+1)]	u(8)
}	
}	
}	
for(b = 0; i<number_proba; b++) {	
for(t = 1; t< number_type_16; t++) {	
if (block_type_16[t] == 1) {	
block_proba_16[b*(number_type_16+1)+t]	u(8)
}	
if (texture_16 ==1) {	
block_proba_16[b*(number_type_16+1)]	u(8)
}	
}	
	u(1)
for(b = 0; i<number_proba; b++) {	
for(t = 1; t< number_type_32; t++) {	
if (block_type_8[t] == 1) {	
block_proba_8[b*(number_type_8+1)+t]	u(8)
}	
if (texture_8 ==1) {	
block_proba_8[b*(number_type_8+1)]	u(8)
}	
}	
}	
}	
for(t = 1; t< number_type_32; t++) { /* For Block type 32x32 */	
aps_residual_stats(10) /* For Y component */	
aps_residual_stats(8) /* For U component */	
aps_residual_stats(8) /* For V component */	
}	
for(t = 1; t< number_type_16; t++) { /* For Block type 16x16 */	
aps_residual_stats(8) /* For Y component */	
aps_residual_stats(6) /* For U component */	
aps_residual_stats(6) /* For V component */	

}	
for(t = 1; t< number_type_8; t++) { /* For Block type 8x8*/	
aps_residual_stats(6) /* For Y component */	
aps_residual_stats(4) /* For U component */	
aps_residual_stats(4) /* For V component */	
}	
}	
aps_residual_stats (n) { /*Statistics for each block type */	Descriptor
stat_presence_flag	u(1)
if (stat_presence_flag == 1) {	
num_coeff_minus_1	u(n)
num_bit_minus_1	u(4)
m = num_bit_minus_1+1;	
for (t = 0; t< 2^n; t++) {	
while (num_coeff_minus_1!= 0)	
data_flag	u(1)
if (data_flag == 1) {	
alpha	u(3)
beta	u(m)
}	
}	
}	
}	
}	

Table 1 (Where number_type_32 =10, and the number_type_16 =10,and number_type_8 =19)

According to another possible variation, a given set of statistics could be used for non-adjacent images. For instance, images I_8 and I_9 refer to a new set of statistics but the following image I_{10} may refer to the previous set of statistics (i.e. the set used by images I_0 to I_7) and thus uses an aps_id equal to 0.

The "*proximity criterion*" in S202 could be replaced by other suitable tests. For example, a scene change could be detected and, when it is detected, new statistics could be calculated and a new restat frame sent. Also, detecting a difference in distortion between images is just one way of detecting a decrease in quality of images and other ways of achieving the same result can be used in embodiments.

It will also be appreciated that the restat_flag is merely one example of information supplied by the encoder indicating when the parameters of the parametric probabilistic model are reusable or are no longer reusable. Other ways are possible. For example, when the identifier aps_id changes each time a restat frame is sent, the restat_flag can be omitted and the identifier aps_id itself indicates when the parameters are no longer reusable (or when new parameters are being supplied).

It is now proposed to encode the segmentation of the frame into block types using the syntax of a quad-tree. To achieve that, the selected segmentation is represented as a quad tree having a plurality of levels, each associated with a block size, and leaves associated with blocks and having a value indicating either a label for the concerned block or a subdivision of the concerned block. As described below, the encoding comprises a step of compressing the quad tree using an arithmetic entropy coder that uses, when coding the segmentation relating to a given block, conditional probabilities for the various possible leaf values depending on a state of a block in the base layer co-located with said given block.

Conventional quad-tree coding may also be used in a variant.

Back to the coding using conditional probabilities, for example, a generalized quad-tree with a plurality of (more than two) values per level may be used as follows:

- at the level of 32x32 blocks, the following values can be taken: 0 for a skip label, 1 for label 1, *etc.* , N_{32} for label N_{32} and N_{32}+1 for a block split into smaller (here 16x16) blocks;

- at the level of 16x16 blocks, the following values can be taken: 0 for a skip label, 1 for label 1, *etc.* , N_{16} for label N_{16} and N_{16}+1 for a block split into smaller (here 8x8) blocks

- at the level of 8x8 blocks, the following values can be taken: 0 for a skip label, 1 for label 1, *etc.* , N_8 for label N_8.

The generalized quad-tree may then be compressed using an arithmetic entropy coder associating the conditional probability $p(L \mid s_B)$ to each label L, where s_B is the state of the co-located block in the base layer, for instance computed based on the pixel morphological energy of the co-located base layer block. The various possible conditional probabilities are for instance determined during the encoding cost competition process described above.

A representation of the probabilities $p(L \mid s_B)$ is sent to the video decoder 30 (in the bit stream) to ensure decodability of the quad-tree by a context-free arithmetic decoder. This representation is for instance a table giving the probability $p(L \mid s_B)$ for the various labels L and the various states s_B considered. Indeed, as the video decoder 30 decodes the base layer, it can compute the state of the co-located block in the base layer and thus determine, using the received table, the probabilities respectively associated to the various labels L for the computed state; the arithmetic decoder then works using these determined probabilities to decode the received quad-tree.

The bit stream may also include frame merits m^Y, m^U, m^V determined at step S112.

Transmitting the frame merits makes it possible to select the quantizers for dequantization at the decoder according to a process similar to **Figure 21** (with respect to the selection of quantizers), without the need to perform the dichotomy process.

According to a first possible embodiment (as just mentioned), the transmitted parameters may include the parameters defining the distribution for each DCT channel, *i.e.* the parameter α (or equivalently the standard deviation σ) and the parameter β computed at the encoder side for each DCT channel, as shown in step S22.

Based on these parameters received in the data stream, the decoder may deduce the quantizers to be used (a quantizer for each DCT channel) thanks to the selection process explained above at the encoder side (the only difference being that the parameters β are for instance computed from the original data at the encoder side whereas they are received at the decoder side).

Dequantization (step 120A of **Figure 12**) can thus be performed with the selected quantizers (which are the same as those used at encoding because they are selected the same way).

According to a second possible embodiment, the transmitted parameters may include a flag per DCT channel indicating whether the coefficients of the concerned DCT channel are encoded or not, and, for encoded channels, the parameters β and the standard deviation σ (or equivalently the parameter α). This helps minimizing the amount of information to be sent because channel parameters are sent only for encoded channels. According to a possible variation, in addition to flags indicating whether the coefficients of a given DCT channel are encoded or not, information can be transmitted that designates, for each encoded DCT channel, the

quantizer used at encoding. In this case, there is thus no need to perform a quantizer selection process at the decoder side.

Dequantization (step 120A of **Figure 12**) can thus be performed at the decoder by use of the identified quantizers for DCT channels having a received flag indicating the DCT channel was encoded.

Figure 25 shows the adaptive post-filtering applied at the encoder as mentioned above (see also **Figure 9**) in order to determine the parameters of post-filters to be used at the decoder.

As explained in the context of **Figure 9**, the enhanced image (*i.e.* the sum of the image obtained by decoding the base layer and by upsampling, and of the image obtained by decoding the enhancement layer) is reconstructed at the encoder (according to a process similar to the decoding process implemented at the decoder) in order to produce a rough decoded image.

A deblocking filter DBF, a sample adaptive offset filter SAO and an adaptive loop filter ALF are successively applied to obtain the (post-filtered) decoded version of the image. In a variant, part of these filters may be deactivated. As already noted, parameters of these filters (in particular for the sample offset filter and then for the adaptive loop filter) are selected at this stage such that the post-filtered version is as close as possible to the original image (raw video), according to a proximity criterion, which is for instance in practice a Lagrangian cost (rate-distortion cost).

The deblocking filter DBF is a conventional HEVC deblocking filter as described for instance in JCTVC H1003. Such a deblocking filter receives as an input a quantization parameter QP. The quantization parameter is for instance used to adjust tap filters in the deblocking filter.

It is proposed here that the quantization parameter QP input to the deblocking filter DBF is determined by a first converter CONV1 based on the luminance frame merit m^Y determined during the encoding process (see above step S112).

The quantization parameter QP input to the deblocking filter is for instance deduced from the luminance frame merit using a high rate asymptotic approximation on uniform quantifiers: $QP = INT(3.\log_2(m^Y)+9)$, where INT is the integer truncation.

It may be noted that the same quantization parameter QP is used for the application of the deblocking filter to the luminance and chrominance components.

It is also possible to process in a distinct manner IDR ("*Instantaneous Decoder Refresh*") images, i.e. images "I": the blocks of such an image are processed

considering the blocks as intra with their coded block flags set to 0. The quantization parameter QP input to the deblocking filter for IDR images is computed by the first converter CONV1 in accordance with the following formula: $QP = INT(3.\log_2(m^Y) + 9)$ where INT is the integer truncation.

The sample adaptive offset filter SAO is a picture-based SAO filter having for instance the features proposed in JCTVC-G490 and JCTVC-G246.

It is proposed here that the rate-distortion slope (also called "*lambda parameter*") λ_* input to the SAO filter for a colour component * is determined by a second converter CONV2 based on the frame merit m^* determined during the encoding process (see above step S112) for the concerned colour component *.

The rate-distortion slope is for instance used in the SAO filter when computing the rate-distortion cost C (C=distortion+λ_*.rate, where distortion is the distortion after SAO filtering and rate is the rate of coding the SAO filter parameters) of each of a plurality of configurations of the filter, in order to select the configuration (pair geometry-intensity) having the lower cost.

The second converter CONV2 computes the rate-distortion slope λ_* by applying a conversion ratio to the corresponding frame merit m^*. This ratio depends on the area unit used when computing the block merits (and hence when taking into account the merit at pixel level) as described in connection with Figure 12.

In the present example, as mentioned above, the area unit is the area of a 16x16 block ($\nu_k = 1$ for block types of size 16x16). For each component *, the rate-distortion slope λ_* input to the SAO filter is thus here taken as equal to the frame merit m^*.

The adaptive loop filter ALF is a conventional picture-based HEVC adaptive loop filter, for instance as described in JCTVC H0274 and JCTVC H0068.

It is proposed here that the rate-distortion slope λ_Y input to the adaptive loop filter ALF for the luminance component is the rate-distortion slope determined by the second converter CONV2 as described above (*i.e.* based on the luminance frame merit m^Y determined during the encoding process).

As for the SAO filter, the rate-distortion slope is for instance used in the ALF filter when computing the rate-distortion cost C of each of a plurality of configurations of the filter, in order to select the configuration having the lower cost.

The common rate-distortion slope λ_{UV} input to the adaptive loop filter ALF for the chrominance components is determined by the second converter CONV2 based on the frame merit(s) for chrominance components.

In the context of the embodiment mentioned above (**Figure 21B**), the chrominance rate-distortion slope λ_{UV} is for instance determined by the second converter CONV2 as the harmonic mean of the two chrominance merits m^U, m^V resulting from the encoding process (step S112): $\lambda_{UV} = \frac{2}{\frac{1}{m^U} + \frac{1}{m^V}}$.

Figure 25A shows the post-filtering applied at the decoder as mentioned above with reference to **Figure 10**.

As shown in **Figure 10** and explained above, post-filtering is applied to the version of the image resulting from the sum of the decoded and up-sampled base layer image and of the enhancement layer (residual) image.

As shown in **Figure 25A**, the deblocking filter DBF, the sample adaptive offset filter SAO and the adaptive loop filter ALF, each adjusted based on the parameters received in the bit stream 99'' from the encoder, are successively applied to obtain the (post-filtered) decoded version of the image.

The sample adaptive offset filter SAO and the adaptive loop filter ALF do not need other input than the received parameters to proceed. The deblocking filter DBF on the other hand must be provided the quantification parameter QP.

In conformity with what was done at the encoder, the quantification parameter QP input to the deblocking filter DBF is provided by a first converter CONV1 based on the luminance frame merit m^Y. As already explained above, the luminance frame merit m^Y is available at the decoder either because it was received in the channel model parameter bit stream 99' or because it is computed at the decoder based on the received statistic parameters α (or σ) and β thanks to a method identical to the method performed at the encoder.

The encoding and decoding features of the scalable video coder of **Figure 7** and of the corresponding scalable video decoder of **Figure 8** are now described with reference to **Figures 26** to **47** when these coder and decoder handle enhancement INTER images (i.e. "B" images on **Figures 3** and **4**). Below it is referred to an enhancement INTER images regardless of its component concerned (luminance

component or chrominance components) since the same process is applied to the various components.

As shown, the coder and decoder implement conventional INTRA prediction and INTER prediction ("INTRA" and "INTER" on the Figures) as in H.264/AVC or HEVC. INTER prediction is known to include so-called MERGE SKIP and MERGE modes. INTER prediction mode consists in the motion compensated temporal prediction of the prediction unit. This uses two lists of past and future reference images depending on the temporal coding structure used (see **Figures 3** and **4**). This temporal prediction process as specified by HEVC is re-used here.

Note that, in some embodiments, during encoding, a constraint INTRA prediction is applied to intra blocks preventing the usage of neighbouring inter predicted blocks for performing the INTRA prediction.

Embodiments of the invention may implement all or part of the new coding modes based on inter-layer prediction 76 as mentioned above.

In the example of **Figure 7**, five inter-layer prediction tools are implemented to offer two new coding modes, namely the "base mode prediction mode" ("Base Mode" on the Figure) and the "Intra Base Layer (BL) mode" ("IntraBL" on the Figure), and to offer various sub-modes of the conventional INTER prediction mode 55, namely a "generalized residual inter-layer prediction" ("GRILP" on the Figure) sub-mode, an "inter-layer motion vector prediction" ("ILMVP" on the Figure) sub-mode and an "inter difference mode" ("InterDiff" on the figure) sub-mode.

Briefly, Intra BL mode consists in predicting a coding unit or block thanks to its co-located area in the up-sampled decoded base image that temporally coincides. Intra BL mode is already implemented in SVC.

The Base Mode prediction mode consists in predicting a block from its co-located area in a so-called Base Mode prediction image, constructed both on the encoder and decoder sides using data and prediction data from the base layer. Motion compensation 55' and post-processing 49' are implemented to generate this Base Mode prediction image. As described below with reference to **Figures 30** to **36**, this new mode generates a prediction image that reveals often to offer good coding performance.

The Generalized Residual Inter-Layer Prediction (GRILP) sub-mode mainly supplements the conventional INTER prediction for a given coding unit or block. This conventional INTER prediction outputs a prediction block together with a motion vector, from which a temporal residual is obtained.

GRILP sub-mode then consists in predicting such temporal residual of the INTER block from a block residual computed from the reconstructed base layer. In practice, the temporal residual conventionally obtained is further predicted with the block residual corresponding to the same (co-located) prediction in the base layer. As explained below with reference to **Figures 37** to **41**, this approach tends to reduce the content in the resulting residual, thus reducing the coding cost.

As explained below, GRILP may also be applied to the conventional INTRA prediction (**Figure 41** below).

The Inter-layer motion vector prediction (ILMVP) sub-mode also supplements the conventional INTER prediction which may implement an augmented motion vector prediction (AMVP) where the motion vector obtained from the INTER prediction is predicted from one vector predictor of a predefined set. ILMVP sub-mode attempts to exploit the correlation between the motion vectors coded in the base image and the motion contained in the enhancement layer, by adding the motion vector associated with the co-located block in the base layer to the set of vector predictors. Due to this correlation, better prediction of the motion information can be obtained, thus reducing the coding cost. This is described below with reference to **Figures 44** to **47**.

The inter difference sub-mode can be viewed as another kind of inter layer residual prediction method that can be used in competition or in replacement of the GRILP mode. This method consists first in performing a motion estimation on a current block of a current Enhancement Layer (EL) image in order to obtain a motion vector designating a reference block in a EL reference image. A difference image is then computed between the EL reference image and an upsampled version of the image of the base layer temporally co-located with the EL reference image. A motion compensation is then performed on the block of the obtained difference image pointed by the motion vector determined during the motion estimation step to obtain a residual block. This residual block is then added to the reference block to obtain a final block predictor used to predict the current block.

These prediction modes or sub-modes may be activated by the user. Flags may thus be added into a slice header of the bit-stream by the encoder, that indicate for the slice (1) whether or not the GRILP prediction mode is active, (2) the HEVC INTRA prediction is used, (3) the inter-layer prediction modes IntraBL and Base Mode are active, (4) the inter difference prediction mode is active. In addition as discussed below with reference to **Figure 27 and Figure 48A to 48F**, indication of the coding mode within the bit-stream is made for each coding unit or block that is encoded.

Therefore, several candidates for prediction of a coding unit or block can be obtained: predictor from IntraBL, predictor from the Base Mode prediction image, conventional INTRA predictor, conventional INTER predictor, GRILP predictor over the conventional INTER predictor and/or inter difference predictor over the conventional inter predictor.

Note that the ILMVP sub-mode impacts the cost of coding the motion vectors.

A competition between all the available coding modes is thus performed by the coding mode selection module 75 to select, for each successive coding unit or block, a corresponding coding mode that minimizes a rate distortion cost (or coding cost). As for HEVC, the competition is performed for each LCU (Largest Coding Unit) to determine the best coding representation in the considered enhancement INTER image.

An exemplary rate distortion optimal mode decision involves a recursive search of the best quad-tree representation of each Largest Coding Unit. At each possible quad-tree decomposition depth (coding units from 32x32 to 8x8), a set of coding units is considered. For each of these coding units, a set P of available coding modes is being considered. The coding mode p_{opt} that minimizes the lagrangian cost function if selected for the considered CU:

$$p_{opt} = \text{Argmin}_{p \in P}\{D(p) + \lambda_{final} \cdot R(p)\}$$

where the λ_{final} parameter is obtained for Luma and Chroma components from a user-specified Quality Parameter as shown in step S62 of the above **Figure 21A**.

The same steps as described above for INTRA images are performed for INTER images.

However, at step S60, an offset of quality parameter QP_{offset} is defined for the image. As shown in the Figure, the quality parameter offset QP_{offset} is obtained from a table QP_{config} storing a QP offset scheme that is function of the layer *layerId* and the image index *PicIdx* within the GOP as described below.

In the process of obtaining the λ_{final} parameter, a final quality parameter QP_{final} is also generated that will be is used as a quantization parameter during quantization 58/58' in the considered layer.

Note that due to a coding mode competition performed successively for each block, the conventional INTRA prediction mode is available for a given block

since the blocks in its causal neighbouring are available in their encoded and reconstructed states.

In the prior art, the quantization parameter is typically chosen as follows. A global quantization parameter is determined for the whole video. Afterwards, for then encoding of a given INTER image (base or enhancement), a quantization parameter offset is added to this global quantization parameter based on the temporal layer the image belongs to. This quantization parameter offset is typically a positive integer between 0 and 4. By increasing the quantization parameter for images of the higher temporal depths based on this quantization parameter offset, the bitrate used by these images is reduced while the images of the lower temporal depths get more bitrate. It has been observed that such approach leads to increased rate distortion performances in the coding of the enhancement layer

Temporal depth of an image is illustrated in **Figure 4**. The temporal hierarchy is defined for the GOP and will be the same for each GOP until a new INTRA image "I". The last image of each GOP corresponds to the depth 0. It means that, in the most degraded mode of temporal scalability, only this image remains: the video sequence is then reduced to the succession of images having a GOP index of 7. The GOP index is the index of the image within its GOP and corresponds to the POC modulo the GOP size, the Modulo(x) operator consisting in taking the reminder of a integer division by x. Of course, as no decoding is possible without decoding this image, the first image of the sequence, the INTRA image "I" is also at the temporal depth 0.

The next depth, the depth 1 consists in the image of GOP index 4, which is the last image of the first half of the GOP. The depth 2 is constituted by images of GOP index 2 and 6. The depth 3 is constituted by images of GOP index 1, 3, 5 and 7.

The temporal depth of an image thus reflects the number of defined successive prediction dependencies between the image and a self-coded image.

Taking into account this temporal hierarchy, **Figure 26A** illustrates the quantization offsets typically used for a GOP of size 8 in the prior art. It can be contemplated that, with the exception of the INTRA image "I" which is encoded using an offset of 0 to insure a maximum quality to this image, all the B images are encoded with an offset corresponding to their temporal depth. All images at the temporal depth 0 are provided with an offset of 1, the images at temporal depth 1 are provided with an offset of 2. The images at temporal depth 2 are provided with an offset of 3. The

images at temporal depth 3 are provided with an offset of 4. This is in line with the idea that images must be encoded with an offset depending on the number of images they are reference for.

In the context of scalability, all the layers (base and enhancement) inherit the temporal and encoding hierarchies as described in the foregoing. Typically, the same quantization offsets are used for the base layer and for the enhancement layers. For example, the quantization offset scheme shown in **Figure 26A** is used for all the spatial layers in some prior art.

Since the inventors have noticed that for predictive encoding the quality of the images does not depend exclusively on the allocated bitrate but also on the quality of the predictors and therefore, the quality of the reference images used for encoding, a new scheme for quality parameter offset, as shown in **Figure 26B**, is proposed for the enhancement layer, in the random access coding structure case.

The quantization or quality parameter QP offset is no longer determined only based on the level of the temporal layer the image belongs to but is now also based on the quality of the reference image used for the predictive encoding.

Due to scalability (see **Figure 4**) the encoding of a given enhancement image involves predictive encoding based on the corresponding base image in the base layer, and sometimes even from other reference base images within the base layer.

For example, when considering image of POC index 8, the encoding of the enhancement image 8 is not only based on the enhancement image 0 in the enhancement layer, it is also based on corresponding base image 8 in the base layer. It is likely that this corresponding base image is a better predictor of the enhancement image 8 than enhancement image 0 in the enhancement layer. This is due to the temporal distance of 8 images between image 0 and image 8. Due to this availability of a better predictor, it may be contemplated to use a different QP offset for the image 8 in the enhancement layer from the one used in the base layer.

The quantization offset schemes of **Figures 26A** for the base layer and **26B** for the enhancement layer thus provides each layer to be encoded with its own quantization offset scheme. Accordingly more flexibility is given to tailor the encoding process.

As an example, the case of image with POC index 1 is examined. This image belongs to the temporal depth 3, and as is, it is encoded with an offset 4 in the prior art. Indeed, image 1 is not used as a reference for any further image encoding.

Actually image 1 is encoded based on reference images 0 and 2. Image 0 is encoded with a maximum quality and due to this quality and to its temporal proximity with image 1, it is likely to be an excellent predictor image for image 1. It means that the residue obtained when encoding image 1 using image 0 as a reference are small, their encoding will result in a low bitrate image whatever the chosen quantization offset. Due to the excellent quality of image 0 as a predictor, the residue concerns mainly the high frequency details, the low frequencies being well predicted. The use of a large quantization offset results in discarding the main part of these high frequencies. It comes that using a lower level of quantization preserves these high frequencies which are of interest. And, icing on the cake, using a lower level of quantization does not lead to increase significantly the bitrate due to the quality of image 0 as a predictor and the resulting small size of the residue. By contrast, taking image 3, while in the same temporal depth as image 1, namely the temporal depth 3, its encoding is based on images 2 and 4, respectively from temporal depth 2 and 1. The quality of these latter is likely to be worse than the quality of image 0 or other images from temporal depth 0. The same reasoning as was made for image 1 does not apply. A higher quantization offset may be contemplated for image 3.

As explained on the example of image 1 and 3, determining the quantization offset based only to the temporal depth the image belongs to does not lead necessarily to the most relevant choice. Taking into account the number of images N_{im_ref} using the current image as a reference image for temporal prediction and the temporal distance D_t of the current image to its reference image having the lowest quantization offset and the value of the quantization parameter $\widetilde{QP}$ applied to this reference image is advantageous and leads to more relevant quantization offsets.

$$QP_{offset} = f(N_{im_ref}, D_t, \widetilde{QP})$$

By more relevant, one means a set of chosen quantization offset that leads to a better rate distortion performance in the overall scalable video coding scheme.

In this quantization approach, the quantization offset of a current image is equal or above the quantization offset of its reference image having the lowest quantization parameter. Typically, some of the images belonging to the same temporal depth will not share the same quantization offset.

In the example of **Figure 26B**, the quantization offset determined for image 1 is 2 instead of 4 in the prior art (**Figure 26A**). This choice results in a best restitution of high frequency details in image 1. By contrast, for image 3, a quantization offset of 3 is determined which is higher than the level of 2 determined for image 1 while being

lower than the level of 4 chosen by the prior art. It may also be contemplated that the quantization offsets are more evenly distributed as it is in the prior art leading to a smoother result in quality over time.

For example, non-INTRA images of temporal depth 0 (namely images with POC 8 and 16 on **Figure 26B**) are given a quantization offset that is higher than in the prior art (**Figure 26A**). This can be explained by the following. In the non-scalable coding case, it is commonly admitted that emphasizing the quality of such "anchor" images of each GOP is important, in order to ensure good temporal prediction, hence good coding efficiency, of other images contained in the considered GOPs. However, in the scalable context, this assumption can be moderated because base images are available to predict enhancement images, in addition to the above "anchor images". Therefore, reducing the relative bitrate allocated to the anchor images, as compared to the non-scalable context, tends to lower the bitrate of those images, without impacting the rate distortion trade-off for other images of the same GOP in a significant way. For this reason, such approach is of interest in the scalable context, in order to improve the rate distortion performance obtained for the whole GOP coding in the enhancement layer.

As explained in the foregoing, it is advantageous to use different quantization offset schemes for the different spatial or SNR layers in a scalable encoder. In an embodiment, for a GOP size of 8, a quantization offset scheme depending only on the temporal depth the image belongs to is adopted for the base layer, for example the one illustrated by **Figure 26A**, and the quantization offset scheme taking into account the quality of the reference image used for the predictive encoding is adopted for all the enhancement layers, for example the one illustrated by **Figure 26B**. Alternatively, a quantization offset scheme taking into account the quality of the reference image used for the predictive encoding is adopted for all the spatial or SNR layers, for example the one illustrated by **Figure 26B**.

Figures 26C and **26D** provides examples of QP configurations assigned according to other embodiments, for a video sequence with GOP size equal to 16.

Figures 26E and **26F** provides examples of QP configurations assigned according to different embodiments, for a video sequence with GOP size equal to 4.

Back to **Figure 21A**, the table QP_{config}, function of the layer *layerId* and the image index *PicIdx* within the GOP thus corresponds to either the scheme of **Figure 26A** (if *layerId=0*) or the scheme of **Figure 26B** (if *layerId>0*).

Once the QP_{offset} for the considered INTER image *PicIdx* is obtained, then a QP value for that image is computed as follows: $QP = QP_{video} + QP_{offset}$, where the QP_{video} parameter is as obtained from step S54.

The Lagrange parameter λ_{final} is then computed at step S62 which is follows by step S68 (in case of INTER image) where the final quality parameter QP_{final} is obtained from $QP_{final} = 4.2005 \times \ln(\lambda_{final}) + 13.7122$. This final quality parameter QP_{final} is used as a quantization parameter for the HEVC uniform scalar quantization 58/58' in the considered layer.

According to an embodiment, step 68 can be omitted in which case the quantization parameter used during quantization 58 is equal to $QP_{video} + QP_{offset}$.

Back to the minimization of the lagrangian cost function during competition and segmentation of the enhancement INTER image, the consideration of each coding unit at each depth level may lead to a very big number of possible segmentations to evaluate. Fortunately, by using the classical so-called bottom-to-top competition technique (as shown above with reference to **Figure 23**), one can dramatically decrease the number of segmentations to deal with.

Once the best CU coding configuration is obtained at each LCU quad-tree depth level, then the best quad-tree representation is determining for that LCU. This is done by performing a bottom-to-top rate distortion cost minimization (see **Figure 23** above) during the recursive LCU processing. The bottom-to-top approach thus gives an optimal segmentation of the image (or of each LCU) into coding units and associated coding modes. A conventional representation of the image is therefore a quad-tree representing the segmentation in coding units and the coding mode of each coding unit (which is quite equivalent to the block type quad-tree mentioned above).

The obtained quad-tree representing the coding unit segmentation and associated coding mode is coded using an arithmetic coding. The latter can be based on a specific syntax or arrangement of the available coding modes through a tree as shown in **Figure 27** or **Figure 48A** to **48F** which are known by both encoder and decoder.

The tree depicted in figure 27 shows an organization of the coding modes within the arithmetic coding that is to be used when the Inter Difference mode is not considered by the encoder nor the decoder. The arithmetic coding thus provides flags to encode each coding mode. Flags are made of syntax elements. Such tree means that a first syntax element from an arithmetic code (note that efficient arithmetic codes

make it possible that a coding mode is coded upon less than 1 bit) indicates whether the coding mode associated with a given enhancement block is based on temporal/Inter prediction ("Inter CU") or not ("Non Inter CU", i.e. on spatial/Intra prediction). Then a second syntax element indicates whether the sub-mode GRILP is activated or not in case of Inter CU or indicates whether the coding mode is a conventional Intra prediction ("Intra CU") or based on Inter-layer prediction ("Inter-layer CU") in case of Non Inter CU. Lastly, in case of Inter-layer CU, a third syntax element from the arithmetic code indicates whether the coding mode is the IntraBL mode or the Base Mode prediction mode.

At the decoder, the latter reads the coding mode flags in the quad tree representing the coding unit segmentation and decodes them according the coding mode tree to know the coding mode of each coding unit.

The specific order of the flags in the coding mode tree of figure 27 is designed to minimize the mean number of bits necessary to encode the coding unit mode thus reducing the bit-stream size.

Some other flags may be inserted by the encoder in the slice header to indicate the list of possible coding modes used in the coding units of the slice. The encoder can then simplify the syntax of the following coding mode tree used in the slice to encode only the modes indicated in the slice header for each coding unit or block in the slice. For example if no intraCU is used in a slice, the coding mode tree is simplified by removing the choice between IntraCU and Inter-Layer CU in the coding mode tree.

The decoder can then read the flags in the slice header and know the syntax to use to decode the flags into coding mode for each coding unit in the slice.

Figures 27A to **27C** illustrate simplified trees organizing the coding modes within the arithmetic coding. In particular, each simplified tree may be used when at least one specific coding mode is deactivated, thus resulting in the corresponding tree branch to be useless. Using simplified tree makes it possible to further reduce the cost for coding each coding modes of the quad-tree to encode.

Since the four trees of **Figures 27** to **27C** can be used, the encoder specified in a syntax element of the slice header which coding modes are activated and deactivated. Corresponding tree can thus be used by both the encoder and the decoder to respectively encode and decode.

The tree of **Figure 27** is the complete tree corresponding to all the coding modes activated when the Inter Difference mode is not used. The syntax element in the slice header is made of three sub-elements that all have their flag set to 1:

enable_intra_pred_flag=1 to indicate that the Intra BL prediction mode is active; *enable_inter_layer_flag*=1 to indicate that the inter-layer prediction modes (including IntraBL and Base Mode) are active; and *enable_grilp_flag*=1 to indicate that the GRILP is active. Table 2 represents a modified version of the HEVC slice syntax introducing the above three flag: *enable_intra_pred_flag, enable_inter_layer_flag, enable_grilp_flag.*

Slice_header( ) {	Descriptor
first_slice_in_pic_flag	u(1)
if(first_slice_in_pic_flag = = 0)	
slice_address	u(v)
slice_type	ue(v)
entropy_slice_flag	u(1)
if(!entropy_slice_flag && nal_unit_type != 20) {	
pic_parameter_set_id	ue(v)
if(output_flag_present_flag)	
pic_output_flag	u(1)
if(**separate_colour_plane_flag** = = 1)	
colour_plane_id	u(2)
if(IdrPicFlag) {	
idr_pic_id	ue(v)
no_output_of_prior_pics_flag	u(1)
} else {	
pic_order_cnt_lsb	u(v)
short_term_ref_pic_set_sps_flag	u(1)
if(!short_term_ref_pic_set_sps_flag)	
short_term_ref_pic_set(num_short_term_ref_pic_sets)	
else	
short_term_ref_pic_set_idx	u(v)
if(long_term_ref_pics_present_flag) {	
num_long_term_pics	ue(v)
for(i = 0; i < num_long_term_pics; i++) {	
delta_poc_lsb_lt[i]	ue(v)
delta_poc_msb_present_flag[i]	u(1)
if(delta_poc_msb_present_flag[i])	
delta_poc_msb_cycle_lt_minus1[i]	ue(v)
used_by_curr_pic_lt_flag[i]	u(1)
}	
}	
}	
if(sample_adaptive_offset_enabled_flag) {	
slice_sao_interleaving_flag	u(1)
slice_sample_adaptive_offset_flag	u(1)
if(slice_sao_interleaving_flag && slice_sample_adaptive_offset_flag) {	
sao_cb_enable_flag	u(1)
sao_cr_enable_flag	u(1)
}	
}	
if(scaling_list_enable_flag | | deblocking_filter_in_aps_enabled_flag | | (sample_adaptive_offset_enabled_flag && !slice_sao_interleaving_flag) | | adaptive_loop_filter_enabled_flag)	

aps_id	ue(v)
if(slice_type == P || slice_type == B) {	
num_ref_idx_active_override_flag	u(1)
if(num_ref_idx_active_override_flag) {	
num_ref_idx_l0_active_minus1	ue(v)
if(slice_type == B)	
num_ref_idx_l1_active_minus1	ue(v)
}	
}	
if(lists_modification_present_flag) {	
ref_pic_list_modification( )	
ref_pic_list_combination( )	
}	
if(slice_type = = B)	
mvd_l1_zero_flag	u(1)
}	
if(cabac_init_present_flag && slice_type != I)	
cabac_init_flag	u(1)
if(!entropy_slice_flag) {	
slice_qp_delta	se(v)
if(deblocking_filter_control_present_flag) {	
if(deblocking_filter_in_aps_enabled_flag)	
inherit_dbl_params_from_aps_flag	u(1)
if(!inherit_dbl_params_from_aps_flag) {	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
}	
if(slice_type = = B)	
collocated_from_l0_flag	u(1)
if(slice_type != I && ((collocated_from_l0_flag && num_ref_idx_l0_active_minus1 > 0) || (!collocated_from_l0_flag && num_ref_idx_l1_active_minus1 > 0))	
collocated_ref_idx	ue(v)
if((weighted_pred_flag && slice_type = = P) || (weighted_bipred_idc == 1 && slice_type == B))	
pred_weight_table( )	
}	
if(slice_type = = P || slice_type = = B)	
if (layer_id > 0) {	
six_minus_max_num_merge_cand	ue(v)

} else {	
five_minus_max_num_merge_cand	ue(v)
}	
enable_grilp_flag	u(1)
enable_intra_pred_flag	u(1)
enable_inter_layer_flag	u(1)
if(adaptive_loop_filter_enabled_flag) {	
slice_adaptive_loop_filter_flag	u(1)
if(slice_adaptive_loop_filter_flag && alf_coef_in_slice_flag)	
alf_param( )	
if(slice_adaptive_loop_filter_flag && !alf_coef_in_slice_flag)	
alf_cu_control_param( )	
}	
if(seq_loop_filter_across_slices_enabled_flag && (slice_adaptive_loop_filter_flag || slice_sample_adaptive_offset_flag || !disable_deblocking_filter_flag))	
slice_loop_filter_across_slices_enabled_flag	u(1)
if(tiles_or_entropy_coding_sync_idc > 0) {	
num_entry_point_offsets	ue(v)
if(num_entry_point_offsets > 0) {	
offset_len_minus1	ue(v)
for(i = 0; i < num_entry_point_offsets; i++)	
entry_point_offset[i]	u(v)
}	
}	
}	

Table 2

The tree of **Figure 27A** corresponds to the case where the GRILP mode is deactivated (*enable_grilp_flag*=0). Less codes are thus needed to encode the remaining coding modes.

The tree of **Figure 27B** corresponds to the case where the IntraBL mode is deactivated (*enable_intra_pred_flag* =0).

The tree of **Figure 27C** corresponds to the case where the inter layer prediction modes are deactivated (*enable_inter_layer_flag* =0).

Table 3 and 4 respectively describe the coding unit and prediction unit syntax corresponding to the four encoding trees of **Figures 27** to **27C.**

coding_unit(x0, y0, log2CbSize) {	Descriptor
CurrCbAddrTS = MinCbAddrZS[x0 >> Log2MinCbSize][y0 >> Log2MinCbSize]	
if(slice_type != I)	
skip_flag[x0 **][** y0 **]**	ae(v)
if(skip_flag[x0][y0])	
prediction_unit(x0, y0 , log2CbSize)	
else if(slice_type != I || log2CbSize == Log2MinCbSize) {	
if(slice_type != I)	
pred_mode_flag	ae(v)
if(PredMode == MODE_INTRA && enable_intra_pred_flag = = 1 && enable_inter_layer_flag ==1)	
inter_layer_flag	ae(1)
if((PredMode != MODE_INTRA || log2CbSize = = Log2MinCbSize) && inter_layer_flag == 0)	
part_mode	ae(v)
if(PredMode != MODE_INTRA && **enable_grilp_flag = = 1)**	
grilp_flag	ae(v)
x1 = x0 + ((1 << log2CbSize) >> 1)	
y1 = y0 + ((1 << log2CbSize) >> 1)	
x2 = x1 – ((1 << log2CbSize) >> 2)	
y2 = y1 – ((1 << log2CbSize) >> 2)	
x3 = x1 + ((1 << log2CbSize) >> 2)	
y3 = y1 + ((1 << log2CbSize) >> 2)	
if(PartMode = = PART_2Nx2N)	
prediction_unit(x0, y0 , log2CbSize)	
else if(PartMode = = PART_2NxN) {	
prediction_unit(x0, y0 , log2CbSize)	
prediction_unit(x0, y1 , log2CbSize)	
} else if(PartMode = = PART_Nx2N) {	
prediction_unit(x0, y0 , log2CbSize)	
prediction_unit(x1, y0 , log2CbSize)	
} else if(PartMode = = PART_2NxnU) {	
prediction_unit(x0, y0 , log2CbSize)	
prediction_unit(x0, y2 , log2CbSize)	
} else if(PartMode = = PART_2NxnD) {	
prediction_unit(x0, y0 , log2CbSize)	
prediction_unit(x0, y3 , log2CbSize)	
} else if(PartMode = = PART_nLx2N) {	
prediction_unit(x0, y0 , log2CbSize)	
prediction_unit(x2, y0 , log2CbSize)	
} else if(PartMode = = PART_nRx2N) {	
prediction_unit(x0, y0 , log2CbSize)	
prediction_unit(x3, y0 , log2CbSize)	
} else { /* PART_NxN */	
prediction_unit(x0, y0 , log2CbSize)	
prediction_unit(x1, y0 , log2CbSize)	
prediction_unit(x0, y1 , log2CbSize)	
prediction_unit(x1, y1 , log2CbSize)	
}	

if(!pcm_flag)	
transform_tree(x0, y0, x0, y0, log2CbSize, log2CbSize, log2CbSize, 0, 0)	
}	
}	

Table 3

prediction_unit(x0, y0, log2CbSize) {	Descriptor
if(skip_flag[x0][y0]) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0 **][** y0 **]**	ae(v)
} else if(PredMode == MODE_INTRA) {	
if(PartMode == PART_2Nx2N && pcm_enabled_flag && log2CbSize >= Log2MinIPCMCUSize && log2CbSize <= Log2MaxIPCMCUSize)	
pcm_flag	ae(v)
if(pcm_flag) {	
num_subsequent_pcm	tu(3)
NumPCMBlock = num_subsequent_pcm + 1	
while(!byte_aligned( ))	
pcm_alignment_zero_bit	u(v)
pcm_sample(x0, y0, log2CbSize)	
} else {	
if (enable_intra_pred_flag = = 0 || inter_layer_flag = = 1) {	
inter_layer_mode	ae(1)
} else {	
prev_intra_luma_pred_flag[x0 **][** y0 **]**	ae(v)
if(prev_intra_luma_pred_flag[x0][y0])	
mpm_idx[x0 **][** y0 **]**	ae(v)
else	
rem_intra_luma_pred_mode[x0 **][** y0 **]**	ae(v)
intra_chroma_pred_mode[x0][y0]	ae(v)
SignalledAsChromaDC = (chroma_pred_from_luma_enabled_flag ? intra_chroma_pred_mode[x0][y0] = = 3 : intra_chroma_pred_mode[x0][y0] = = 2)	
}	
}	
} else { /* MODE_INTER */	
merge_flag[x0 **][** y0 **]**	ae(v)
if(merge_flag[x0][y0]) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0 **][** y0 **]**	ae(v)
} else {	
if(slice_type = = B)	
inter_pred_flag[x0 **][** y0 **]**	ae(v)
if(inter_pred_flag[x0][y0] = = Pred_LC) {	
if(num_ref_idx_lc_active_minus1 > 0)	
ref_idx_lc[x0 **][** y0 **]**	ae(v)
mvd_coding(mvd_lc[x0][y0][0], mvd_lc[x0][y0][1])	

mvp_lc_flag[x0][y0]	ae(v)
} else { /* Pred_L0 or Pred_BI */	
if(num_ref_idx_l0_active_minus1 > 0)	
ref_idx_l0[x0][y0]	ae(v)
mvd_coding(mvd_l0[x0][y0][0], mvd_l0[x0][y0][1])	
mvp_l0_flag[x0][y0]	ae(v)
}	
if(inter_pred_flag[x0][y0] == Pred_BI) {	
if(num_ref_idx_l1_active_minus1 > 0)	
ref_idx_l1[x0][y0]	ae(v)
if(mvd_l1_zero_flag) {	
mvd_l1[x0][y0][0] = 0	
mvd_l1[x0][y0][1] = 0	
} else	
mvd_coding(mvd_l1[x0][y0][0], mvd_l1[x0][y0][1])	
mvp_l1_flag[x0][y0]	ae(v)
}	
}	
}	
}	

Table 4

Of course, further simplified tree may be used when two of these flags are set to 0.

One skilled in the art will adapt the above explanation about the first to third syntax elements to each of these simplified trees.

The tree depicted in **figure 48A** depicts an alternative organization of the coding modes within the arithmetic coding that is to be used when the Inter Difference sub-mode is considered by the encoder in competition with other modes and by the decoder. Again, the arithmetic coding provides flags to encode each coding mode. As can be seen in this figure, in case the GRILP sub-mode is not activated for an inter CU, an additional syntax element from the arithmetic code indicates whether the Inter Difference sub-mode is active or not for the considered CU.

At the decoder, the latter reads the coding mode flags in the quad tree representing the coding unit segmentation and decodes them according to the coding mode tree to know the coding mode of each coding unit.

The specific order of the flags in the coding mode tree of **figure 48A** is designed to minimize the mean number of bits necessary to encode the coding unit mode thus reducing the bit-stream size.

Figure 48B describes an alternative coding mode tree to the coding mode tree of **figure 48A**. In that alternative, the position of the GRILP sub-mode and the inter difference sub-mode are inverted. This alternative could be useful when it appears that

the inter difference sub-mode has a greater probability of occurrence than the GRILP mode.

In addition to the three flags mentioned with regard to the embodiment of the coding mode tree of **figure 27**, a fourth flag related to the Inter Difference sub-mode is proposed to be inserted in the slice header. This flag called Enable_InterDiff_Flag allows enabling or disabling the usage of the inter difference mode. Similarly to oher flags, it allows simplification of the syntax at the encoder level. The decoder can then read this flag in the slice header and know if the Inter Difference mode is enabled for blocks belonging to this slice.

Figures 48C to **48F** illustrate simplified trees organizing the coding modes within the arithmetic coding.

As can be seen in **figure 48C**, when the GRILP mode is disabled (enable_grilp_flag=0), the flag related to the inter difference mode takes the position of the flag related to the GRILP mode.

We can remark that the obtained coding mode tree is the same than the one that would have been obtained if the Inter Difference mode had replaced the GRILP mode instead of being used in competition with GRILP mode and other modes. In that case there is no need of the Enable_grilp_flag, this flag being replaced by the inter difference flag.

Similarly to **figure 27A** to **27C**, further examples of simplifications of the coding mode tree are represented in **figure 48D** to **48F**. Here we considered that the statistics of the occurences of modes are kept similar when a mode is disabled. As a consequence, it can be seen that the order of modes in the coding mode tree is not modified. If a mode is disabled, the mode (or the modes) following the disabled mode replaces (or replace) the disabled mode. But these statistics may be slightly modified and the disabling of a coding mode may induce an important re-ordering of modes in the coding mode tree. In **figure 48G** we provide an example of a coding mode tree where the modification of statistics due to the disabling of the GRILP mode and the inter difference mode induces an important re-ordering of the coding mode tree.

As seen above, each coding mode tree 48A to 48G is associated to a combination of slice's flags allowing a decoder, when receiving these flags, to identify which coding mode tree have to be applied.

The output enhancement bit-stream for an enhancement INTER image thus comprises: the above quad-tree (which enable the decoder to know the segmentation

of the image and the associated coding modes); the encoded texture residual data (i.e. the coded quantized transformed coefficients after quantization 58).

Note that the residual blocks obtained for each of the available coding modes (original coding unit less the predictor used) are encoded using conventional residual coding as in HEVC, where the quantization uses the quality parameter QP_{final} as quantization parameter as explained above with reference to **Figure 21A** (through step S68). In a variant, the low complexity coding (LCC) mechanism as described above with reference to **Figures 9** to **25** may be used given the user-specified quality parameter QP and corresponding merit obtained through the process of **Figure 21A** (through step S66).

Figure 28 illustrates an encoder architecture 2800 according to an embodiment of the present invention for handling enhancement INTER image coding. This scalable Inter codec design exploits inter-layer redundancy in an efficient way through inter-layer prediction, while enabling the use of the low-complexity texture encoder of **Figure 11**.

The diagram of **Figure 28** illustrates the base layer coding, and the enhancement layer coding process for a given enhancement INTER image.

The first stage of the process corresponds to the processing of the base layer, and is illustrated on the bottom part of the figure 2800A.

First, the input image to code 2810 is down-sampled 28A to the spatial resolution of the base layer, to obtain a raw base layer 2820. Then it is encoded 28B in an HEVC compliant way, which leads to the "encoded base layer" 2830 and associated base layer bit-stream 2840.

In the next step, some information is extracted from the coded base layer that will be useful afterwards in the inter-layer prediction of the enhancement INTER image. The extracted information comprises at least:

- reconstructed (decoded) base images 2850 which are later used for inter-layer texture prediction (used by Intra BL mode) and correspond to reference images that temporally coincide with reference images used in the enhancement layer (used GRILP mode);
- prediction information 2870 of the base layer which is used in several inter-layer prediction tools in the enhancement INTER image (used by Base Mode and ILMVP). It comprises, among others, coding unit information, prediction unit partitioning information, prediction modes, motion vectors, reference image indices, etc.

- temporal residual data 2860, used for temporal prediction in the base layer, is also extracted from the base layer, and is used next in the prediction of the enhancement INTER image (used by Base Mode).

Once all these information have been extracted from the coded base image, it undergoes an up-sampling process (see 78 in **Figure 7**), which aims at adapting this information to the spatial resolution of the enhancement layer. The up-sampling of the extracted base information is affected as described below, for the three types of data listed above.

- with respect to the reconstructed base image 2850, it is being up-sampled to the spatial resolution of the enhancement layer 2880A. In the same way as for the INTRA LCC coder of **Figure 9**, an interpolation filter corresponding to the DCTIF 8-tap filter used for motion compensation in HEVC is employed.
- the base prediction information 2870 is being transformed, so as to obtain a coding unit representation that is adapted to the spatial resolution of the enhancement layer. The prediction information up-sampling mechanism is introduced below.
- the temporal residual information 2860 associated with INTER predicted blocks in the base layer is collected into an image buffer, and is up-sampled to 2880C by means of a 2-tap bi-linear interpolation filter. This bi-linear interpolation of residual data is identical to that used in the former H.264/SVC scalable video coding standard.

Once all the information extracted from the base layer is available in its up-sampled form, then the encoder is ready to predict 28C the enhancement image. The prediction process used in the enhancement layer is executed in a strictly identical way on the encoder side and on the decoder side.

As mentioned above, the prediction process involves selecting the enhancement INTER image segmentation in a rate distortion optimal way in terms of coding unit (CU) representation, prediction unit (PU) partitioning and prediction mode selection.

The prediction process 28C attempts to construct each prediction block or unit 2891 for each prediction unit of the LCUs in current enhancement image to code. To do so, it determines the best rate distortion trade-off between the quality of decoded blocks according to the considered prediction blocks, and the summed rate cost of the corresponding prediction and residual information.

INTER prediction mode is referred to "HEVC temporal predictor" 2890 on **Figure 28**. It is enhanced using the above-mentioned GRILP and Inter layer motion

vector prediction sub-modes. Note that in the temporal predictor search, the prediction process searches the best one or two (respectively for uni- and bi-directional prediction) reference areas to predict a current prediction unit of current image. GRILP and Inter layer motion information prediction sub-modes are described below with reference to **Figures 37** to **41** and **44** to **47**.

Briefly, GRILP sub-mode consists in determining a block predictor candidate of a block or prediction unit within the enhancement layer and an associated first order residual; determining a block predictor in the base layer co-located with the determined block predictor candidate within the enhancement layer ; determining a block residual in the base layer co-located with at least a part of the first order residual; and predicting at least a part of the first order residual using the determined block residual in the base layer as a predictor. This is equivalent to adding the block residual from the base layer to the block predictor candidate from the enhancement layer, to obtain a prediction block.

ILMVP sub-mode consists in supplementing the set of vector predictors used to encode the motion vector of the INTER prediction, by adding the motion vector associated with the base block that is co-located with the concerned prediction unit in the corresponding image of the base layer, if that motion vector exists.

GRILP and ILMVP sub-modes are combined with the conventional INTER prediction mode (including the well-known MERGE and MERGE SKIP modes), thus providing up to 10 modes (because GRILP has no meaning with the MERGE SKIP mode since no residual is coded).

Conventional INTRA prediction from HEVC ("INTRA" on the Figure) is also provided, in particular when the encoding of the enhancement INTER image residual is performed according to conventional HEVC residual encoding.

In addition to the INTRA prediction mode, the "Intra BL" mode and the "Base Mode" prediction mode are implemented.

The Intra BL prediction type comprises predicting a prediction unit or block of the enhancement INTER image with the spatially corresponding area in the up-sampled corresponding decoded base image.

The "Base Mode" prediction mode comprises predicting an enhancement prediction unit or block from the spatially corresponding area in a so-called "Base Mode prediction image". This Base Mode prediction image is constructed with the help of inter-layer prediction tools. The construction of this base mode prediction image is explained in detail below, with reference to **Figures 35** and **36**. Briefly, it is constructed

by predicting current enhancement image by means of the up-sampled prediction information and temporal residual data that has previously been extracted from the base layer and re-sampled to the enhancement spatial resolution.

In the case of SNR scalability, the derived prediction information corresponds to the Coding Unit structure of the base image, taken as is, before the motion information compression step performed in the base layer.

In the case of spatial scalability, the prediction information of the base layer firstly undergoes a so-called prediction information up-sampling process.

Once the derived prediction information is obtained, a Base Mode prediction image is computed, by means of temporal prediction of derived INTER CUs and Intra BL prediction of derived INTRA CUs

It may be noted that the "Intra BL" and "Base Mode" prediction modes try to exploit the redundancy that exists between the underlying base image and current enhancement image.

It may be further noted that an additional mode corresponding to a difference INTRA coding mode could be added to the list of INTRA modes comprising the conventional INTRA, INTRA BL and Base mode. Briefly, this mode consists first in predicting a block in the enhancement layer with the block co-located to this block in the base layer. The difference (residual) between these two blocks is then predicted using an INTRA prediction. But since this differential block contains pixels in the differential domain, it must be predicted from pixels in its neighborhood in the same domain. As a consequence, a same difference between blocks in the enhancement layer containing pixels used for intra prediction and corresponding co-located blocks in the base layer is computed. Difference INTRA coded blocks are then coded in the form of a mode identifier, an intra-prediction direction and an INTRA prediction residual corresponding to the difference between the predictor obtained by the INTRA prediction in the differential domain and the obtained differential block.

The "rate distortion optimal mode decision" of **Figure 28** results in the following elements:

- a set of coding unit representations with associated prediction information for current image. This is called prediction information 2892 on **Figure 28**. All this information then undergoes a prediction information coding step 2897, which constitutes a part of the coded video bit-stream. An implementation of the set is a quad-tree as mentioned above representing the coding modes used for each coding unit within the LCUs. The quad-tree makes it possible to signal, in the bit-

stream, the prediction mode used for each coding unit .

It may be noted that according to an embodiment, the "Intra BL" and/or "Base Mode" prediction images of **Figure 28** can be inserted into the list of reference images used in the temporal prediction of current enhancement image.

- a set 2891 of predictor blocks for each prediction unit segmenting the current enhancement INTER image to encode. This set is involved in the encoding of the texture data part of current enhancement INTER image.

The next encoding step illustrated in **Figure 28** comprises computing the difference 2893 between each original block and the obtained corresponding prediction block. This difference comprises the residual data of current enhancement INTER image 2894, which is then processed by a conventional HEVC texture coding process 28D. In a variant, the LCC texture coding process as described above (i.e. 90F) may be used.

The process provides encoded DCT X values 2895 which comprise enhancement coded texture for output and possibly (not shown) decoder information such as parameters of the channel model for output when the LCC process is implemented.

A further available output is the enhancement coded prediction information 2898 derived from the prediction information 2892. This comprises the encoded quad-tree as described above with reference to **Figure 27**. This also comprises the prediction information (e.g. motion vector) obtained from INTER prediction mode and/or INTRA prediction mode. The motion vectors and motion information are encoded using motion information prediction as illustrated below with reference to **Figures 44** to **47**.

Figure 29 schematically illustrates a decoder architecture 2900 according to an embodiment of the invention for handling enhancement Inter frame decoding. This decoder architecture performs the reciprocal process of the encoding process of **Figure 28**.

Inputs to the decoder illustrated in **Figure 29** include:

- coded base layer bit-stream 2901
- coded enhancement layer bit-stream 2902, including the data 2896, 2897 and 2898 defined above.

The first stage of the decoding process 29A corresponds to the base layer, starting with the decoding 29A' of the base layer encoded base image 2910. This decoding is then followed by the preparation of all data useful for the inter-layer

prediction of the enhancement layer if any. The data extracted from the base layer decoding step is of three types:

- the decoded base image 2911 undergoes a spatial up-sampling step 29C, in order to form the "Intra BL" prediction image 2912 and provides reference images for GRILP. The up-sampling process 29C used here is identical to that of the encoder (**Figure 29**).
- the prediction information contained in the base layer (base motion information 2913) is extracted and re-sampled 29D towards the spatial resolution of the enhancement layer. The prediction info up-sampling process is the same as that used on the encoder side.
- the temporal residual texture data contained in the base layer (base residual 2915) is extracted and up-sampled 29E, in the same way as on the encoder side, to give up-sampled residual information .

Once all the base layer texture and prediction information has been up-sampled, then it is used to construct the "Base Mode" prediction image 2916, exactly in the same way as on the encoder side.

Motion vectors from the base layer are now available for the ILMVP decoding process, i.e. available to be added to the set of vector predictors from which the coded prediction information may be correctly decoded.

Next, the processing of the enhancement layer 29B is effected as illustrated in the upper part of **Figure 29**. This begins with the entropy decoding 29F of the prediction information contained in the enhancement layer bit-stream to provide decoded prediction information 2930. This, in particular, provides the segmentation of the enhancement INTER image and the prediction mode (coding mode 2931) associated to each prediction unit.

Once the prediction mode of each prediction unit of the enhancement INTER image is obtained, the decoder 2900 is able to construct each prediction block 2950 that was used in the encoding of current enhancement INTER image.

The next decoder steps then comprises decoding 29G the texture data (encoded DCT X 2932) associated to current enhancement image. Conventional HEVC texture decoding process is used (LCC decoding as shown in **Figure 12** is used if LCC is applied at the encoder) and produces decoded residual data X_{deQ} 2933.

Once the entire residual image 2933 is obtained from the texture decoding process, it is added 29H to each prediction block 2950 previously constructed. This leads to the decoded current enhancement INTER image 2935 which, optionally,

undergoes some in-loop post-filtering process 29I. Such processing may comprise the HEVC deblocking filter, Sample Adaptive Offset (specified by HEVC) and Adaptive Loop Filtering (also specified by the HEVC standard).

The decoded image 2960 is ready for display and the individual images can each be stored as a decoded reference image 2961, which may be useful for motion compensation 29J in association with the HEVC temporal predictor 2970, as applied for subsequent images.

The base mode prediction mode is now described with more details with reference to **Figures 30** to **36**.

As mentioned above, this mode requires a base mode prediction image to be generated for a given enhancement INTER image to encode. In particular, it is generated using prediction information from the base image that temporally coincides.

To be consistent, the prediction information from the base layer has to be up-sampled. This is described below with reference to **Figures 30** to **34**.

Then the base mode prediction image is generated, possibly using a specific post-filtering, as illustrated through **Figures 35** and **36**.

A method of deriving prediction information, in a base-mode prediction mode, for encoding or decoding at least part of an image of an enhancement layer of video data, in accordance with an embodiment of the invention will now be described. Embodiments described below addresses, in particular, HEVC prediction information up-sampling in the case of spatial scalability with scaling ratio 1.5 between two successive scalability layers.

Figures 30, 31A and **31B** schematically illustrate a prediction information up-sampling process, executed both by the encoder and the decoder in at least one embodiment of the invention for constructing a "Base Mode" prediction image. The organization of the coded base image, in terms of LCU, coding units (CUs) and prediction units (PUs) is schematically illustrated in **Figure 30(a)**. **Figure 30(b)** schematically illustrates the enhancement image organization in terms of LCUs, CUs and PUs, resulting from a prediction information up-sampling process applied to the base image prediction information. By prediction information, in this example is meant a coded image structure in terms of LCUs, CUs and PUs.

Figure 30(a) illustrates a part 3010 of a base layer image of the base layer. In particular, the Coding Unit representation that has been used to encode the base image is illustrated, for the two first LCUs (Largest Coding Unit) 3011 and 3012 of

the base image. The LCUs have a height and width, as illustrated, and an identification number, here shown running from zero to two. The Coding Unit quad-tree representation of the second LCU 3012 is illustrated, as well as prediction unit (PU) partitions e.g. partition 3016. Moreover, the motion vector associated with each prediction unit, e.g. vector 3017 associated with prediction unit 3016, is shown.

In **Figure 30(b)**, the result 3050 of the prediction information up-sampling process applied on base layer 3010 is illustrated. **Figure 30** illustrates a case where the LCU size in the enhancement layer is identical to the LCU size in the base layer. As can be seen with reference to **Figure 30(b)**, the prediction information that corresponds to one LCU in the base image spatially overlaps several LCUs in the enhancement image. For example, the up-sampled version of base LCU 3012 results in the enhancement LCUs 1, 2, 5 and 6. The individual prediction units exist in a scaling relationship known as a quad-tree. It may be noted that the coding unit quad-tree structure of coding unit 3012 has been re-sampled in 3050 as a function of the scaling ratio, for example a non-integer ratio as 1.5, that exists between the enhancement image and the base image. The prediction unit partitioning is of the same type (i.e. the corresponding prediction units have the same shape) in the enhancement layer and in the base layer. Finally, motion vector coordinates e.g. 3057 have been re-scaled as a function of the spatial ratio between the two layers.

As a result of the prediction information up-sampling process, prediction information is available on the encoder and on the decoder side, and can be used in various inter-layer prediction mechanisms in the enhancement layer.

In the scalable encoder and decoder architectures according to embodiments of the invention, this up-scaled prediction information is used in two ways.

– in the construction of a "Base Mode" prediction image of a considered enhancement image,

– for the inter-layer prediction of motion vectors (ILMVP) in the coding of the enhancement image.

Figure 31A schematically illustrates prediction modes that can be used in the proposed scalable codec architecture, according to an embodiment, for prediction of a current enhancement image. Schematic 3110 corresponds to the current enhancement image to be predicted. The base image 3120 corresponds to the base layer decoded image that temporally coincides with current enhancement image. Schematic 3130 corresponds to an example reference image in the enhancement layer

used for the temporal prediction of the current image 3110. Schematic 3140 corresponds to the Base Mode prediction image as described with reference to **Figure 35**.

As illustrated by **Figure 31A**, the prediction of current enhancement image 3110 comprises determining, for each block 3150 in current enhancement image 3110, the best available prediction mode for that block 3150, considering prediction modes including temporal prediction, Intra BL prediction and Base Mode prediction.

Figure 31B also illustrates how the prediction information contained in the base layer is extracted, and then used in two different ways.

First, the prediction information of the base layer is used to construct 3160 the "Base Mode" prediction image 3140. This construction is discussed below with reference to **Figure 35**.

Second, the base layer prediction information is used in the predictive coding 3170 of motion vectors in the enhancement layer. Therefore, the INTER prediction mode illustrated on **Figure 31B** makes use of the prediction information contained in the base image 3120. This allows inter-layer prediction of the motion vectors of the enhancement layer, hence increases the coding efficiency of the scalable video coding system.

The overall prediction up-sampling process of **Figure 30** involves up-sampling first the coding unit structure, and then up-sampling the prediction unit partitions. The goal of inter-layer prediction information derivation is to keep as much accuracy as possible in the up-scaled prediction unit and motion information, in order to generate as accurate a Base Mode prediction image as possible.

In the case of spatial scalability having a scaling ratio of 1.5, the block-to-block correspondence between the base image and the enhancement image is more complex that would be in a dyadic case as is schematically illustrated in **Figure 31B**.

A method in accordance with an embodiment of the invention for deriving prediction information in the case of a scaling ratio of 1.5 is as follows:

Each Largest Coding Unit (LCU) in the enhancement image to be encoded or decoded is split into coding units (CU)s having a minimum size (e.g. 4x4). Each CU obtained in this way is then considered as a prediction unit having a prediction unit type 2Nx2N.

The prediction information of each obtained 4x4 prediction unit is computed as a function of prediction information associated with the co-located area in

the base layer as will be described in more detail. The prediction information derived from the base layer includes the following:

- Prediction mode,
- Merge information,
- Intra prediction direction (if relevant),
- Inter direction,
- Cbf (coded block flag) values,
- Partitioning information,
- CU size,
- Motion vector prediction information,
- Motion vector values (It may be noted that the motion field is inherited prior to the motion compression that takes place in the base layer).

Derived motion vector coordinates are computed as follows:

$$mv_x = mvbase_x \times \frac{PicWidthEnh}{PicWidthBase} \quad (1)$$

$$mv_y = mvbase_y \times \frac{PicHeightEnh}{PicHeightBase} \quad (2)$$

where:

(mv_x, mv_y) represents the derived motion vector, $(mvbase_x, mvbase_y)$ represents the base motion vector, and $(PicWidthEnh \times PicHeightEnh)$ and $(PicWidthBase \times PicHeightBase)$ are the sizes of the enhancement and base images, respectively.

- reference image indices
- QP value (used afterwards when applying the DBF onto the Base Mode prediction image) – see **Figure 22**

Each LCU of the enhancement image is thus organized regardless of the way the corresponding LCU in the base image has been encoded.

The prediction information derivation for a scaling ratio 1.5 aims at generating up-scaled prediction information that may be used later during the predictive coding of motion information. As explained the prediction information can be used in the construction of the Base Mode prediction image. The Base Mode prediction image quality highly depends on the accuracy of the prediction information used for its prediction.

Figure 31B schematically illustrates the correspondence between each 4x4 enhancement coding unit (processing block) being considered, and the respective

corresponding co-located spatial area in the base image in the case of a 1.5 scaling ratio. As can be seen, the corresponding co-located area in the base image may be fully contained within a coding unit (prediction unit) of the base layer, or may overlap two or more coding units of the base layer. This happens for enhancement CUs having coordinates (XCU, YCU) such that:

$$(XCU \bmod 3=1) \text{ or } (YCU \bmod 3=1) \quad (3)$$

In the first case in which the corresponding co-located area in the base image is fully contained within a coding unit of the base layer, the prediction information derivation for the considered 4x4 enhancement CU is simplified. It comprises obtaining the prediction information values of the corresponding base prediction unit within which the enhancement CU is fully contained, transforming the obtained prediction information values towards the resolution of the enhancement layer, and providing the considered 4x4 enhancement CU with the so-transformed prediction information.

In the second case where the corresponding co-located area in the base image overlaps, at least partially, each of a plurality of coding units of the base layer a different approach is adopted.

For these particular coding units, each 4x4 enhancement coding unit is split into 2x2 Coding Units. Each 2x2 enhancement CU contained in a 4x4 enhancement CU then has a unique co-sited CU in the base image and inherits the prediction information coming from that co-located base image CU. For example, with reference to **Figure 32**, the enhancement 4x4 CU with coordinates (1,1) inherits prediction data from 4 different elementary 4x4 CUs {(0,0); (0,1); (1,0); (1,1)} in the base image.

As a result of the prediction information up-sampling process for scaling ratios of 1.5 the Base Mode image construction process is able to apply motion compensated temporal prediction on 2x2 coding units and hence benefits from all the prediction information issued from the base layer.

The method of determining where the prediction information is derived from, according to a particular embodiment of the invention is illustrated in the flow chart of **Figure 33**.

The algorithm of **Figure 33** is repeatedly applied to each Largest Coding Unit LCU of the considered enhancement image. The first part of the algorithm is to determine, for a considered enhancement LCU, the one or more LCU's of the base image that are concerned by current enhancement LCU.

In step S3301, it is determined whether or not the current LCU in the enhancement image is fully covered by the spatial area that corresponds to an up-sampled Largest Coding Unit of the base layer. For example, LCU's 0 and 2 of **Figure 30(b)** are fully covered by their respective co-located LCU in its up-scaled form, while LCU 1 is not fully covered by the spatial area corresponding to an up-sampled LCUs of the base layer, and is covered by spatial areas corresponding to parts of two up-sampled LCUs of the base layer.

This determination, based on expression (3) may be expressed by:

$$LCU.addr.x \bmod 3 \neq 1 \text{ and } LCU.addr.y \bmod 3 \neq 1 \quad (4)$$

where LCU.addr.x is the coordinate x of the address of the considered LCU in the enhancement layer, LCU.addr.y is the coordinate y of the LCU in the enhancement layer, and mod (3) is the modulo operation providing the reminder of the division by 3.

Once the result of the above test is obtained, then the coder or decoder is able to known which LCU's and which coding units inside these LCU's should be considered in the next steps of the algorithm of **Figure 33**.

In case of a positive test at step S3301, i.e. the current LCU of the base layer is fully covered by an up-sampled LCU of the base layer, then only one LCU in the base layer is concerned by current LCU in the enhancement image. This base layer LCU is determined as a function of the spatial coordinates of current enhancement layer LCU by the following expression:

$$BaseLCU.addr.x = LCU.addr.x * 2/3 \quad (5)$$

$$BaseLCU.addr.y = LCU.addr.y * 2/3 \quad (6)$$

where *BaseLCU.addr.x* represents the x co-ordinate of the spatially co-located coding unit of the base image and *BaseLCU.addr.y* represents the y co-ordinate of the spatially co-located coding unit of the base image. By virtue of the obtained coordinates of the base LCU, the raster scan index of that LCU can be obtained:

(BaseLCU.addr.x/LCUWidth)+(PicHeight/LCUWidth)(BaseLCU.addr.y/LCUHeight)* (7)

Then in step S3303 the current enhancement layer LCU is divided into four Coding Units of equal sizes, noted *subCU,* providing the set S of coding units:

$$S = \{subCU_0, subCU_1, subCU_2, subCU_3\} \quad (8)$$

The next step of the algorithm of **Figure 33** involves a loop on each of these coding units. For each of these coding units, the algorithm of **Figure 34** is invoked at step S3315, in order to perform the prediction information derivation.

In the case where the test of step S3301 leads to a negative result, i.e. the current LCU of the base layer is not fully covered by a single up-sampled LCU of the base layer, then this means the region of the base layer, spatially corresponding to the processing block (LCU) of the enhancement layer, overlaps several largest coding units (LCU) of the base layer in their up-scaled version. The algorithm of **Figure 33** then proceeds from step S3312 to step S3314. In step S3312 the LCU of size 64x64 of the enhancement layer is split into a set *S* of four sub coding units of size 32x32: $S = \{subCU_0 \ldots subCU_3\}$. In subsequent step S3313 the first sub coding unit $subCU_0$ is taken from the set S for further processing in step S3314.

Since the enhancement LCU is overlapped by at least two base LCU areas in their up-sampled version, the each subCU of the set S may belong to a different LCU of the base image. As a consequence, the next step of the algorithm of **Figure 33** involves determining, for each coding *subCU* in set *S,* the largest coding unit of the base layer that is concerned by that *subCU*. In step S3314 for each sub coding unit *subCU* of set *S* the collocated coding unit CU in the base layer is obtained:

$$BaseLCU.addr.x = subCU.addr.x*2/3 \quad (9)$$

$$BaseLCU.addr.y = subLCU.addr.y*2/3 \quad (10)$$

By virtue of the obtained coordinates of the base LCU, the raster scan index of that LCU is obtained:

(BaseLCU.addr.x/LCUWidth)+(PicHeight/LCUWidth)(BaseLCU.addr.y/LCUHeight)*

(11)

In step S3315 the prediction information derivation algorithm of **Figure 34** is called in order to derive the prediction information for the current sub coding unit of step S3304 or step S3314 from the collocated largest coding unit LCU in the base image.

In step S3316 it is determined if the last sub coding unit of set S has been processed. The process returns to step S3314 or S3315 through step S3318 depending on the result of test S3301 so that all the sub coding units of set S are processed and ends in step S3317 when all the sub-coding units S have been processed for the enhancement processing block LCU.

The method of deriving the prediction information from the collocated largest coding unit of the base layer, in step S3315 of **Figure 33**, is illustrated in the flow chart of **Figure 34**.

In step S3401 it is determined if the current coding unit has a size greater than 2x2. If not the method proceeds to step S3402 where the current coding unit is assigned a prediction unit type 2Nx2N and the prediction information is derived for the prediction unit b_{2x2} in step S3403.

Otherwise, if it is determined that the current coding unit has a size NxN greater than 2x2, for example 32x32, then, in step S3412 the current coding unit is split into a set S of four sub coding units of size N/2xN/2, 16x16 in the example,: $S=\{subCU_0 \ldots subCU_3\}$. The first sub-coding unit $subCU_0$ is then selected for processing in step S3413 and each of the sub-processing units are looped through for processing in steps S3414 and S3415. Step S3414 involves a recursive call to the algorithm of **Figure 34** itself. Therefore, the algorithm of **Figure 34** is called with the current coding unit *subCU* as the input argument. The recursive call to the algorithm then aims at processing the coding units in their successively reduced size, until the minimal size 2x2 is reached.

When the test of step S3401 indicates that the input coding unit *subCU* to the algorithm of **Figure 34** has the minimal size 2x2, then an effective inter-layer prediction information derivation process takes place at steps S3402 and S3403. Step S3402 involves giving current coding unit *subCU* the prediction unit type 2Nx2N, signifying that the considered coding unit is made of one single prediction

unit. Then, step S3403 involves computing the prediction information that will be attributed to current coding unit *subCU*. To do so, the 4x4 block in the base image that is co-located with the current coding unit is searched for in the base image, as a function of the scaling ratio, which in the present example is 1.5, that links the base and enhancement images. The prediction information of the found co-located 4x4 blocks is then transformed towards the spatial resolution of the enhancement layer. Mostly, this involves multiplying the considered base motion vector by the scaling factor, 1.5. Other prediction information parameters may be assigned, without transformation, to the enhancement 2x2 coding unit.

When the inter-layer prediction information derivation is done, the algorithm of **Figure 34** ends and the method returns to the process that called it, i.e. step S3315 of **Figure 33** returning to step S3415 of the algorithm of **Figure 34**, which loops to the next coding unit *subCU* to process at the considered recursive level. When all CU's at the considered recursive level are processed, then the algorithm of **Figure 34** proceeds to step S3416.

In step S3416 it is determined whether or not the sub coding units of the set S all have equal derived prediction information with respect to each other. If not, the process ends. In the case where the prediction information is equal, then the coding units in set S are merged together in step S3417, in order to form one single coding unit of greater size. The merging step involves assigning a size to the merged CU that is twice the size of the initial coding units in width and height. In addition, with respect to derived motion vectors and other prediction information, the merged CU is given, the prediction information values that are commonly shared by the four coding units being merged. Once the merging step S3417 is done, the algorithm of **Figure 34** ends.

As has already been explained, the mechanisms of **Figures 33** and **34** are dedicated to the inter-layer derivation of prediction information in the case of a scaling factor 1.5 between the base and the enhancement layer.

In the case of SNR scalability the inter-layer derivation of prediction information is trivial. The derived prediction information corresponds to the prediction information of the coded base image.

Once the prediction information of the base image has been derived towards the spatial resolution of the enhancement layer, the derived prediction information can be used, in particular to construct the so-called base mode

prediction image. The base mode prediction image is used later on in the prediction coding/decoding of the enhancement image.

The following depicts a construction of the base mode prediction image, in accordance with one or more embodiments of the invention. In the case of temporal residual data derivation for the computation of a Base Mode prediction image the temporal residual texture coded and decoded in the base layer is inherited from the base image, and is employed in the computation of a Base Mode prediction image. The inter-layer residual prediction used involves applying a bi-linear interpolation filter on each INTER prediction unit contained in the base image. This bi-linear interpolation of temporal residual is similar to that used in H.264/SVC.

According to an alternative embodiment, the residual data that is derived may be computed in a different way. Instead of taking the decoded residual data and up-sampling it, it may comprise re-calculating a new residual data block between reconstructed base images. Technically, the difference between the decoded residual data in the base mode prediction image and such a re-calculated residual would involve the following. The decoded residual data in the base mode prediction image results from the inverse quantization and then inverse transform applied to coding units in the base image. On the other hand, fully reconstructed base images have undergone some in-loop post-processing steps, which may include the de-blocking filter, Sample Adaptive Offset (SAO) and Adaptive Loop Filter (ALF). As a consequence, the reconstructed base images are of better quality in their fully post-processed versions, i.e. are closer to the original image than the image obtained just after inverse transform. Therefore, since the fully reconstructed base image are available in the proposed codec architecture, it is possible to re-calculate some residual blocks from fully reconstructed base images, as a function of the motion information of these base images. Such residual blocks differ from the residuals obtained after inverse transform, and can be advantageously employed to perform motion compensated temporal prediction during the Base Mode prediction construction process. This particular embodiment for inter-layer prediction of the residual data can be seen as analogous to the GRILP coding mode as briefly described previously in the scope of INTER prediction in the enhancement INTER image, but is dedicated to the construction of the base mode prediction image

In some embodiments, each of the enhancement layer LCUs being processed may be systematically sub divided into coding units of size 2x2. In other embodiments, only LCUs of the enhancement layer which overlap, at least partially, two or more up-sampled base layer LCUs are sub divided into coding units of size 2x2. In yet another embodiment, only LCUs of the enhancement layer which overlap, at least partially, two or more up-sampled base layer LCUs are sub divided into smaller sized coded units up until they no longer overlap more than one up-sampled base layer LCU.

Figure 35 schematically illustrates how a Base Mode prediction image is computed. This image is referred to as a Base Mode Image because it is predicted by means of the prediction information issued from the base layer 3501. The inputs to this process are as follows:

- lists of reference images e.g.3503 useful in the temporal prediction of the current enhancement image,
- prediction information e.g. temporal prediction 35A extracted from the base layer and re-sampled to the enhancement layer resolution. This corresponds to the prediction information resulting from the process of **Figure 34**
- temporal residual data issued from the base layer decoding, and re-sampled to the enhancement layer resolution e.g. inter-layer temporal residual prediction 35C. For example, the DCTIF interpolation filters of quarter-pixel motion compensation in HEVC may be used.
- base layer reconstructed image 3504.

The Base Mode image construction process comprises predicting each coding unit e.g. 3505 of the enhancement image 3500, conforming to the prediction modes and parameters inherited from the base layer.

The method proceeds as follows.

- For each LCU 3505 in the current enhancement image 3500:
  obtain the up-sampled Coding Unit representation issued from the base layer
  - For each CU contained in the current LCU
    - For each prediction unit (PU) e.g. sub coding unit, in the current coding unit
      - Predict current PU with its prediction information inherited from the base layer

The PU prediction step proceeds as follows. In case the corresponding base PU was Intra-coded e.g. base layer intra coded block 3506, then the current

prediction unit of the base mode prediction image 1200 is predicted by the reconstructed base coding unit, re-sampled to the enhancement layer resolution 3507. In practice, the corresponding spatial area in the Intra BL prediction image is copied.

In the case of an INTER coded base coding unit, then the corresponding prediction unit in the enhancement layer is temporally predicted as well, by using the motion information inherited from the base layer. This means the reference image(s) in the enhancement layer that correspond to the same temporal position of the reference images(s) of the base coding unit are used. A motion compensation step 35B is applied by applying the motion vector 3510 inherited from the base layer onto these reference images. Finally, the up-sampled temporal residual data of the co-located base coding unit is applied onto the motion compensated enhancement PU, which provides the predicted PU in its final state.

Once this process has been applied on each PU in the enhancement image, a full "Base Mode" prediction image is available.

It may be noted that by virtue of the proposed base mode prediction image illustrated in **Figure 35**, the base mode prediction mechanism employed in the proposed scalable codec has the following property.

For coding units of the enhancement image that are coded using the base mode prediction mode, the data that is predicted is the texture data only. On the contrary, in the former H.264/SVC scalable video compression system, processing blocks (macroblocks) that were encoded using a base layer prediction mode were fully inferred from the base image, in terms of prediction information and macroblock (LCU) representation. For example, the macroblocks organization in terms of splitting macroblocks LCU into sub-macroblocks CU (sub processing blocks) 8x8, 16x8, 8x16 or 4x4 was imposed as a function of the way the underlying base macroblock was split. For instance, in the case of dyadic spatial scalability, if the underlying base macroblock was of type 4x4, then the corresponding enhancement macroblocks, if coded with the base mode, was split into four 8x8 sub-macroblocks.

On the contrary, in embodiments of the present invention, the coding structure chosen in the enhancement image is independent of the coding structure representations that were used in the base layer, including for enhancement coding units using a base layer prediction mode.

This technical result comes from the fact that the base mode prediction image construction is used as an intermediate step between the base layer and the enhancement layer coding. An enhancement coding unit that employs the base mode

prediction type only makes use of the texture data contained in its co-located area in the base mode prediction image, and no prediction data issued from the base layer. Once the base mode prediction image is obtained the base mode prediction type involved in the enhancement image coding ignores the prediction information of the base layer.

As a result, an enhancement coding unit that employs the base mode prediction type may spatially overlap several coding units of the base layer, which may have been encoded by different modes.

This decoupling property of the base mode prediction type makes it different from the base mode previously specified in the former H.264/SVC standard.

With reference to **Figure 36**, a further step in the computation of a Base Mode prediction image involves de-blocking filtering the base mode prediction image. To do so, each LCU of the enhancement layer is de-blocked by considering the inter-layer derived CU structure associated with that LCU.

According to the default codec configuration, the de-blocking filter is applied only on Coding Unit boundaries but not in Transform Unit boundaries. Optionally the de-blocking can also be activated on Transform Unit boundaries. In that case, inter-layer derived Transform Units are considered.

The Quantization Parameter (QP) used during the Base Mode image de-blocking process is equal to the QP of the Co-located base CU of the CU currently being de-blocked. This QP value is obtained during the inter-layer CU derivation step of **Figures 32** to **34**.

The following description presents a deblocking filtering step applied to the base mode prediction image provided by the mechanisms of **Figure 35**. The constructed base mode prediction image is made up of a series of temporally and intra prediction units. These prediction units are derived from the base layer through the prediction information up-sampling process previously described for example with reference to **Figure 30**. Therefore, these derived prediction units (PU's) have some prediction data which differs from one enhancement prediction unit to another. As can be appreciated, some blocking artefacts may appear at the boundaries between these prediction units. The blocking artefacts so-obtained in the base mode prediction image are even stronger than those of traditional coded/decoded image in standard video coding, since no prediction error data is added to the predicted blocks contained in it.

As a consequence, it is proposed in one particular embodiment, to apply a deblocking filtering process to the base mode prediction image. According to one embodiment, the deblocking filtering step may be applied to the boundaries of inter-layer derived prediction units. In a further, more advanced, embodiment the de-blocking filter may also apply to the boundaries of inter-layer derived transform units. To do so, in the inter-layer derivation of prediction information, it is needed to additionally derive the transform unit organization from the base layer towards the spatial resolution of the enhancement layer.

Figure 36A is a flow chart illustrating the de-blocking filtering of the base mode prediction image.

For each LCU, noted *currLCU*, the transform tree is derived in step S366 for each CU of the LCU from the base layer, according to an embodiment.

Figure 36B illustrates an example of enriched inter-layer derivation of prediction information in the case of dyadic spatial scalability. The derivation process for enhancement LCUs has already been explained, concerning the derivation of coding unit quad-tree representation, prediction unit partition, and associated motion vector information. In addition, the derivation of transform unit splitting information is illustrated in **Figure 36B**. As can be seen, the transform unit splitting, also called transform tree in the HEVC standard consists in further dividing the coding units in a quad-tree manner, which provides so-called transform units. A transform unit specifies an elementary image area or block on which the DCT transform and quantization are actually performed during the HEVC coding process. Reciprocally, a transform unit is the elementary image area where inverse DCT and inverse quantization are performed on the decoder side.

As illustrated by **Figure 36B**, the inter-layer derivation of a transform tree aims at providing an enhancement coding unit with a transform tree which is the same shape as the transform tree of the co-located base coding unit.

Figure 36C and **Figure 36D** depict how the inter-layer transform tree derivation proceeds, in one embodiment, in a dyadic spatial scalability case. **Figure 36C** recalls the prediction information derivation process, applied to coding units, prediction units and motion vectors. In particular, the coding depths transformation from the base to the enhancement layer, in the case of dyadic spatial scalability, is shown. As can be seen, in this context, the derivation of the coding tree information consists in decreasing by one the depth value associated with each coding unit. With respect to base coding units that have a depth value equal to 0, hence have maximal size and

correspond to an LCU, their corresponding enhancement coding units are also assigned the depth value 0.

Figure 36D illustrates the way the transform tree is derived from the base layer towards the enhancement layer. In HEVC, the transform tree is a quad-tree embedded in each coding unit. Thus, each transform unit is fully specified by virtue of its relative depth. In other words, a transform unit with a zero depth has a size equal to the size of the coding unit it belongs to. In that case, the transform tree is made of a single transform unit.

The transform unit (TU) depth thus specifies the size of the considered TU relative to the size of the CU that it belongs to, as follows:

$$TU_{width} = CU_{width} * 2^{-TUdepth}$$

$$TU_{height} = CU_{height} * 2^{-TUdepth}$$

where (TU_{width}, TU_{height}) and (CU_{width}, CU_{height}) respectively represent size, in width and height, of the considered TU and CU, and TU_{depth} represents the TU depth.

As shown in **Figure 36D**, to obtain the same transform tree depth in the enhancement layer as in the base layer, the TU derivation simply includes providing the enhancement coding units with the same transform tree representations as in the base layer.

Once the derived transform unit is obtained, then both the encoder and the decoder are able to apply the de-blocking filtering step onto the constructed base mode image.

Back to **Figure 36A**, the following step S367 comprises obtaining a quantization parameter to use during the actual de-blocking filtering operation. In one embodiment, the QP used is equal to the QP that was used during the encoding of the base image of the current enhancement image. In another embodiment, the QP used during the encoding of current enhancement image may be considered. According to another embodiment, a mean between the two can be used. In yet a further embodiment, the enhancement image QP can be considered when de-blocking the boundaries of the derived coding units, while the QP of the base image can be employed when de-blocking the boundaries between adjacent transform units.

Once the QP used for the subsequent de-blocking filtering is obtained, this effective de-blocking filtering is applied in subsequent step S368. It is noted that the CBF parameter (flag indicated, for each coding unit, if it contains at least non-zero

quantized coefficient) is forced to zero for each coding unit during the base mode image de-blocking filtering step.

Once the last LCU in current enhancement image has been de-blocked in step S369 the algorithm of **Figure 36A** ends. Otherwise, the algorithm considers the next LCU in the image as the current LCU to process (S370), and loops to transform tree derivation step S366.

In another embodiment, the base mode image may be constructed and/or de-blocked only on a part of the whole enhancement image. In particular, this may be of interest on the decoder side. Indeed, only a part of the coding units may use the base mode prediction mode. It is possible to construct and/or de-block the base mode prediction texture data only for an image area that at least covers these coding units. Such image area may consist, in a given embodiment, in the spatial area co-located with current LCU being processed. The advantage of such approach would be to save some memory and complexity, as the motion compensated temporal prediction and/or de-blocking filtering is applied on a sub-part of the image.

According to one embodiment, such an approach with reduced memory and complexity takes place only on the decoder side, while the full base mode prediction image is computed on the encoder side.

According to yet another embodiment, the partial base mode image computing is applied both on the encoder and on the decoder side.

Once the Base Mode prediction image has been generated for the current enhancement INTER image to encode, the Base Mode prediction mode consists in predicting a block of the enhancement INTER image from the co-located block in the Base Mode prediction image. The obtained residual is then encoded using convention H.264/AVC or HEVC mechanisms, or LCC mechanisms.

A rate distortion cost of the Base Mode prediction mode may be based on the cost function:

$$C = D + \lambda_{final}(R_s + R_r);$$

where C is the obtained cost, D is the distortion between the original block to encode and its reconstructed version after encoding and decoding. $R_s + R_r$ represents the bitrate of the encoding, where R_s is the component for the size of the syntax element representing the coding mode (see **Figure 27**), and R_r is the component for the size of encoded data representing the encoded residual. λ_{final} is the usual Lagrange parameter.

The cost of the Base Mode predictor will be then compared to the costs of other predictors available for blocks in the enhancement INTER image to select the best prediction mode (see 75 in **Figure 7**). If the Base Mode prediction mode is finally selected, a coding mode identifier, information and the encoded residual are inserted in the bit stream.

The Generalized Residual Inter-Layer Prediction (GRILP) mode is now described with more details with reference to **Figures 37** to **41**.

As mentioned above, the GRILP mode lies on conventional INTER prediction mode. In this context, the encoding of the enhancement layer is predictive, meaning that a predictor is found in a previous (reference) image of the enhancement layer to encode a block of the enhancement INTER image. This encoding leads to the computation of a residual, called the first order residual, being the difference between the block to encode and its predictor. It may be attempted to improve the encoding by realizing a second order prediction, namely by using predictive encoding of this first order residual itself.

The enhancement INTER image to encode, or decode, is the image 37.1. This image is constituted by the original pixels. Image 37.2 in the enhancement layer is available in its reconstructed version and will serve as a reference image for inter prediction.

Regarding the base layer or any lower layer (in case of multiple enhancement layers), it depends on the scalable decoder architecture considered. If the encoding mode is single loop, meaning that the base layer reconstruction is not brought to completion, the image 37.4 is composed of inter blocks decoded until obtaining their residual but on which is not applied the motion compensation and of intra blocks which may be integrally decoded or partially decoded until obtaining their intra prediction residual and a prediction direction. Note that in **Figure 37**, both layers are represented at the same resolution as in SNR scalability. In Spatial scalability, two different layers will have different resolutions which require an up-sampling (as described above for example) of the residual and motion information before performing the prediction of the residual.

In case the encoding mode is multi loop, a complete reconstruction of the base layer is conducted. In this case, the image 37.4 of the previous image and the image 37.3 of the current image (i.e. temporally coinciding with the enhancement

INTER image to encode) both in the base layer are available in their reconstructed version.

In a first embodiment we describe a first version of the GRILP adapted to temporal prediction in the enhancement layer. This embodiment starts with the determination of the best temporal GRILP predictor in a set comprising several potential temporal GRILP predictors obtained using a block matching algorithm.

In a first step 38.1, a predictor candidate contained in the search area of the motion estimation algorithm is obtained for block 37.5. This predictor candidate represents an area of pixels 37.6 in the reconstructed reference enhancement image 37.2 in the enhancement layer pointed by a motion vector 37.10. A difference between block 37.5 and block 37.6 is then computed to obtain a first order residual block in the enhancement layer. For the considered reference area 37.6 in the enhancement layer, the corresponding co-located area 37.12 in the reconstructed reference base image 37.4 in the base layer is identified in step 38.2. In step 38.3 a difference is computed between block 37.8 (co-located with block 37.5 in the base layer) and block 37.12 to obtain a first order residual block for the base layer. In step 38.4, a prediction of the first order residual block of the enhancement layer by the first order residual block of the base layer is performed. During this prediction, the difference between the first order residual block of the enhancement layer and the first order residual block of the base layer is computed. This last prediction allows obtaining a second order residual. It is to be noted that the first order residual block of the base layer does not correspond to the residual used in the predictive encoding of the base layer which is based on the predictor 37.7. This first order residual block is a kind of virtual residual obtained by reporting in the base layer the motion vector obtained by the motion estimation conducted in the enhancement layer. Accordingly, by being obtained from co-located pixels, it is expected to be a good predictor for the residual obtained in the enhancement layer. To emphasize this distinction and the fact that it is obtained from co-located pixels, it will be called the co-located residual in the following.

In step 38.5, the rate distortion cost of the GRILP mode under consideration is evaluated. This evaluation is based on a cost function depending on several factors. An example of such a cost function is :

$$C = D + \lambda_{final}(R_s + R_{mv} + R_r);$$

where C is the obtained cost, D is the distortion between the original block to encode and its reconstructed version after encoding and decoding. $R_s + R_{mv} + R_r$ represents the bitrate of the encoding, where R_s is the component for the size of the

syntax element representing the coding mode (see **Figure 27**), R_{mv} is the component for the size of the encoding of the motion information, and R_r is the component for the size of the second order residual. λ_{final} is the usual Lagrange parameter.

In step 38.6, a test is performed to determine if all predictor candidates contained in the search area have been tested. If some predictor candidates remain, the process loops back to step 38.1 with a new predictor candidate. Otherwise, all costs are compared during step 38.7 and the predictor candidate minimizing the rate distortion cost is selected.

The cost of the best GRILP predictor will be then compared to the costs of other predictors available for blocks in the enhancement INTER image to select the best prediction mode (see 75 in **Figure 7**). If the GRILP mode is finally selected, a coding mode identifier, the motion information and the encoded residual are inserted in the bit-stream.

The decoding of the GRILD mode is illustrated by **Figure 39**. The-bit stream comprises the means to locate the predictor and the second order residual.

In a first step 39.1, the location in the enhancement layer of the predictor used for the prediction of the block to decode and the associated residual are obtained from the bit-stream. This residual corresponds to the second order residual obtained at encoding. In a step 39.2, similarly to encoding, the co-located predictor is determined in the base layer. It is the location in the base layer of the pixels corresponding to the predictor obtained from the bit-stream. In a step 39.3, the co-located residual is determined. This determination may vary according to the particular embodiment similarly to what is done in encoding. In the context of multi loop and inter encoding it is defined by the difference between the co-located block and the co-located predictor in the base layer. In a step 39.4, the first order residual block is reconstructed by adding the residual obtained from the bit-stream which corresponds to the second order residual and the co-located residual from the base layer. Once the first order residual block has been reconstructed, it is used with the predictor which location has been obtained from the bit-stream to reconstruct the block in a step 39.5.

In an alternative embodiment allowing reducing the complexity of the determination of the best GRILP predictor, it is possible to perform the motion estimation in the enhancement layer without considering the prediction of the first order residual block. The motion estimation becomes classical and provides a best temporal predictor in the enhancement layer. In **Figure 38**, this embodiment consists in replacing step 38.1 by a complete motion estimation step determining the best

temporal predictor among the predictor candidates in the enhancement layer and by removing steps 38.6, 38.7 and 38.8. All other steps remain identical and the cost of the GRILP mode is then compared to the costs of other modes.

By default, during the computation, the following images are stored in memory, the current enhancement INTER image to encode in the enhancement layer, the previous (reference) image in the enhancement layer in its reconstructed version, the current corresponding base image in the base layer in its reconstructed version and the previous (reference) image in the base layer in its reconstructed version. The base images of the base layer are typically up-sampled to fit the resolution of the enhancement layer.

Advantageously, the blocks in the base layer are up-sampled only when needed instead of up-sampling the whole base image at once. The encoder and the decoder may be provided with on-demand block up-sampling means to achieve the up-sampling. This may be implemented for all or part of the images of the base layers (i.e. current base image and reference base images). For example, only the blocks of the reference base images can be obtained on-demand while the current base image is stored in an up-sampled version to offer quick access. Alternatively, to save some computation, the up-sampling is done on the block data only. The decoder must use the same up-sampling function to insure proper decoding. It is to be noted that all the blocks of an image are typically not encoded using the same coding mode. Therefore, at decoding, only some of the blocks are to be decoded using the GRILP mode. Using on-demand block up-sampling means is then particularly advantageous at decoding as only some of the blocks of a base image have to be up-sampled during the process.

In another embodiment, which is advantageous in terms of memory saving, the first order residual block in the base layer may be computed between reconstructed images which are not up-sampled, thus are stored in memory at the spatial resolution of the base layer.

The computation of the first order residual block in the base layer then includes a down-sampling of the motion vector considered in the enhancement layer, towards the spatial resolution of the base layer. The motion compensation is then performed at reduced resolution level in the base layer, which provides a first order residual block predictor at reduced resolution.

Last inter-layer residual prediction step then consists in up-sampling the so-obtained first order residual block predictor, through a bi-linear interpolation filtering for instance. Any spatial interpolation filtering could be considered at this step of the

process (examples: 8-Tap DCTIF, 6-tap DCT-IF, 4-tap SVC filter, bi-linear). This last embodiment may lead to slightly reduced coding efficiency in the overall scalable video coding process, but does not need additional reference image storing compared to standard approaches that do not implement the present embodiment.

An alternative embodiment is now described in the context of single loop decoding in relation with **Figure 40.** Single loop decoding means that the reconstruction process of the base (or lower) layer is not brought to completion. The only data available for the base layer is therefore the image of the residual 40.4 used for the encoding. **Figure 40** is adapted from **Figure 37** in which image 40.4 is composed of residual instead of the reconstructed version of the previous image in the base layer. References with the same secondary number correspond to the same element in **Figure 37.** The process is globally the same as the one conducted for multi loop decoding configuration. When coming to step 38.3 in **Figure 38**, the way the co-located residual is determined need to be adapted. Block 40.12 is the co-located predictor in the residual image 40.4 corresponding to the predictor 40.6 in the reference image 40.2 in the enhancement layer. Block 40.7 is the block corresponding to the location of the predictor found in the base layer. The case where these two blocks overlap is considered. The overlap defines a common part 40.13 represented as dashed in **Figure 40**. The residual values pertaining to this common part 40.13 are composed with the residual of the prediction of the corresponding dashed part 40.14 in the current image of the base layer. This prediction has been done according to the motion vector 40.9 found in the base layer. The dashed part 40.14 that has been predicted pertained to the block 40.8 which is co-located with the block 40.5 to encode in the enhancement layer.

This common part 40.13 is co-located with the dashed part 40.16 of the predictor 40.6, this dashed part 40.16 of the predictor being used for the prediction of the dashed part 40.15 of the block 40.5 to encode. The prediction of block 40.5 based on the predictor 40.6 generates a first order residual block. It is to be noted that residuals in the co-located predictor 40.12 outside the common part are related to pixels outside the co-located block 40.8. In the same manner, residuals in the predictor 40.7 pixels outside the common part are related to predictor pixels outside the predictor 40.6. It comes that the only relevant part of the residuals available in the residual image 40.4 for the prediction of the first order residual block obtained in the enhancement layer is the common part. This common part constitutes the co-located residual that is used for the prediction of the co-located part of the first order residual block in the

enhancement layer. In this embodiment, the prediction of the residual is realized partially for the part corresponding to the overlap in the residual image in the base layer between the co-located predictor and the predictor used in the base layer when this overlap exists. Other parts of block 40.5 not corresponding to the part 40.15 corresponding to the overlap are predicted directly by predictor 40.6.

Figure 41 illustrates an embodiment in the context of intra prediction. In intra prediction coding, the current enhancement INTER image 41.1 is the only one used for encoding a block 41.2. Base image 41.5 is the image that temporally coincides in the base layer.

The process works according to the same principle except temporal predictors are replaced by spatial predictors. Predictors are blocks of the same size as block 41.2 to encode obtained using a set of neighbour pixels as conventionally known. As illustrated in **Figure 41**, the prediction in the enhancement layer taking into account a spatial GRILP prediction, has determined predictor pixels 41.3 and a prediction direction 41.4. The prediction direction plays the role of the motion vector in the inter prediction coding. They constitute both means to locate the predictor. The encoding of the base layer has determined for the co-located block 41.6 pixel predictor 41.7 and a prediction direction 41.8. The co-located predictor 41.9 is determined in the base layer with the corresponding prediction direction 41.10. Similarly to the inter prediction coding, the prediction direction and the predictor obtained in the different layers may be correlated or not. For the sake of clarity, **Figure 41** illustrates a case where both the predictor and the prediction direction are clearly different.

Similarly to the method described for inter prediction coding, the co-located residual is computed in the base layer as the difference between the co-located block 41.6 and the predictor obtained from the co-located border pixels 41.9 using the prediction direction 41.10 determined in the enhancement layer. This co-located residual is used as a predictor for the first order residual block obtained in the enhancement layer. This prediction of the first order residual block leads to a second order residual which is embedded in the bit-stream as the result of the encoding of block 41.2.

Considering the single loop mode, two cases occur.

In a first case the following conditions are considered. First the constrained intra prediction mode is selected. Then INTRA prediction is allowed only from INTRA blocks that are completely decoded by the decoder. If blocks containing the pixels used for intra prediction in the enhancement layer are co-located with the blocks containing

the pixels used for intra prediction in the base layer then, all pixels used for intra prediction are reconstructed pixels and the process is the same than in the multi-loop decoding.

In a second case, when these conditions are not fulfilled, meaning that some blocks containing the pixels used for intra prediction in the enhancement layer are not co-located with the blocks containing the pixels used for intra prediction in the base layer or if we are not in the constrained intra prediction mode, then inter layer prediction of the residual will be possible only if the prediction direction in the enhancement layer and the prediction direction in the base layer are the same.

Figure 42 illustrates an algorithm used to encode an INTER image. The input to the algorithm comprises the original image to encode, respectively re-sampled to the spatial resolution of each scalability layer to encode.

In what follows the term "base layer" can be used to designate a reference layer used for inter layer prediction. This terminology is adapted to the case where a scalable coder generates 2 layers. However, it is well known that for a coder generating more than 2 layers, any layer lower than the layer to be encoded can be used for inter layer prediction. It may be noted that in general, the layer immediately below the layer to encode is used.

The overall algorithm includes a loop over each scalability layer to encode. The current INTER image is being encoded with each scalability layer being successively or sequentially processed through the algorithm. The layers are indexed 4202. For each scalability layer in succession, the algorithm tests 4203 if current layer corresponds to the base layer, the base layer being indexed as layer 0 (zero). If so, then a standard image encoding process is applied on the current image. For the case illustrated in **Figure 42**, the base image is HEVC-encoded 4204.

When a current layer is not the base layer (e.g. is a first enhancement layer), the algorithm switches to preparing all the prediction data useful to predict current enhancement INTER image to code, according to embodiments. This data includes three main parts:

- the corresponding decoded base image is obtained 4205 and up-sampled 4206 in the pixel domain towards the spatial resolution of current enhancement layer. This provides one prediction image, called the "Intra BL" prediction image. This also provides reference images of the base layer to compute the base residual to be added in GRILP mode.

- all the prediction information contained in the coded base layer is extracted from the base image 4207, and then is up-sampled 4208 towards current enhancement layer, e.g. as previously explained with reference to **Figure 30**. Next, this up-sampled prediction info is used in the construction of the "Base Mode" prediction image 4209 of current enhancement INTER image, as previously explained with reference to **Figure 35**. Up-sampled prediction info is also used to supplement the set of vector predictors in the inter-layer motion vector prediction mode if activated.
- temporal residual data contained in the base image is extracted from the base layer 4210, and then is up-sampled 4211 towards the spatial resolution of current enhancement layer.

Next, the up-sampled prediction info, together with this up-sampled temporal residual data, are used in the construction of the "Base Mode" prediction image of current enhancement image, as previously explained with reference to **Figure 35**.

The next step of the algorithm includes searching the best way to predict each block of the current enhancement INTER image, given the available set of prediction data previously prepared. The algorithm performs the best prediction search 4212 based on obtained prediction blocks from temporal reference(s) with GRILP activated or not and with Inter-layer MV prediction activated or not, Intra BL prediction image, Base Mode prediction image, currently decoded blocks for INTRA prediction.

This step corresponds to the rate distortion optimal mode decision described above, for example based on minimization of a lagrangian cost function.

For a candidate block to encode, the best candidate prediction block for that block is thus selected. Finally the best block splitting configuration (see **Figure 5A**) for the considered LCU is selected.

Once the prediction search for current image is done, then a set of prediction information is available for current image. This prediction information is able to fully describe how current enhancement INTER image is segmented and predicted. Therefore, this prediction information is encoded 4213 and written to the output enhancement bit-stream, in order to indicate the decoder how to predict current image.

In the next step of the algorithm of **Figure 42**, each obtained prediction block is subtracted from the corresponding original block to code in current enhancement INTER layer i.e. the residual image for the current image is progressively obtained 4214.

In parallel, the next step comprises applying the HEVC texture coding on each residual block of the residual image 4215 issued from previous step.

Once the current image is encoded at the current scalability level, then the algorithm checks whether current layer is the last scalability layer to encode 4216. If yes, then the algorithm ends 4217. If no, the algorithm moves to process the next scalability layer, i.e. it increments current layer index 4218, and returns to the testing step 4203 described above.

Figure 43 schematically illustrates the overall algorithm used to decode an INTER image, according to at least one embodiment. The input to this algorithm includes the compressed representations of the input image, comprising a plurality of scalability layers to be decoded, indexed as 4302.

Similar to the coding algorithm of **Figure 42**, this decoding algorithm comprises a main loop on the scalability layers that constitutes the scalable input bit-stream to process.

Each layer is considered sequentially, the following is applied. The algorithm tests 4303 if a current layer corresponds to the lowest layer of the stream, the base layer normally being assigned a value 0 (zero). If so, then a standard, e.g. HEVC, decoding process is applied 4304 on current image.

If not, then the algorithm prepares all the prediction data useful to construct the prediction blocks for current enhancement INTER image. Thus the same base layer data extraction and processing as on the encoder side is performed (4205 to 4211). This leads to restoration of the set of prediction data schemes used to construct the prediction image of current enhancement image. This is facilitated by computation of the same Intra BL, Base Mode and temporal reference (with or without GRILP) prediction images.

The next step of the algorithm comprises decoding the prediction information for the current image from the input bit-stream 4305. This provides information on how to construct the current prediction image 4306, given the Intra BL, Base Mode and temporal reference images available.

The decoded prediction data (quad-tree) thus indicates how each LCU is decomposed into coding units (CU) and prediction units (PU), and how each prediction unit is predicted. The decoder is then able to construct the corresponding prediction blocks. At this stage of the decoder, exactly the same prediction blocks as on the encoder side are available.

The next step comprises the texture decoding of the input coded texture data on the current residual image 4307. The same decoding algorithm is applied as described with reference to **Figure 12**.

Once the decoded residual image is available, the obtained residual image is added to the prediction blocks previously computed 4308, which provides the reconstructed version of current enhancement image.

Additionally it is possible to follow this with post-processing of current image (not shown), i.e. a deblocking filter, sample adaptive offset and adaptive loop filtering.

Finally, the algorithm tests if current scalability layer is the last layer to decode 4309. If so, the algorithm of **Figure 42** ends 4310. If not, the algorithm increments the layer 4311 and returns to the testing step 4303, which checks if the current layer is the base layer.

Back to the encoding and decoding of motion information (in particular motion vectors), the next figures illustrate the particular motion vector prediction mechanism for the enhancement layer according to the invention.

Figure 44 shows a schematic of the AMVP predictor set derivation for an enhancement image of a scalable codec of the HEVC type according to a particular embodiment.

According to this particular embodiment, the standard process of AMVP predictor set derivation is applied to the base layer.

It is to be noted that determination of the motion estimation predictors that are to be used for encoding or decoding an enhancement image is based on temporal and spatial motion information predictors that can be used with regard, in particular, to the determination of motion estimation predictors for the base image (e.g. the order of temporal and spatial predictor of the base image and the available predictors of the base image) so as to improve coding efficiency.

As depicted in **Figure 44**, the same spatial positions A0, A1, B0, B1, and B2 (4400 to 4408) (as shown in **Figure 45**) as the ones used in the standard derivation process of motion vector predictors in AMVP are used to derive two spatial predictors. However, if the positions of the spatial predictors are the same, their order in the list of motion vector predictors is different.

Temporal predictor 4410 is defined as the first predictor of the list of motion vector predictors. Only the center position of the co-located block (i.e. the block at the

same position as the current enhancement block of the current enhancement original INTER image in an encoded reference image – e.g. reference image) is considered as a possible motion vector predictor (while in the standard derivation process of motion vector predictors in AMVP, applied here to the base image, the bottom right position and the center position are used.

The availability of a motion vector corresponding to the center position of the co-located block is checked (4412) as done in the standard derivation process of motion vector predictors in AMVP and scaled (4414) if required. This motion vector predictor is scaled (4414) as a function of the temporal distance between the current image and the selected reference image. If the motion vector corresponding to the center position is available, it is considered as a first predictor (*Pred_1*, 4416).

Next, the left blocks A0 and A1 (4400, 4402) are selected to derive, if it is possible, a first spatial predictor. After having checked the availability of the motion vectors, the following conditions are evaluated (4418) in the specific order of the selected blocks and then of the conditions, the first block whose conditions are fulfilled being used as a predictor:

- the motion vector from the same reference list and the same reference image;
- the motion vector from the other reference list and the same reference image;
- the scaled motion vector from the same reference list and a different reference image; or
- the scaled motion vector from the other reference list and a different reference image.

If no value is found, the left predictor is considered as being unavailable. In this case, this indicates that the related blocks were Intra coded or those blocks do not exist. On the contrary, if a predictor is identified, it is considered as a second predictor (*Pred_2*, 4420).

Next, the top blocks B0, B1, and B2 (4404, 4406, and 4408) are selected to derive, if it is possible, a second spatial predictor. Again, after having checked the availability of the motion vectors, the above conditions are evaluated (4422) in the specific order of the selected blocks and then of the conditions, the first block whose conditions are fulfilled being used as a predictor.

Again, if no value is found, the top predictor is considered as being unavailable. In this case, this indicates that the related blocks were Intra coded or that those blocks do not exist. On the contrary, if a predictor is identified, it is considered as a third predictor (*Pred_3*, 4424).

In a particular embodiment, the check of availability for left blocks A0 and A1 and top blocks B0, B1, and B2 is modified as follows:

if no motion vector exist for a given neighbouring block (A0, A1, B0, B1, B2), the base block (AL0, AL1, BL0, BL1, BL2) spatially corresponding to the given neighbouring block in the corresponding base image is considered. The availability of the motion vector corresponding to the center position of this base block is checked before it is added to the set of motion vector predictors, after up-scaling if needed.

Next, a fourth predictor (*Pred_4*, 4430), referred to as a base layer (BL) predictor, is determined (if possible). To that end, the bottom right (BR) position of the co-located block in the base image is selected (4426) and the availability of the corresponding motion vector is checked (4427). As it belongs to the base layer, this motion vector predictor (BL) is firstly scaled as a function of the spatial ratio between the base layer and the enhancement layer as described above (e.g. **Figures 30** to **34**). In this embodiment, the motion vector predictor corresponds to a base block of the base layer that is not co-located with the current enhancement block for which the motion vector has to be encoded.

In addition and if needed, this motion vector predictor is scaled (4428) as a function of the temporal distance between the current image and the selected reference image.

In a variant to selecting the BR position, the co-located block in the base image is selected (4426) and the availability of the motion vector corresponding to the center position of this co-located block is checked (4427) before it is added to the set of motion vector predictors. If needed, the motion vector predictor is scaled (4428).

Figure 45 illustrates spatial and temporal blocks, in particular the bottom right block of the base image, the AL0, AL1, BL0, BL1, BL2 base blocks (which may have different sizes) that can be used to generate motion vector predictors in AMVP and Merge modes of scalable HEVC coding and decoding systems according to a particular embodiment.

Returning to **Figure 44**, a test is performed, in a following step (4432), to remove duplicate predictors amongst the four possible predictors (*Pred_1* to *Pred_4*). To that end, the available motion vectors are compared with each other.

Next, if the number of remaining predictors (*Nb_Pred*) is greater than or equal (4434) to the maximum number of predictors (*Max_Pred*), e.g. three in this particular embodiment, the resulting predictors form the ordered list or set of motion vector predictors (4438).

On the contrary, if the number of remaining predictors (*Nb_Pred*) is smaller than the maximum number of predictors (*Max_Pred*), a zero predictor is added (4436) to the resulting predictors to form the ordered list or set of motion vector predictors (4438). The zero predictor is a motion vector equal to (0,0).

As illustrated in **Figure 44**, the ordered list or set of motion vector predictors (4438) is built, in particular, from a subset of temporal predictor (4410) and from a subset of spatial predictors (4400 to 4408), and from a predictor coming from the base layer (4426). The subset of spatial predictors and the predictor coming from the base layer are preferably considered as being part of a single subset.

Figure 46 shows a schematic of the derivation process of motion vectors for an enhancement image of a scalable codec of the HEVC type, according to a particular embodiment, for the Merge modes (including GRILP sub-mode).

According to this particular embodiment, the standard process of derivation motion vectors for Merge modes, as described in HEVC, is applied to the base layer.

Again, it is to be noted that determination of the motion estimation predictors that are to be used for encoding or decoding an enhancement image is based on temporal and spatial motion information predictors that can be used with regard, in particular, to the determination of motion estimation predictors for the base image (e.g: the order of temporal and spatial predictor of the base image and the available predictors of the base image) so as to improve the coding efficiency.

In a first step, a temporal predictor (*Cand_1*, 4622) is obtained, if possible, and set as a first candidate in the list of motion vector candidates. Compared to the HEVC base layer, only the center position (4616) of the co-located block in the corresponding temporal enhancement encoded image is processed. Availability checking and scaling steps 4618 and 4620 are as described above.

In following steps of the derivation process, the five spatial block positions A1, B1, B0, A0, and B2 (4600 to 4608) are considered. The availability of the spatial motion vectors is checked and at most five motion vectors are selected (4610). A predictor is considered as available if it exists and if the block is not Intra coded. Therefore, selecting the motion vectors corresponding to the five blocks as candidates is done according to the following conditions:

- if the "left" A1 motion vector (4600) is available (4610), i.e. if it exists and if this block is not Intra coded, the motion vector of the "left" block is selected and used as a first candidate in the list of candidates (4614);

- if the "top" B1 motion vector (4602) is available (4610), the candidate "top" block motion vector is compared to "left" A1 motion vector (4612), if it exists. If B1 motion vector is equal to A1 motion vector, B1 is not added to the list of spatial candidates (4614). On the contrary, if B1 motion vector is not equal to A1 motion vector, B1 is added to the list of spatial candidates (4614);
- if the "top right" B0 motion vector (4604) is available (4610), the motion vector of the "top right" is compared to B1 motion vector (4612). If B0 motion vector is equal to B1 motion vector, B0 motion vector is not added to the list of spatial candidates (4614). On the contrary, if B0 motion vector is not equal to B1 motion vector, B0 motion vector is added to the list of spatial candidates (4614);
- if the "bottom left" A0 motion vector (4606) is available (4610), the motion vector of the "bottom left" is compared to A1 motion vector (4612). If A0 motion vector is equal to A1 motion vector, A0 motion vector is not added to the list of spatial candidates (4614). On the contrary, if A0 motion vector is not equal to A1 motion vector, A0 motion vector is added to the list of spatial candidates (4614); and
- if the list of spatial candidates doesn't contain four candidates, the availability of "top left" B2 motion vector (4608) is checked (4610). If it is available, it is compared to A1 motion vector and to B1 motion vector. If B2 motion vector is equal to A1 motion vector or to B1 motion vector, B2 motion vector is not added to the list of spatial candidates (4614). On the contrary, if B2 motion vector is not equal to A1 motion vector or to B1 motion vector, B2 motion vector is added to the list of spatial candidates (4614).

At the end of this stage, the list of spatial candidates comprises up to four spatial candidates (*Cand_2* to *Cand_5*).

When the temporal candidate (4622) and the up to four spatial candidates (4614) are generated, a base layer merge motion candidate (*Cand_6*, 4628) is generated.

To generate the base layer Merge motion candidate, the base layer motion vector at the bottom right position (4624) of the co-located block in the base image, as illustrated in **Figure 45**, is selected and the availability of a corresponding motion vector is checked (4626). If it is available a corresponding candidate is derived. As this base layer motion vector belongs to the base layer, this motion vector predictor (BL) used for the enhancement layer is firstly scaled as a function of the spatial ratio between the base layer and the enhancement layer.

In a variant to selecting the BR position, the co-located block in the base image is selected.

Next, if it exists, this base layer candidate is used to generate a motion (MV) base layer (BL) offset (4630) in order to create four motion vector candidates *MVo1*, *MVo2*, *MVo3* and *MVo4* (4632, 4634, 4636, and 4638). These motion vector candidates can be generated as described herein after.

After the candidates (*Cand_1* to *Cand_6* and *MVo1* to *MVo4*) have been created, a duplicate check is performed (4640) and duplicate candidates are removed. To that end, the available motion vectors are compared with each other.

Next, if the number (*Nb_Cand*) of candidates is strictly less (4642) than the maximum number of candidates (*Max_Cand*, e.g. ten in this embodiment) and if the current image is of the B type, combined candidates are generated (4644). Combined candidates are generated based on available candidates of the list of Merge motion vector predictor candidates (e.g. combined candidates can be obtained by linear combination of available candidates of the list of Merge motion vector predictor candidates). This mainly consists in combining the motion vector of one candidate of the list L0 with the motion vector of one candidate of list L1.

If the number (*Nb_Cand*) of candidates remains strictly less (4646) than the maximum number of candidates (*Max_Cand*), zero motion candidates are generated (4648) until the number of candidates of the list of Merge motion vector predictor candidates reaches the maximum number of candidates.

At the end of this process, the list or set of Merge motion vector predictor candidates for an enhancement image is built (4650).

As illustrated in **Figure 46**, the list or set of Merge motion vector predictor candidates for an enhancement image is built (4650), in particular, from a subset of temporal candidates (4616), from a subset of spatial candidates (4600 to 4608), and from a subset of base layer candidates (4624) The subset of spatial candidates and the subset of base layer candidates are preferably considered as being part of a single subset.

As described above, offset predictors *MVo1*, *MVo2*, *MVo3* and *MVo4* are included in the list of Merge candidates.

According to a particular embodiment, the offset predictors are generated by adding an offset value to one or more components of a reference motion vector such as a motion vector candidate of a motion vector associated with a neighbouring block of a block corresponding to a motion vector candidate. Therefore, an offset

predictor results from a combination of a reference motion vector and one or more offsets.

For the sake of illustration, the reference motion vector *MV(mvx, mvy)* combined with a single offset value *o* can lead to several offset predictors, for example the following ones:

MVo1(mvx + o, mvy)

MVo2(mvx, mvy + o)

MVo3(mvx + o, mvy + o)

MVo4(mvx - o, mvy)

To obtain a good coding efficiency with offset predictors, the inventors observed that the following parameters have to be carefully adapted (in particular for scalable enhancement layers):

- the reference motion vector (*MV(mvx, mvy)*);
- the offset values; and
- the number of offsets predictors.

In a particular embodiment, offset predictors are generated from base motion vector as reference, i.e., a reference motion vector is chosen from amongst motion vectors of the base layer. For example, such a reference motion vector can be the one associated with the bottom right block of the collocated block in the base layer.

Still in a particular embodiment, the offset predictors are generated from the collocated base motion vector as reference, i.e., the reference motion vector is the motion vector of the collocated block in the base layer.

Still in a particular embodiment, the offset value is added alternatively to the horizontal and vertical component of motion vectors of list L0 and its inverse value is added to the corresponding motion vectors of list L1 if the motion vectors do not have the same temporal direction. For the sake of illustration, one can consider motion information referring to two motion vectors *MVL0(mvL0x, mvL0y)* and *MVL1(mvL1x, mvL1y)*, *MVL0* being associated with list L0 and *MVL1* being associated with list L1, wherein motion vector *MVL0* refers to a backward reference image and vector *MVL1* refers to a forward reference image. According to the embodiment, if only one offset value *o* is to be used, generated offset predictors can be the following:

MVo1 (MVL0 (mvL0x + o, mvL0y) ; MVL1 (mvL1x - o, mvL1y))

MVo2 (MVL0 (mvL0x - o, mvL0y) ; MVL1 (mvL1x + o, mvL1y))

MVo3 (MVL0 (mvL0x, mvL0y + o) ; MVL1 (mvL1x, mvL1y - o))

MVo4 (MVL0 (mvL0x, mvL0y - o) ; MVL1 (mvL1x, mvL1y + o))

MVo5 (*MVL0* (*mvL0x* + *o*, *mvL0y* + *o*) ; *MVL1* (*mvL1x* - *o*, *mvL1y* - *o*))

MVo6 (*MVL0* (*mvL0x* - *o*, *mvL0y* - *o*) ; *MVL1* (*mvL1x* + *o*, *mvL1y* + *o*))

MVo7 (*MVL0* (*mvL0x* - *o*, *mvL0y* + *o*) ; *MVL1* (*mvL1x* + *o*, *mvL1y* - *o*))

MVo8 (*MVL0* (*mvL0x* + *o*, *mvL0y* - *o*) ; *MVL1* (*mvL1x* - *o*, *mvL1y* + *o*))

In a particular embodiment, the absolute offset value *o* added to each component is always the same and it is equal to 4 whatever the scalability ratio.

Still in a particular embodiment, the absolute offset value *o* is equal to two and it is multiplied by the scalability ratio. For example, for an enhancement layer whose size is twice that of the base layer, the offset value *o* to be used is equal to four (4 = 2 x 2). Similarly, if the spatial ratio is equal to 1.5, the offset value is 3. Still for the sake of illustration, regarding SNR scalability (according to which the base layer and an enhancement layer are of the same size, i.e. the spatial ratio is one), the offset value is set to two.

In a particular embodiment, the offset value depends on the value of the base motion vector. For example, the sum of absolute value of horizontal and vertical components is multiplied by a fixed value to obtain the offset value to be used. As described above, this offset value *o* can be added alternatively to the horizontal and vertical components. Likewise, this particular embodiment can be combined with one or several other embodiments described above.

Still in a particular embodiment, two offset values *ox* and *oy* are computed, one being associated with each component. Again, the value of each offset can be computed as a function of the value of the component. For example, offset values *ox* and *oy* can be computed according to the following formula:

$ox = mvx / c + 1$ and $oy = mvy / c + 1$

where *c* is a constant value.

Offset values *ox* and *oy* can be computed according to the following formula:

$ox = mvx / mvy$ and $oy = mvy / mvx$

These offsets can be alternatively added to their respective components. When two lists are considered, four offsets can be computed one for each list and each component. These offset can be added alternatively by taking into account the direction (forward and backward) to determine the sign.

As described above by reference to **Figures 44** and **46**, the order of the predictors and of the candidates in the lists of predictors and of candidates is not the same for the base layer and the enhancement layers.

In particular, temporal predictors and candidates (references 4416 and 4622 of **Figures 44** and **46**, respectively) are considered as first predictors and candidates for processing enhancement images since the inventors observed that such order leads to improve coding efficiency (compared to using spatial predictors or candidates at the first rank in the list). This is because entropy or arithmetic coding for coding the index of the selected motion vector predictor allocates shortest codes to first predictors in the list.

Regarding the enhancement layer, several temporal prediction and Inter and Inter layer modes can be used, in particular the Base mode prediction mode which competes with the AMVP and Merge modes to exploit temporal correlation of pixels.

The Base mode prediction mode mainly consists in deriving the encoding syntax of enhancement layers from the encoding syntax of the base layer. Therefore, its coding efficiency is close to the one provided by the use of spatial predictors. Indeed, the base layer prediction can be considered as a spatial prediction based on a co-located block of base layer (and not on neighbouring predictors).

Motion vector selection according to the Base mode leads to similar selections of spatial motion vector predictors. Indeed, this mode is mostly selected when the motion activity is low and when the temporal distance between images is high. As a consequence, the spatial predictors of AMVP and Merge mode are redundant with the Base mode. Moreover, the first predictor of the set is the most selected. For example, with HEVC common test conditions, the selection of the first predictor represents sixty percent of the selections. Accordingly, it seems important to select another predictor as the first predictor for AMVP and Merge modes to provide diversity. The temporal predictor is, in that case, a good choice.

For the same reason as setting the temporal predictor at the first position is a good choice, the use of base motion vector (references 4426 and 4624 of **Figures 44** and **46**, respectively) leads to better coding efficiency when it is set at the end of the list of predictors. Indeed, the base motion vector leads to a similar block predictor as the Base mode and exactly the same block predictor for the same block size. For the same block size, if the Base mode has only one motion vector, the Merge modes can produce exactly the same block prediction. Accordingly, it is preferable to set it at the end of the list of predictors. It is also possible to select a different spatial position for this base motion vector as explained herein below.

According to a particular embodiment, offset predictors *MVo1* to *MVo4* (references 4632 to 4638 in **Figure 46**) are added to the list of candidates before the base motion vector (reference 4628 in **Figure 46**).

As described above, the motion information coding is different between AMVP and Merge modes. For AMVP mode, the motion vector predictor only predicts a value of a motion vector and a residual between this motion vector predictor and the real motion vector is inserted in the bit-stream. The Merge mode is different in that complete motion information is predicted and no motion vector residual is to be transmitted in the bit-stream. Consequently, the list order can depend, for AMVP mode, on several other parameters and so the embodiment that is directed to the Merge mode is simpler than those that are directed to AMVP mode.

In a particular embodiment, if the reference image index and the list index used for the AMVP mode are different from those used for the Base mode, the base motion vector is positioned in one of the first positions (e.g. position 1, 2 or 3).

Still in a particular embodiment, the base motion vector is ordered at the end of the list of predictors if the residual of the motion vector is equal to zero, otherwise it is positioned in one of the first positions (e.g. position 1, 2 or 3).

Still in a particular embodiment, the value of a variable is set to the sum of absolute value of the residual motion vector. If the value of this variable is less than a predetermined threshold, the base motion vector is added at the end of the list of predictors. Otherwise, it is positioned in one of the first positions (e.g. position 1, 2 or 3). In another embodiment, the base motion vector predictor is positioned in one of the possible positions according to the value of this variable. In this embodiment, the further the variable strays from zero the weaker the rank of the base layer predictor in the list of predictors is.

In still another embodiment, all criteria proposed below can be used to determine whether or not the base motion vector is to be added to the list of predictors. In that case, instead of adding the base motion vector at the end of the list of predictors, it is removed. For example, when the residual of the motion vector is equal to zero, the base motion should be removed from the Inter predictors list.

It is to be noted that the spatial positions of predictors can be taken into consideration even if these predictors come from a previous encoded or decoded image or from the base layer. This means that it is possible to consider that temporal block positions or inter layer block positions are projected in the current image.

As described above, **Figure 45** illustrates the spatial positions of motion vector predictors (references 4410, 4400 to 4408, and 4426 in **Figure 44** and references 4616, 4600 to 4608, and 4624 in **Figure 46**) that are used in the predictor/candidate derivation processes described by reference to **Figures 44** and **46**. Compared to the standard HEVC Merge and Inter predictor/candidate derivation process according to which the temporal predictor/candidate is based on two block positions, the temporal predictor is based on only one block position (the center position). Such a choice comes from the fact that the temporal predictor/candidate being the first predictor/candidate of the list, it has to represent as much as possible the motion information of the current block and so, on average, the center block is the best spatial position to represent the motion information of the current block.

For the predictor/candidate obtained from the base image, only one position is considered (bottom right position of the co-located block). Firstly, the predictor/candidate obtained from the base image should create diversity in the spatial positions. Accordingly, the bottom right position can be considered as a good choice in that it is the farthest position to the average spatial positions of predictors already in the list of predictors/candidates (e.g. references 4616 and 4600 to 4608 in **Figure 46**). Secondly, the predictor/candidate obtained from the base image should not be redundant in comparison to the predictors/candidate selection of the Base mode (according to which the base motion vector can be added, for example, at the end of the list of predictors/candidates). Indeed, for the Merge mode, if the center of the base image is used, this means that exactly the same block predictor is derived as Base mode. Consequently, the Merge modes can give exactly the same decoded block as the Base mode. Therefore, to avoid such redundancies, the base motion vector should be selected at position other than the co-located block.

Nevertheless, if the use of the bottom right position avoids producing the same block predictor as the Base mode, it is possible that the bottom right position does not change the motion information.

To handle such a case, all motion predictors/candidates can be compared to the motion information of the Base mode in order to remove those equal to it from the list.

According to another embodiment, all neighbouring blocks of the base image are checked in order to find one that is different from the base motion vector at the center position.

When the up-sampled syntax of the base image (Base mode) gives a smaller block size than the current block size, the predictor generated from the Base mode could be different because several motion vectors have been used to encode the base image. Consequently, the derivation process considers, in a particular embodiment, the block size to switch between the center position of the base image to another neighbouring position.

As mentioned above, the use or not of some predictors has to be carefully adapted for the derivation process of motion predictors in enhancement image.

Figure 47 shows an example of spatial positions of the neighbouring blocks of the current block in the enhancement image (A0, A1, B0, B1, B2) and their co-located blocks in the base image (AL0, AL1, BL0, BL1, BL2). For the sake of clarity, it is considered that all blocks in the enhancement image are of the same size and that the latter is exactly the same as the up-sampled co-located blocks of the base image.

According to a particular embodiment, the step of checking motion vector availability (references 4418 and 4422 in **Figure 44** and reference 4610 in **Figure 46**) checks, if the neighbouring blocks (A0, A1, B0, B1, B2) are encoded with the Base mode, one after another. In that case, the corresponding motion predictor/candidate is not added to the list of predictors/candidates.

For example, if A0 is encoded with the Base mode, it means that it is encoded with the motion information of AL0. If the base motion vector in the co-located block is different from AL0, it means that AL0 is not the best motion vector in the base image. Moreover, the base motion vector of the co-located block is more correlated with the current block than its neighbouring blocks (the motion vector selected for the current block is theoretically a better representation of the motion of the current block than the motion vector of its neighbouring blocks). Therefore, in such a case, AL0 being different from the co-located block, A0 does not need to be added to the list of predictors/candidates. Where AL0 is the best predictor in the base image, the base motion vector is equal to AL0. In view of the previous remark regarding the use of the co-located base motion vector which is redundant with the Base mode, A0 motion information does not need to be added. Consequently, the neighbouring motion vector of a neighbouring block (A0, A1, B0, B1, B2) does not need to be added if it is encoded with the Base mode. It is to be noted that this embodiment needs only to check the encoding mode of neighbouring blocks and so, it can be easily implemented.

According to a particular embodiment, if the motion information of a neighbouring block is equal to the base motion information, it is not added to the list of

predictors (the co-located block and not the bottom right block being considered for the base motion information).

In another particular embodiment, if the motion information of one predictor is equal to the base motion information, it is not added to the list of predictors.

Still in another embodiment, if the motion information of one predictor is equal to the base motion information or to the base motion information of a neighbouring block, it is not added to the list of predictors.

These embodiments can be extended to temporal predictors. Since the temporal predictor corresponds to the center of the co-located block and not to the bottom right block (as it is for the base image), the temporal predictor should be different from the temporal motion vector used in the derivation process of the base image.

The predictors that are strictly equal to their own base motion vector can be removed from the list of predictors.

According to a particular embodiment directed to spatial scalability, the motion vector of a predictor block is not added into the list of predictors if it does not contain a motion vector refinement in the spatial increase.

Conventional implementation of the above processes is through a computer device as shown in **Figure 1B**. As defined in HEVC HM-6.1, image samples (spatial values/pixels or transformed coefficients) are handled using 8-bit words, known as a 8-bit internal bit-depth. Similarly other data such as motion vectors are handled using 6-bit words.

In an embodiment of the invention, in the all-intra configuration (**Figure 2**) or in the random access configuration (**Figure 3** or **4**) or in both, the enhancement layer is processed with 10-bit internal bit-depth (using 10-bit words), while the base layer processing is kept with a 8-bit internal bit-depth.

This higher bit-depth at the enhancement layer applies mainly to the images samples, but may also apply to other data such as motion vectors.

This is to decrease information losses due to rounding when DCT transforming or inverse transforming the values or when up-sampling data (residual or pixel blocks, motion vectors, etc.) from the base layer without an integer ratio.

In practice, the original image is scaled to 10-bit values by multiplying each value by 4; this shifts the original 8 bits to the 8 MSBs (most significant bit) of the 10-bit word, and the two LSBs (least significant bit) are 0s.

In addition, some inter-layer prediction data from the base layer (e.g. residual or pixel blocks in the base layer, motion vectors) needs to be scaled to be provided to the enhancement layer coding/decoding process. Again, the scaling converts the value into 10-bit words. This up-scaling includes the following:

- The base image samples are scaled to 10-bit values before undergoing the spatial up-sampling process described above. Once the spatial up-sampling is done, the base image is restored in 8-bit representation, for the coding/decoding of subsequent base images.

  When the GRILP coding mode is active, the up-sampled base images used are stored in memory with 10-bit representation.
- The temporal residual data issued from the decoding of the base image are up-scaled before starting to use them in the computation of the Base Mode prediction image.
- Also, any other data such as the motion vectors are first scaled to 10-bit values before undergoing the spatial up-sampling process as described above.

Of course, any other word size may be used as long as the enhancement layer is processed using words of larger size than for the base layer.

The above examples are merely embodiments of the invention, which is not limited thereby.

CLAIMS

1. A method for encoding a video sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, comprising:

encoding a base layer made of base images;

encoding an enhancement layer made of enhancement images, including encoding at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame prediction only; and encoding at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame prediction;

wherein encoding the enhancement original INTRA image comprises the steps of:

obtaining a residual enhancement image as a difference between the enhancement original INTRA image and a decoded version of the corresponding encoded base image in the base layer, the residual enhancement image comprising a plurality of blocks of pixels, each block having a block type;

transforming pixel values for a block among said plurality of blocks into a set of coefficients each having a coefficient type, said block having a given block type;

determining an initial coefficient encoding merit for each coefficient type;

selecting coefficients based, for each coefficient, on the initial coefficient encoding merit for said coefficient type and on a predetermined block merit;

quantizing the selected coefficients into quantized symbols;

encoding the quantized symbols.

2. The encoding method according to Claim 1, wherein a coefficient type is selected if the initial encoding merit for this coefficient type is greater than the predetermined block merit.

3. The encoding method according to Claim 1 or 2, comprising a prior step of determining the predetermined block merit based on a predetermined frame merit and on a number of blocks of the given block type per area unit.

4. The encoding method according to Claim 3, wherein determining the predetermined frame merit comprises determining a frame merit and a distortion at the image level using a balancing parameter such that a video merit, computed based on said distortion and said frame merit, corresponds to a target video merit.

5. The encoding method according to Claim 4, wherein the step of determining a frame merit and a distortion at the image level is such that a product of the determined distortion at the image level and of the target video merit essentially equals a product of the balancing parameter and the determined frame merit.

6. The encoding method according to any of Claims 3 to 5, wherein the enhancement original INTRA image is a luminance image, the enhancement layer comprises at least one corresponding colour image comprising a plurality of colour blocks; and the method comprises steps of :

determining a colour frame merit;

determining, for each colour block of said plurality of colour blocks, a colour block merit for the concerned colour block based on the colour frame merit;

transforming, for each colour block of the plurality of blocks, pixel values for the concerned colour block into a set of coefficients each having a coefficient type;

selecting coefficient types based, for each coefficient, on an initial encoding merit for said coefficient type and on the colour block merit for the concerned colour block;

for each block of said plurality of colour blocks, selecting, for each selected coefficient type, a quantizer based on the colour block merit for the concerned colour block;

for each selected coefficient type, encoding coefficients having the concerned type using the selected quantizer for the concerned coefficient type.

7. The encoding method according to Claim 6, wherein determining the colour frame merit uses a balancing parameter.

8. The encoding method according to Claim 7, wherein determining the predetermined frame merit comprises determining a frame merit and a distortion at the image level such that a product of the determined distortion at the image level and of a target video merit essentially equals the determined frame merit and wherein the step of determining the colour frame merit is such that a product of a corresponding distortion for the colour frame and of the target video merit essentially equals a product of the balancing parameter and the determined colour frame merit.

9. The encoding method according to any of Claims 1 to 8, wherein determining an initial coefficient encoding merit for a given coefficient type includes estimating a ratio between a distortion variation provided by encoding a coefficient having the given type and a rate increase resulting from encoding said coefficient.

10. The encoding method according to any of Claims 1 to 9, wherein encoding the enhancement original INTRA image comprises the following steps:

determining, for each coefficient type and each block type, at least one parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type in the concerned block type;

determining the initial coefficient encoding merit for given coefficient type and block type based on the parameter for the given coefficient type and block type.

11. The encoding method of Claim 10, wherein encoding the enhancement original INTRA image comprises, for each coefficient for which the initial coefficient encoding merit is greater than the predetermined block merit, selecting a quantizer depending on the parameter for the concerned coefficient type and block type and on the predetermined block merit.

12. The encoding method of Claim 10 or 11, wherein a parameter obtained for a previous enhancement INTRA image and representative of a probabilistic distribution of coefficients having the concerned coefficient type in the concerned block type in the previous enhancement INTRA image is reused as the at least one parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type in the concerned block type in the enhancement original INTRA image being encoded.

13. The encoding method of any of Claims 10 to 12, wherein the coefficient types respectively associated with the encoded selected coefficients form a first group of coefficient types; and

the method further comprises:

transmitting the encoded selected coefficients and parameters associated with coefficient types of the first group;

transforming pixel values for at least one block in a second enhancement original INTRA image of the enhancement layer into a set of second-image coefficients each having a coefficient type;

encoding only a subset of the set of second-image coefficients for said block in the second enhancement original INTRA image, wherein the coefficient types respectively associated with the encoded second-image coefficients form a second group of coefficient types;

transmitting the encoded second-image coefficients and parameters associated with coefficient types of the second group not included in the first group.

14. The encoding method of Claim 13, wherein at least one parameter representative of the probabilistic distribution includes the standard deviation of the probabilistic distribution; and

the method further comprises the following steps:

for each coefficient type, computing a standard deviation for the probabilistic distribution of coefficients having the concerned coefficient type in the enhancement original INTRA image;

determining a number of bits necessary for representing the ratio between the maximum standard deviation, among the computed standard deviations associated with a coefficient type of the first group, and a predetermined value;

for each coefficient type of the first group, transmitting a word having a length equal to the determined number of bits and representing the standard deviation associated with the concerned coefficient type.

15. The encoding method of Claim 13 or 14, wherein the parameters associated with coefficient types of the first group are transmitted in a first transport unit and wherein the parameters associated with coefficient types of the second group not in the first group are transmitted in a second transport unit, distinct from the first transport unit.

16. The encoding method of Claim 15, wherein the encoded first-image coefficients are transmitted in the first transport unit and wherein the encoded second-image coefficients are transmitted in the second transport unit.

17. The encoding method of Claim 16, wherein the first and second transport units are parameter transport units.

18. The encoding method according to any of Claims 15 to 17, wherein the first transport unit carries a predetermined identifier and wherein the second transport unit carries said predetermined identifier.

19. The encoding method of any of Claims 13 to 18, comprising a step of estimating a proximity criterion between the enhancement original INTRA image being encoded and a third enhancement original INTRA image included in the enhancement layer,

the method further comprising the following steps if the proximity criterion is fulfilled:

transforming pixel values for at least one block in the third enhancement original INTRA image into a set of third-image coefficients each having a coefficient type;

encoding third-image coefficients for said block in the third enhancement original INTRA image, wherein the coefficient types respectively

associated with the encoded third-image coefficients form a third group of coefficient types;

transmitting the encoded third-image coefficients, parameters associated with coefficient types of the third group not included in the first and second groups and a flag indicating previously received parameters are valid.

20. The encoding method of Claim 19, further comprising the following steps if the proximity criterion is not fulfilled:

for each of a plurality of blocks in the third enhancement original INTRA image, transforming pixel values for the concerned block into a set of third-image coefficients each having a coefficient type;

for each coefficient type, computing at least one parameter representative of a probabilistic distribution of the third-image coefficients having said coefficient type;

for at least one block in the third enhancement original INTRA image, encoding third-image coefficients for said block;

transmitting the encoded third-image coefficients, parameters associated with coefficient types of transmitted third-image coefficients and a flag indicating previously received parameters are no longer valid.

21. The encoding method of Claim 19 or 20, wherein estimating the proximity criterion includes estimating a difference between a distortion relating to the first enhancement original INTRA image and a distortion relating to the third enhancement original INTRA image.

22. A method for encoding a video sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, comprising:

encoding a base layer made of base images;

encoding an enhancement layer made of enhancement images, including encoding at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame prediction only; and encoding at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame prediction;

wherein encoding the enhancement original INTRA image comprising the steps of:

obtaining a residual enhancement image as a difference between the enhancement original INTRA image and a decoded version of the corresponding encoded base image in the base layer, the residual enhancement image comprising a plurality of blocks of pixels, each block having a block type;

performing an initial segmentation of the residual enhancement image into a set of initial blocks, thus determining, for each initial block, a block type associated with the concerned initial block;

determining, for each block type, an associated set of quantizers based on data corresponding to pixels of blocks having said block type;

selecting, among a plurality of possible segmentations defining an association between each block of this segmentation and an associated block type, the segmentation which minimizes an encoding cost estimated based on a measure of the rate necessary for encoding each block using the set of quantizers associated with the block type of the encoded block according to the concerned segmentation.

23. The encoding method according to Claim 22, wherein the encoding cost is computed using a predetermined frame merit and a number of blocks per area unit for the concerned block type.

24. The encoding method according to Claim 22 or 23, wherein the measure of the rate is computed based on the set of quantizers associated with the concerned block type and on parameters representative of probabilistic distributions of transformed coefficients of blocks having the concerned block type.

25. The encoding method according to any of Claims 22 to 24, wherein the encoding cost includes a cost for luminance, taking into account luminance distortion generated by encoding and decoding a luminance block using the set of quantizers associated with the concerned block type, and a cost for chrominance, taking into account chrominance distortion generated by encoding and decoding a chrominance block using the set of quantizers associated with the concerned block type.

26. The encoding method according to any of Claims 22 to 25, wherein the initial segmentation into blocks is based on block activity along several spatial orientations.

27. The encoding method according to any of Claims 22 to 26, wherein the selected segmentation is represented as a quad-tree having a plurality of levels, each associated with a block size, and leaves associated with blocks and having a value indicating either a label for the concerned block or a subdivision of the concerned block.

28. The encoding method according to Claim 27, wherein encoding the enhancement original INTRA image comprising a step of compressing the quad tree using an arithmetic entropy coding that uses, when coding the segmentation relating to

a given block, conditional probabilities for the various possible leaf values depending on a state of a block in the base layer co-located with said given block.

29. A method for encoding a video sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, combining the method according to any of Claims 1 to 21 and the method according to any of Claims 22 to 28.

30. The encoding method according to any of Claims 1 to 29, comprising:

down-sampling video data having a first resolution to generate video data having a second resolution lower than said first resolution, and encoding the second resolution video data to obtain video data of the base layer having said second resolution;

decoding the base layer video data, up-sampling the decoded base layer video data to generate decoded video data having said first resolution, forming a difference between the generated decoded video data having said first resolution and said received video data having said first resolution to generate residual data,

compressing the residual data to generate video data of the enhancement layer, including determining an image segmentation into blocks for the enhancement layer, wherein the segmentation is represented as a quad-tree having a plurality of levels, each associated with a block size, and leaves associated with blocks and having a value indicating either a label for the concerned block or a subdivision of the concerned block;

arithmetic entropy coding the quad-tree using, when coding the segmentation relating to a given block, conditional probabilities for the various possible leaf values depending on a state of a block in the base layer co-located with said given block.

31. The encoding method according to any of Claims 1 to 30, comprising:

decoding the encoded coefficients of the enhancement original INTRA image and decoding the corresponding encoded base image in the base layer, to obtain a rough decoded image corresponding to an original image of the sequence;

processing the rough decoded image through at least one adaptive post-filter adjustable depending on a parameter, wherein said parameter is derived based on pixel values and input to the adaptive post-filter.

32. The encoding method according to Claim 31 when depending on Claim 3 or 23, comprising determining the parameter input to the adaptive post-filter based on the predetermined frame merit.

33. A method for encoding a sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, comprising:

encoding a base layer made of base images;

encoding an enhancement layer made of enhancement images, including encoding at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame prediction only; and encoding at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame prediction.

wherein encoding the enhancement original INTER image comprises the steps of:

selecting a prediction mode, from among a plurality of prediction modes, for predicting an enhancement block of the enhancement original INTER image, wherein the plurality of prediction modes includes at least one of:

a base mode prediction mode involving computation, from the base layer, of a base mode prediction image corresponding to the enhancement original INTER image, the base mode prediction image being composed of base mode blocks obtained using prediction information derived from prediction information of the base layer and wherein the enhancement block is derived from one or more base mode blocks of a spatially corresponding region of the base mode prediction image; and

a GRILP prediction mode including: obtaining a block predictor candidate for predicting the enhancement block within the enhancement original INTER image and an associated enhancement-layer residual block corresponding to said prediction; determining a block predictor in the base layer co-located with the determined block predictor candidate within the enhancement original INTER image; determining a base-layer residual block associated with the enhancement block in the base layer that is co-located with the enhancement block in the enhancement original INTER image, as the difference between the co-located enhancement block in the enhancement original INTER image and the determined block predictor in the base layer; determining, for the enhancement block of the enhancement original INTER image, a further residual block corresponding, at least partly, to the difference between the enhancement-layer residual block and the base-layer residual block;

obtaining a prediction block from the selected prediction mode and subtracting the prediction block from the enhancement block of the enhancement original INTER image to obtain a residual block;

transforming pixels values of the residual block to obtain transformed coefficients;

quantizing at least one of the transformed coefficients to obtain quantized symbols;

encoding the quantized symbols into encoded data.

34. The encoding method according to Claim 33, wherein the plurality of prediction modes includes an inter difference mode in addition or in replacement of the GRILP mode including: performing a motion estimation on a current block of a current Enhancement Layer (EL) image to obtain a motion vector designating a reference block in a EL reference image; computing a difference image between the EL reference image and an upsampled version of the image of the base layer temporally co-located with the EL reference image; performing a motion compensation on the block of the obtained difference image pointed by the motion vector determined during the motion estimation step to obtain a residual block; adding the obtained residual block to the reference block to obtain a final block predictor used to predict the current block.

35. The encoding method according to Claim 33 or 34, wherein the plurality of prediction modes includes the following prediction modes:

a motion compensated temporal prediction mode within the enhancement layer;

an intra base layer prediction mode where the prediction block is taken from a base block co-located with the enhancement block in an up-sampled decoded version of the corresponding base image;

the base mode prediction mode, wherein each base mode block of the base mode prediction image derives from the co-located base block in the corresponding base image when the co-located base block is intra coded or derives, when the co-located base block is inter coded into a base residual using prediction information, from the block in the enhancement layer that is obtained by applying an up-sampled version of a motion vector of the prediction information onto the base mode block and from an up-sampled decoded version of the base residual;

the GRILP prediction mode and/or the Inter Difference prediction mode; and

a difference INTRA coding mode.

36. The encoding method according to Claim 33, 34 or 5, wherein, in the GRILP prediction mode, when an image of residual data used for the encoding of the base layer is available,

determining the base-layer residual block in the base layer comprises:

determining the overlap in the image of residual data between the determined block predictor and the block predictor used in the encoding of the block co-located with the enhancement block in the base layer; and

using the part in the image of residual data corresponding to this overlap if any to compute a part of said further residual block of the enhancement original INTER image, wherein the samples of said further residual block of the enhancement original INTER image corresponding to this overlap each corresponds to a difference between a sample of the enhancement-layer residual block and a corresponding sample of the base-layer residual block.

37. The encoding method according to any of Claims 33 to 6, wherein, in the GRILP prediction mode, the determination of a predictor of the enhancement block is made using a cost function adapted to take into account the prediction of the enhancement-layer residual block to determine a rate distortion cost.

38. The encoding method according to any of Claims 33 to 37, comprising de-blocking filtering the base mode prediction image before it is used to provide prediction blocks.

39. The encoding method according to Claim 8, wherein the de-blocking filtering is applied to the boundaries of the base mode blocks of the base mode prediction image.

40. The encoding method according to Claim 38 or 9, further comprising deriving the organisation of transform units of base blocks in the base layer towards the enhancement layer wherein the de-blocking filtering is applied to the boundaries of the transform units derived from the base layer.

41. The encoding method according to any of Claims 33 to40, wherein the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained; and

encoding the enhancement original INTER image further comprises encoding the motion information using a motion information predictor taken from a set

including motion information, if any, associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

42. The encoding method according to Claim 1, wherein other motion information of the set is derived from the motion information by adding respective spatial offsets.

43. The encoding method according to Claim 41 or 2, wherein the set includes motion information, if any, associated with blocks spatially neighbouring the enhancement block in the enhancement original INTER image, and if no motion information exist for a given neighbouring block, motion information, if any, associated with the base block spatially corresponding to the given neighbouring block in the corresponding base image.

44. The encoding method according to any of Claims 33 to 3, wherein the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained;

encoding the enhancement original INTER image further comprises encoding the motion information using a motion information predictor taken from a set including more motion information predictors than another set usable for predicting motion information associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

45. The encoding method according to Claim 4, wherein the set of motion information predictors for the enhancement original INTER image includes at least one motion information predictor generated based on motion information from the base layer.

46. The encoding method according to any of Claims 33 to 5, wherein the set of vector predictors comprises at least one temporal motion information predictor and at least one spatial motion information predictor, the at least one temporal motion information predictor being positioned before the at least one spatial motion information predictor.

47. The encoding method according to any of Claims 33 to 6, wherein in case of spatial scalability between the base layer and the enhancement layer with the base layer having a lower spatial resolution than the enhancement layer; and

prediction information, such as a motion vector, is derived or up-sampled for a processing block of size 2Nx2N in the enhancement original INTER image from the base layer of lower spatial resolution, the derivation or up-sampling comprises:

determining whether or not the region of the base layer, spatially corresponding to the processing block, is wholly located within one elementary prediction unit of the base layer; and

in the case where the region of the base layer spatially corresponding to the processing block is fully located within one elementary prediction unit of the base layer, deriving prediction information for that processing block from the base layer prediction information of the said one elementary prediction unit;

otherwise in the case where the region of the base layer spatially corresponding to the processing block overlaps, at least partially, each of a plurality of elementary prediction units,

dividing the processing block into a plurality of sub-processing blocks , each of size NxN such that the region of the base layer spatially corresponding to each sub-processing block is wholly located within one elementary prediction unit of the base layer; and

deriving the prediction information for each sub-processing block from the base layer prediction information of the spatially corresponding elementary prediction unit.

48. The encoding method according to Claim 7, wherein in the case where the corresponding elementary prediction unit of the base layer is Intra-coded then the processing block is predicted from the elementary prediction unit reconstructed and resampled to the enhancement layer resolution.

49. The encoding method according to Claim 47, wherein in the case where the corresponding elementary prediction unit is Inter-coded then the processing block is temporally predicted using motion information derived from the said corresponding elementary prediction unit of the base layer.

50. The encoding method according to any of Claims 7 to 9, wherein the processing block is temporally predicted further using a decoded temporal residual of the corresponding elementary prediction unit of the base layer, said temporal residual being computed between base layer images, as a function of the motion information of the elementary prediction unit.

51. The encoding method according to the preceding claim, wherein the spatial scaling between an image of the enhancement layer and a corresponding image of the base layer is a non-integer ratio

52. The encoding method according to the preceding claim, wherein the non-integer ratio is 1.5.

53. The encoding method according to any of Claims 33 to50, wherein the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the method comprises:

obtaining a first set of quantization offsets to be applied to a group of enhancement images, a different quantization offset being obtained for at least two enhancement INTER images belonging to the same temporal depth; and

determining a quantization parameter used to encode the enhancement INTER images based on the obtained first set of quantization parameters.

54. The encoding method according to any of Claims 33 to50, wherein the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the method comprises:

obtaining a first set of quantization offsets to be applied to a group of enhancement images, the same quantization offset being obtained for at least two enhancement INTER images belonging to different temporal depths; and

determining a quantization parameter used to encode the enhancement INTER images based on the obtained first set of quantization parameters.

55. The encoding method according to any of Claims 53 to 54, wherein among the at least two enhancement INTER images that belong to the same temporal depth, a first offset is obtained for a first enhancement INTER image having a reference image of a first quality and a second offset, larger than said first offset, is obtained for a second enhancement INTER image having a reference image of a second quality lower than said first quality.

56. The encoding method according to any of Claims 53 to 55, wherein the quantization offset obtained for an enhancement INTER image takes into account:

the number of enhancement images using the enhancement INTER image as a reference image for temporal prediction;

the temporal distance of the enhancement INTER image to its reference image having the lowest quantization offset; and

the value of the quantization parameter applied to this reference image.

57. The encoding method according to any of Claims 53 to 56, wherein the quantization offset obtained for an enhancement INTER image is equal or larger than the quantization offset of its reference image having the lowest quantization offset.

58. The encoding method according to any of Claims 53 to 57, further comprising encoding, using a second set of different quantization offsets, a group of base images in the base layer that temporally coincides with the group of enhancement images ;

wherein the second set is obtained based on a temporal depth each base image belongs to.

59. The encoding method according to any of Claims 35 to 58, wherein encoding data representing the enhancement original INTER image further comprises encoding, in the bit-stream, quad-trees representing a segmentation of respective portions of the enhancement original INTER image and coding modes associated with blocks in the segmentation;

wherein the coding mode associated with a given block is encoded through

a first coding mode syntax element that indicates whether the coding mode associated with the given block is based on temporal/Inter prediction or not,

a second coding mode syntax element that indicates whether a prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information is used or not for encoding the block if the first coding mode syntax element refers to temporal/Inter prediction, or indicates whether the coding mode associated with the given block is a conventional Intra prediction or based on Inter-layer prediction if the first coding mode syntax element refers to non temporal/Inter prediction, and

in the case in which the coding mode associated with the given block is based on Inter-layer prediction , a third coding mode syntax element that indicates whether the coding mode associated with the given block is the intra base layer mode or the base mode prediction mode.

60. The encoding method according to claim 59 wherein the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the GRILP prediction sub-mode.

61. The encoding method according to claim 59 wherein the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the inter difference prediction sub-mode.

62. The encoding method according to claim 60 or 61 wherein, a fourth coding mode syntax element indicates whether the inter difference block is used or not for encoding the block if the second coding mode syntax element indicates that the GRILP mode is not used or whether the GRILP mode is used or not for encoding the block if the second coding mode syntax element indicates that the inter difference mode is not used.

63. The encoding method according to any claim 59 to 62 wherein, at least one high level syntax element indicates which of the coding mode syntax elements are present or not in the bit-stream.

64. The encoding method according to claim 63 wherein, when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding mode syntax element following the removed coding mode syntax element replaces the removed coding mode syntax element and the coding order of the remaining coding mode syntax elements is kept.

65. The encoding method according to claim 63 wherein, when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding order of the remaining coding mode syntax elements is modified.

66. The encoding method according to claim 65 wherein the modification of the coding order of the remaining coding mode syntax elements takes into account the probability of occurrence of coding modes represented by remaining coding mode syntax elements.

67. A method for encoding a video sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, combining the method according to any of Claims 1 to 32 and the method according to any of Claims 33 to 66.

68. The encoding method according to Claim 67, wherein

encoding the enhancement original INTRA image comprises selecting quantizers from the predetermined block merit to quantize the selected coefficients, the predetermined block merit deriving from a frame merit;

encoding the enhancement original INTER image comprises selecting quantizers from a quantization parameter to quantize the transformed coefficients; and

the frame merit and the quantization parameter are computed from a user-specified quality parameter and are linked together with a balancing parameter.

69. The encoding method according to any of the preceding claims, implemented by a computer, wherein data, such as image samples, in the base layer are processed using 8-bit words and data, such as image samples, in the enhancement layer are processed using 10-bit words.

70. The encoding method according to Claim 69, wherein data, such as image samples, from the base layer are up-scaled to 10-bit words by multiplying each value by 4, when this data from the base layer is processed for the encoding of the enhancement layer.

71. A method for decoding a scalable video bit-stream, comprising:

decoding a base layer made of base images;

decoding an enhancement layer made of enhancement images, including decoding data representing at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame decoding; and decoding data representing at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame decoding;

wherein decoding data representing at least one block of pixels in the enhancement original INTRA image, comprises the steps of:

receiving said data and parameters each representative of a probabilistic distribution of a coefficient type;

decoding said data into symbols;

selecting coefficient types for which a coefficient encoding merit prior to encoding, estimated based on the parameter associated with the concerned coefficient type, is greater than a predetermined block merit;

for selected coefficient types, dequantizing symbols into dequantized coefficients having a coefficient type among the selected coefficient types;

transforming dequantized coefficients into pixel values in the spatial domain for said block.

72. The decoding method according to Claim71, comprising a prior step of determining the predetermined block merit based on a predetermined frame merit and on a number of blocks of a block type of the block per area unit.

73. The decoding method according to Claim 72, wherein determining the predetermined frame merit comprises determining a frame merit and a distortion at the

image level using a balancing parameter such that a video merit, computed based on said distortion and said frame merit, corresponds to a target video merit.

74. The decoding method according to Claim 73, wherein the step of determining a frame merit and a distortion at the image level is such that a product of the determined distortion at the image level and of the target video merit essentially equals a product of the balancing parameter and the determined frame merit.

75. The decoding method according to any of Claims 71 to 74, wherein the predetermined frame merit is decoded from the bit-stream.

76. The decoding method according to any of Claims 71 to 75, wherein the enhancement original INTRA image is a luminance image, the enhancement layer comprises at least one corresponding colour image comprising a plurality of colour blocks; and the method comprises steps of :

determining a colour frame merit;

decoding data associated with a colour block among said plurality of colour blocks into a set of symbols each corresponding to a coefficient type, said block having a particular block type;

determining a colour block merit based on the colour frame merit and on a number of blocks of the particular block type per area unit;

selecting coefficient types based, for each coefficient type, on a coefficient encoding merit prior to encoding, for said coefficient type, and on the colour block merit;

for selected coefficient types, dequantizing symbols into dequantized coefficients having a coefficient type among the selected coefficient types;

transforming dequantized coefficients into pixel values in the spatial domain for said colour block.

77. The decoding method according to Claim 76, wherein determining the colour frame merit uses a balancing parameter.

78. The decoding method according to Claim 77, wherein determining the predetermined frame merit comprises determining a frame merit and a distortion at the image level such that a product of the determined distortion at the image level and of a target video merit essentially equals the determined frame merit and wherein the step of determining the colour frame merit is such that a product of a corresponding distortion for the colour frame and of the target video merit essentially equals a product of the balancing parameter and the determined colour frame merit.

79. The decoding method according to any of Claims 71 to 78, wherein the coefficient encoding merit prior to encoding for a given coefficient type estimates a ratio between a distortion variation provided by encoding a coefficient having the given type and a rate increase resulting from encoding said coefficient.

80. The decoding method according to any of Claims 71 to 79, wherein decoding data representing at least one block in the enhancement original INTRA image comprises, for each coefficient for which the coefficient encoding merit prior to encoding is greater than the predetermined block merit, selecting a quantizer depending on the received parameter associated with the concerned coefficient type and on the predetermined block merit, wherein dequantizing symbols is performed using the selected quantizer.

81. The decoding method according to Claim 80, wherein decoding data representing the enhancement original INTRA image comprises determining the coefficient encoding merit prior to encoding for given coefficient type and block type based on the received parameters for the given coefficient type and block type.

82. The decoding method of any of Claims 80 to 81, wherein a parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type previously received for a previous enhancement INTRA image is reused as the at least one parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type in the enhancement original INTRA image being decoded.

83. The decoding method according to any of Claims 80 to 82, wherein the selected coefficient types of the enhancement original INTRA image being decoded belong to a first group; and

the method further comprises the following steps:

receiving encoded coefficients relating to a second enhancement original INTRA image of the enhancement layer and having coefficient types in a second group;

receiving parameters associated with coefficient types of the second group not included in the first group;

decoding the received coefficients relating to the second enhancement original INTRA image, wherein decoding a received coefficient having a given coefficient type in the first group includes a step of dequantizing using a dequantizer selected based on the previously received parameter associated with the given coefficient type;

transforming the decoded coefficients into pixel values for the second enhancement original INTRA image.

84. The decoding method according to Claim 83, wherein the parameters associated with coefficient types of the first group are received in a first transport unit and wherein the parameters associated with coefficient types of the second group not in the first group are received in a second transport unit, distinct from the first transport unit.

85. The decoding method according to the preceding claim, wherein the information supplied to the decoder for said second image does not include information about the reused parameter(s).

86. The decoding method according to the preceding claim, wherein such a parametric probabilistic model is obtained for each type of encoded DCT coefficient in said first image.

87. The decoding method according to the preceding claim, wherein parameters of the first-image parametric probabilistic model obtained for at least one said DCT coefficient type are reused for said second image.

88. The decoding method according to Claim 83 or 84, comprising a step of receiving encoded coefficients relating to a third enhancement original INTRA image of the enhancement layer and a flag indicating whether previously received parameters are valid,

the method comprising the following steps if the received flag indicate that the previously received parameters are valid:

receiving parameters associated with coefficient types of a third group not included in the first and second groups;

decoding the received coefficients relating to the third enhancement original INTRA image, wherein decoding a received coefficient having a given coefficient type in the first or second group includes a step of dequantizing using a dequantizer selected based on the previously received parameter associated with the given coefficient type;

transforming the decoded coefficients into pixel values for the third enhancement original INTRA image;

89. The decoding method according to Claim 88, comprising the following steps if the received flag indicate that the previously received parameters are no longer valid:

receiving encoded coefficients relating to the third enhancement original INTRA image and having coefficient types in a first group;

receiving new parameters associated with coefficient types of encoded coefficients relating to the third enhancement original INTRA image;

decoding the received coefficients relating to the third enhancement original INTRA image, wherein decoding a received coefficient having a given coefficient type includes a step of dequantizing using a dequantizer selected based on the received new parameter associated with the given coefficient type;

transforming the decoded coefficients into pixel values for the third enhancement original INTRA image.

90. The decoding method according to any of Claims 71 to 89, further comprising decoding, from the bit-stream, a quad-tree representing a segmentation of the enhancement original INTRA image said plurality of blocks of pixels, each block having a block type, the quad tree having a plurality of levels, each associated with a block size, and leaves associated with blocks and having a value indicating either a label for the concerned block or a subdivision of the concerned block.

91. The decoding method according to the preceding claim, wherein decoding the quad tree uses an arithmetic entropy decoding that uses, when decoding the segmentation relating to a given block, conditional probabilities for the various possible leaf values depending on a state of a block in the base layer co-located with said given block.

92. The decoding method according to Claim 71 to 91, comprising:

receiving video data of the base layer, video data of the enhancement layer, a table of conditional probabilities and a coded quad-tree representing, by leaf values, an image segmentation into blocks for the enhancement original INTRA image;

decoding video data of the base layer to generate decoded base layer video data having a second resolution, lower than a first resolution, and up-sampling the decoded base layer video data to generate up-sampled video data having the first resolution;

for at least one block represented in the quad-tree, determining the probabilities respectively associated with the possible leaf values based on the received table and depending on a state of a block in the base layer co-located with said block;

decoding the coded quad-tree to obtain the segmentation, including arithmetic entropy decoding the leaf value associated with said block using the determined probabilities;

decoding, using the obtained segmentation, video data of the enhancement layer to generate residual data having the first resolution;

forming a sum of the up-sampled video data and the residual data to generate enhanced video data.

93. The decoding method according to Claim 71 to 92, comprising determining the parameter input to the adaptive post-filter based on the predetermined frame merit.

94. A method for decoding a scalable video bit-stream, comprising:

decoding a base layer made of base images;

decoding an enhancement layer made of enhancement images, including decoding data representing at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame decoding; and decoding data representing at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame decoding;

wherein decoding data representing the enhancement original INTER image comprising a plurality of blocks of pixels, each block having a block type, comprises the steps of:

decoding prediction mode information from the bit-stream for at least one enhancement block of the enhancement original INTER image to obtain a prediction mode having been selected from among a plurality of prediction modes, wherein the plurality of prediction modes includes at least one of:

a base mode prediction mode involving computation, from the base layer, of a base mode prediction image corresponding to the enhancement original INTER image, the base mode prediction image being composed of base mode blocks obtained using prediction information derived from prediction information of the base layer and wherein the enhancement block is derived from one or more base mode blocks of a spatially corresponding region of the base mode prediction image; and

a GRILP prediction mode including: obtaining from the bit-stream the location of a block predictor of the enhancement block within the enhancement original INTER image to be decoded and a residual block comprising difference information between enhancement image residual information and base layer residual information; determining a block predictor in

the base layer co-located with the block predictor in the enhancement original INTER image; determining a base-layer residual block corresponding to the difference between the block of the base layer co-located with the enhancement block to be decoded and the determined block predictor in the base layer; reconstructing an enhancement-layer residual block using the determined base-layer residual block and said residual block obtained from the bit stream; reconstructing the enhancement block using the block predictor and the enhancement-layer residual block;

obtaining a prediction block from the selected prediction mode and adding the prediction block to a decoded enhancement residual block to obtain the enhancement block, said enhancement residual block comprising quantized symbols, the decoding of the enhancement original INTER image comprising inverse quantizing these quantized symbols to obtain transformed coefficients.

95. The decoding method according to Claim 94, wherein the plurality of prediction modes includes an inter difference mode in addition or in replacement of the GRILP mode including for a block to decode: obtaining a motion vector designating a reference block in a EL reference image; computing a difference image between the EL reference image and an upsampled version of the image of the base layer temporally co-located with the EL reference image; performing a motion compensation on the block of the obtained difference image pointed by the motion vector determined during the motion estimation step to obtain a residual block; adding the obtained residual block to the reference block to obtain a final block predictor used to decode the current block.

96. The decoding method of Claim 94 or 95 wherein the plurality of prediction modes includes the following prediction modes:

a motion compensated temporal prediction mode within the enhancement layer;

an intra base layer prediction mode where the prediction block is taken from a base block co-located with the enhancement block in an up-sampled decoded version of the corresponding base image;

the base mode prediction mode, wherein each base mode block of the base mode prediction image derives from the co-located base block in the corresponding base image when the co-located base block is intra coded or derives, when the co-located base block is inter coded into a base residual using prediction information, from the block in the enhancement layer that is obtained by applying an

up-sampled version of a motion vector of the prediction information onto the base mode block and from an up-sampled decoded version of the base residual;

the GRILP prediction mode and/or the inter difference prediction mode; and

a difference INTRA coding mode.

97. The decoding method according to Claim 94 or 96, wherein, in the GRILP prediction mode, when an image of residual data of the base layer is available,

determining the base-layer block residual in the base layer comprises:

determining the overlap in the image of residual data between the obtained block predictor and the block predictor used in the encoding of the block co-located with the enhancement block in the base layer;

using the part in the image of residual data corresponding to this overlap if any to reconstruct a part of the enhancement-layer residual block, wherein the samples of the enhancement-layer residual block corresponding to this overlap each involves an addition of a sample of the obtained residual block and a corresponding sample of the base-layer residual block.

98. The decoding method according to any of Claims 94 to 97, comprising de-blocking filtering the base mode prediction image before it is used to provide prediction blocks.

99. The decoding method according to Claim 98, wherein the de-blocking filtering is applied to the boundaries of the base mode blocks of the base mode prediction image.

100. The decoding method according to Claim 98 or 99, further comprising deriving the organisation of transform units of base blocks in the base layer towards the enhancement layer wherein the de-blocking filtering is applied to the boundaries of the transform units derived from the base layer.

101. The decoding method according to any of Claims 94 to 100, wherein the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained; and

decoding the enhancement original INTER image further comprises decoding the motion information using a motion information predictor taken from a set including motion information, if any, associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

102. The decoding method according to Claim 101, wherein other motion information of the set is derived from the motion information by adding respective spatial offsets.

103. The decoding method according to Claim 101 or 102, wherein the set includes motion information, if any, associated with blocks spatially neighbouring the enhancement block in the enhancement original INTER image, and if no motion information exist for a given neighbouring block, motion information, if any, associated with the base block spatially corresponding to the given neighbouring block in the corresponding base image.

104. The decoding method according to any of Claims 94 to 103, wherein the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained;

decoding the enhancement original INTER image further comprises decoding the motion information using a motion information predictor taken from a set including more motion information predictors than another set usable for predicting motion information associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

105. The encoding method according to Claim 104, wherein the set of motion information predictors for the enhancement original INTER image includes at least one motion information predictor generated based on motion information from the base layer.

106. The decoding method according to any of Claims 94 to 105, wherein the set of vector predictors comprises at least one temporal motion information predictor and at least one spatial motion information predictor, the at least one temporal motion information predictor being positioned before the at least one spatial motion information predictor.

107. The decoding method according to any of Claims 94 to 106, wherein in case of spatial scalability between the base layer and the enhancement layer with the base layer having a lower spatial resolution than the enhancement layer; and

prediction information, such as a motion vector, is derived or up-sampled for a processing block of size 2Nx2N in the enhancement original INTER image from the base layer of lower spatial resolution, the derivation or up-sampling comprises:

determining whether or not the region of the base layer, spatially corresponding to the processing block, is wholly located within one elementary prediction unit of the base layer; and

in the case where the region of the base layer spatially corresponding to the processing block is fully located within one elementary prediction unit of the base layer, deriving prediction information for that processing block from the base layer prediction information of the said one elementary prediction unit;

otherwise in the case where the region of the base layer spatially corresponding to the processing block overlaps, at least partially, each of a plurality of elementary prediction units,

dividing the processing block into a plurality of sub-processing blocks , each of size NxN such that the region of the base layer spatially corresponding to each sub-processing block is wholly located within one elementary prediction unit of the base layer; and

deriving the prediction information for each sub-processing block from the base layer prediction information of the spatially corresponding elementary prediction unit.

108. The decoding method according to Claim 107, wherein in the case where the corresponding elementary prediction unit of the base layer is Intra-coded then the processing block is predicted from the elementary prediction unit reconstructed and resampled to the enhancement layer resolution.

109. The decoding method according to Claim 107, wherein in the case where the corresponding elementary prediction unit is Inter-coded then the processing block is temporally predicted using motion information derived from the said corresponding elementary prediction unit of the base layer.

110. The decoding method according to any of Claims 107 to 109, wherein the processing block is temporally predicted further using a decoded temporal residual of the corresponding elementary prediction unit of the base layer, said temporal residual being computed between base layer images, as a function of the motion information of the elementary prediction unit.

111. The encoding method according to the preceding claim, wherein the spatial scaling between an image of the enhancement layer and a corresponding image of the base layer is a non-integer ratio

112. The encoding method according to the preceding claim, wherein the non-integer ratio is 1.5.

113. The decoding method according to any of Claims 94 to 110, wherein the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the method comprises:

obtaining a first set of quantization offsets to be applied to a group of enhancement images, a different quantization offset being obtained for at least two enhancement INTER images belonging to the same temporal depth; and

determining a quantization parameter used for inverse quantization to decode the enhancement INTER images based on the obtained first set of quantization parameters.

114. The decoding method according to any of Claims 94 to 113, wherein the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the method comprises:

obtaining a first set of quantization offsets to be applied to a group of enhancement images, the same quantization offset being obtained for at least two enhancement INTER images belonging to different temporal depths; and

determining a quantization parameter used for inverse quantization to decode the enhancement INTER images based on the obtained first set of quantization parameters.

115. The encoding method according to Claims 113 and 114.

116. The encoding method according to any of Claims 113 to 115, wherein among the at least two enhancement INTER images that belong to the same temporal depth, a first offset is obtained for a first enhancement INTER image having a reference image of a first quality and a second offset, larger than said first offset, is obtained for a second enhancement INTER image having a reference image of a second quality lower than said first quality.

117. The encoding method according to any of Claims 113 to 116, wherein the quantization offset obtained for an enhancement INTER image takes into account:

the number of enhancement images using the enhancement INTER image as a reference image for temporal prediction;

the temporal distance of the enhancement INTER image to its reference image having the lowest quantization offset; and

the value of the quantization parameter applied to this reference image.

118. The encoding method according to any of Claims 113 to 117, wherein the quantization offset obtained for an enhancement INTER image is equal or larger than the quantization offset of its reference image having the lowest quantization offset.

119. The encoding method according to any of Claims 113 to 118, further comprising encoding, using a second set of different quantization offsets, a group of base images in the base layer that temporally coincides with the group of enhancement images ;

wherein the second set is obtained based on a temporal depth each base image belongs to.

120. The decoding method according to any of Claims 94 to 119, wherein decoding data representing the enhancement original INTER image further comprises decoding, from the bit-stream, quad-trees representing a segmentation of respective portions of the enhancement original INTER image and coding modes associated with blocks in the segmentation;

wherein decoding the quad-tree comprises decoding, from a received code associated with a block in the segmentation,

a first coding mode syntax element that indicates whether the coding mode associated with the block is based on temporal/Inter prediction or not,

a second coding mode syntax element that indicates whether a prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information is used or not for encoding the block is activated or not if the first coding mode syntax element refers to temporal/Inter prediction, or indicates whether the coding mode associated with the block is a conventional Intra prediction or based on Inter-layer prediction if the first coding mode syntax element refers to non temporal/Inter prediction, and

in the case in which the coding mode associated with the given block is based on Inter-layer prediction, a third coding mode syntax element that indicates whether the coding mode associated with the given block is the intra base layer mode or the base mode prediction mode.

121. The decoding method according to claim 120 wherein the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the GRILP prediction sub-mode.

122. The decoding method according to claim 120 wherein the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the inter difference prediction sub-mode.

123. A decoding method according to claim 121 or 122 wherein, a fourth coding mode syntax element indicates whether the inter difference block was used or not for encoding the block if the second coding mode syntax element indicates that the GRILP mode was not used or whether the GRILP mode was used or not for encoding the block if the second coding mode syntax element indicates that the inter difference mode was not used.

124. A decoding method according to any claim 120 to 123 wherein, at least one high level syntax element indicates which of the coding mode syntax elements are present or not in the bit-stream.

125. A decoding method according to claim 124 wherein, when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding mode syntax element following the removed coding mode syntax element replaces the removed coding mode syntax element and the coding order of the remaining coding mode syntax elements is kept.

126. A decoding method according to claim 125 wherein, when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding order of the remaining coding mode syntax elements is modified.

127. A decoding method according to claim 126 wherein the modification of the coding order of the remaining coding mode syntax elements takes into account the probability of occurrence of coding modes represented by remaining coding mode syntax elements.

128. A method for decoding a scalable video bit-stream made, combining the method according to any of Claims 71 to 93 and the method according to any of Claims 94 to 120.

129. The decoding method according to Claim 128, wherein

decoding the enhancement original INTRA image comprises selecting quantizers from the predetermined block merit to dequantize symbols of the selected coefficient types, the predetermined block merit deriving from a frame merit;

decoding the enhancement original INTER image comprises selecting quantizers from a quantization parameter to inverse quantize the quantized symbols; and

the frame merit and the quantization parameter are computed from a received quality parameter and are linked together with a balancing parameter.

130. The decoding method according to any of Claims 71 to 129, implemented by a computer, wherein data, such as image samples, in the base layer are processed using 8-bit words and data, such as image samples, in the enhancement layer are processed using 10-bit words.

131. The decoding method according to Claim 130, wherein data, such as image samples, from the base layer are up-scaled to 10-bit words by multiplying each value by 4, when this date from the base layer is processed for the decoding of the enhancement layer.

132. A video encoder for encoding a video sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, comprising:

a base layer encoding module for encoding a base layer made of base images;

an enhancement layer encoding module for encoding an enhancement layer made of enhancement images, including an Intra encoding module for encoding at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame prediction only; and an Inter encoding module for encoding at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame prediction;

wherein the Intra encoding module comprises:

a module for obtaining a residual enhancement image as a difference between the enhancement original INTRA image and a decoded version of the corresponding encoded base image in the base layer, the residual enhancement image comprising a plurality of blocks of pixels, each block having a block type;

a transforming module for transforming pixel values for a block among said plurality of blocks into a set of coefficients each having a coefficient type, said block having a given block type;

a merit determining module for determining an initial coefficient encoding merit for each coefficient type;

a coefficient selector for selecting coefficients based, for each coefficient, on the initial coefficient encoding merit for said coefficient type and on a predetermined block merit;

a quantizing module for quantizing the selected coefficients into quantized symbols;

an encoding module for encoding the quantized symbols.

133. The video encoder according to Claim 132, wherein a coefficient type is selected if the initial encoding merit for this coefficient type is greater than the predetermined block merit.

134. The video encoder according to Claim 132 or 133, comprising a block merit determining module for determining the predetermined block merit based on a predetermined frame merit and on a number of blocks of the given block type per area unit.

135. The video encoder according to Claim 134, comprising a block merit determining module for determining the predetermined frame merit by determining a frame merit and a distortion at the image level using a balancing parameter such that a video merit, computed based on said distortion and said frame merit, corresponds to a target video merit.

136. The video encoder according to Claim 135, wherein determining a frame merit and a distortion at the image level is such that a product of the determined distortion at the image level and of the target video merit essentially equals a product of the balancing parameter and the determined frame merit.

137. The video encoder according to any of Claims 134 to 136, wherein the enhancement original INTRA image is a luminance image, the enhancement layer comprises at least one corresponding colour image comprising a plurality of colour blocks; and the Intra encoding module comprises:

a frame merit determining module for determining a colour frame merit;

a block merit determining module for determining, for each colour block of said plurality of colour blocks, a colour block merit for the concerned colour block based on the colour frame merit;

a transforming module for transforming, for each colour block of the plurality of blocks, pixel values for the concerned colour block into a set of coefficients each having a coefficient type;

a coefficient type selector for selecting coefficient types based, for each coefficient, on an initial encoding merit for said coefficient type and on the colour block merit for the concerned colour block;

a quantizer selector for selecting, for each block of said plurality of colour blocks, and for each selected coefficient type, a quantizer based on the colour block merit for the concerned colour block;

an encoding module for encoding, for each selected coefficient type, coefficients having the concerned type using the selected quantizer for the concerned coefficient type.

138. The video encoder according to Claim 137, wherein the frame merit determining module for determining the colour frame merit uses a balancing parameter.

139. The video encoder according to Claim 138, wherein frame merit determining module is configured to determine a frame merit and a distortion at the image level such that a product of the determined distortion at the image level and of a target video merit essentially equals the determined frame merit and to determine the colour frame merit such that a product of a corresponding distortion for the colour frame and of the target video merit essentially equals a product of the balancing parameter and the determined colour frame merit.

140. The video encoder according to any of Claims 132 to 139, wherein the merit determining module for determining an initial coefficient encoding merit for a given coefficient type includes an estimator for estimating a ratio between a distortion variation provided by encoding a coefficient having the given type and a rate increase resulting from encoding said coefficient.

141. The video encoder according to any of Claims 132 to 140, wherein the Intra encoding module comprises:

a parameter determining module for determining, for each coefficient type and each block type, at least one parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type in the concerned block type;

and the merit determining module for determining the initial coefficient encoding merit for given coefficient type and block type is configured to determine said initial coefficient encoding merit based on the parameter for the given coefficient type and block type.

142. The video encoder of Claim 141, wherein the Intra encoding module comprises the quantizer selector for selecting, for each coefficient for which the initial coefficient encoding merit is greater than the predetermined block merit, a quantizer depending on the parameter for the concerned coefficient type and block type and on the predetermined block merit.

143. The video encoder of Claim 141 or 142, wherein a parameter obtained for a previous enhancement INTRA image and representative of a probabilistic distribution of coefficients having the concerned coefficient type in the concerned block type in the

previous enhancement INTRA image is reused as the at least one parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type in the concerned block type in the enhancement original INTRA image being encoded.

144. The video encoder of any of Claims 141 to 143, wherein the coefficient types respectively associated with the encoded selected coefficients form a first group of coefficient types; and

the video encoder further comprises:

a transmitting module for transmitting the encoded selected coefficients and parameters associated with coefficient types of the first group;

a transforming module for transforming pixel values for at least one block in a second enhancement original INTRA image of the enhancement layer into a set of second-image coefficients each having a coefficient type;

an encoding module for encoding only a subset of the set of second-image coefficients for said block in the second enhancement original INTRA image, wherein the coefficient types respectively associated with the encoded second-image coefficients form a second group of coefficient types;

a transmitting module for transmitting the encoded second-image coefficients and parameters associated with coefficient types of the second group not included in the first group.

145. The video encoder of Claim 144, wherein at least one parameter representative of the probabilistic distribution includes the standard deviation of the probabilistic distribution; and

the video encoder further comprises:

a computing module for computing, for each coefficient type, a standard deviation for the probabilistic distribution of coefficients having the concerned coefficient type in the enhancement original INTRA image;

a determining module for determining a number of bits necessary for representing the ratio between the maximum standard deviation, among the computed standard deviations associated with a coefficient type of the first group, and a predetermined value;

a transmitting module for transmitting, for each coefficient type of the first group, a word having a length equal to the determined number of bits and representing the standard deviation associated with the concerned coefficient type.

146. The video encoder of Claim 144 or 145, wherein the parameters associated with coefficient types of the first group are transmitted in a first transport unit and wherein the parameters associated with coefficient types of the second group not in the first group are transmitted in a second transport unit, distinct from the first transport unit.

147. The video encoder of Claim 146, wherein the encoded first-image coefficients are transmitted in the first transport unit and wherein the encoded second-image coefficients are transmitted in the second transport unit.

148. The video encoder of Claim 147, wherein the first and second transport unit are parameter transport units.

149. The video encoder according to any of Claims 146 to 148, wherein the first transport unit carries a predetermined identifier and wherein the second transport unit carries said predetermined identifier.

150. The video encoder of any of Claims 144 to 149, comprising an estimator for estimating a proximity criterion between the enhancement original INTRA image being encoded and a third enhancement original INTRA image included in the enhancement layer,

the video encoder being configured if the proximity criterion is fulfilled to :

transform pixel values for at least one block in the third enhancement original INTRA image into a set of third-image coefficients each having a coefficient type;

encode third-image coefficients for said block in the third enhancement original INTRA image, wherein the coefficient types respectively associated with the encoded third-image coefficients form a third group of coefficient types;

transmit the encoded third-image coefficients, parameters associated with coefficient types of the third group not included in the first and second groups and a flag indicating previously received parameters are valid.

151. The video encoder of Claim 150, further configured if the proximity criterion is not fulfilled to:

for each of a plurality of blocks in the third enhancement original INTRA image, transform pixel values for the concerned block into a set of third-image coefficients each having a coefficient type;

for each coefficient type, compute at least one parameter representative of a probabilistic distribution of the third-image coefficients having said coefficient type;

for at least one block in the third enhancement original INTRA image, encode third-image coefficients for said block;

transmit the encoded third-image coefficients, parameters associated with coefficient types of transmitted third-image coefficients and a flag indicating previously received parameters are no longer valid.

152. The video encoder of Claim 150 or 151, wherein the estimator is configured to estimate the proximity criterion by estimating a difference between a distortion relating to the first enhancement original INTRA image and a distortion relating to the third enhancement original INTRA image.

153. A video encoder for encoding a video sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, comprising:

a base layer encoding module for encoding a base layer made of base images;

an enhancement layer encoding module for encoding an enhancement layer made of enhancement images, including an Intra encoding module for encoding at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame prediction only; and an Inter encoding module for encoding at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame prediction;

wherein the Intra encoding module comprises:

a module for obtaining a residual enhancement image as a difference between the enhancement original INTRA image and a decoded version of the corresponding encoded base image in the base layer, the residual enhancement image comprising a plurality of blocks of pixels, each block having a block type;

a module for performing an initial segmentation of the residual enhancement image into a set of initial blocks, thus determining, for each initial block, a block type associated with the concerned initial block;

a module for determining, for each block type, an associated set of quantizers based on data corresponding to pixels of blocks having said block type;

a module for selecting, among a plurality of possible segmentations defining an association between each block of this segmentation and an associated block type, the segmentation which minimizes an encoding cost estimated based on a measure of the rate necessary for encoding each block using the set of quantizers associated with the block type of the encoded block according to the concerned segmentation.

154. The video encoder according to Claim 153, wherein the encoding cost is computed using a predetermined frame merit and a number of blocks per area unit for the concerned block type.

155. The video encoder according to Claim 153 or 154, wherein the measure of the rate is computed based on the set of quantizers associated with the concerned block type and on parameters representative of probabilistic distributions of transformed coefficients of blocks having the concerned block type.

156. The video encoder according to any of Claims 153 to 155, wherein the encoding cost includes a cost for luminance, taking into account luminance distortion generated by encoding and decoding a luminance block using the set of quantizers associated with the concerned block type, and a cost for chrominance, taking into account chrominance distortion generated by encoding and decoding a chrominance block using the set of quantizers associated with the concerned block type.

157. The video encoder according to any of Claims 153 to 156, wherein the initial segmentation into blocks is based on block activity along several spatial orientations.

158. The video encoder according to any of Claims 153 to 157, wherein the selected segmentation is represented as a quad-tree having a plurality of levels, each associated with a block size, and leaves associated with blocks and having a value indicating either a label for the concerned block or a subdivision of the concerned block.

159. The video encoder according to Claim 158, wherein the Intra encoding module comprises a tree compressor for compressing the quad tree using an arithmetic entropy coding that uses, when coding the segmentation relating to a given block, conditional probabilities for the various possible leaf values depending on a state of a block in the base layer co-located with said given block.

160. A video encoder for encoding a video sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, combining the video encoder according to any of Claims 132 to 152 and the video encoder according to any of Claims 153 to 159.

161. The video encoder according to any of Claims 132 to 160, comprising:

a down-sampling module for down-sampling video data having a first resolution to generate video data having a second resolution lower than said first resolution, and encoding the second resolution video data to obtain video data of the base layer having said second resolution;

a module for decoding the base layer video data, up-sampling the decoded base layer video data to generate decoded video data having said first resolution, forming a difference between the generated decoded video data having said first resolution and said received video data having said first resolution to generate residual data,

a module for compressing the residual data to generate video data of the enhancement layer, including determining an image segmentation into blocks for the enhancement layer, wherein the segmentation is represented as a quad-tree having a plurality of levels, each associated with a block size, and leaves associated with blocks and having a value indicating either a label for the concerned block or a subdivision of the concerned block;

a module for arithmetic entropy coding the quad-tree using, when coding the segmentation relating to a given block, conditional probabilities for the various possible leaf values depending on a state of a block in the base layer co-located with said given block.

162. The video encoder according to any of Claims 132 to 161, comprising:

a module for decoding the encoded coefficients of the enhancement original INTRA image and decoding the corresponding encoded base image in the base layer, to obtain a rough decoded image corresponding to an original image of the sequence;

a module for processing the rough decoded image through at least one adaptive post-filter adjustable depending on a parameter, wherein said parameter is derived based on pixel values and input to the adaptive post-filter.

163. The video encoder according to Claim 162 when depending on Claim 134 or 154, comprising a module for determining the parameter input to the adaptive post-filter based on the predetermined frame merit.

164. A video encoder for encoding a sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, comprising:

a base layer encoding module for encoding a base layer made of base images;

an enhancement layer encoding module for encoding an enhancement layer made of enhancement images, including an Intra encoding module for encoding at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame prediction only; and an Inter encoding module for encoding at least

one other enhancement image, referred to as enhancement original INTER image, using inter-frame prediction.

wherein the Inter encoding module comprises:

a module for selecting a prediction mode, from among a plurality of prediction modes, for predicting an enhancement block of the enhancement original INTER image, wherein the plurality of prediction modes includes at least one of:

a base mode prediction mode involving computation, from the base layer, of a base mode prediction image corresponding to the enhancement original INTER image, the base mode prediction image being composed of base mode blocks obtained using prediction information derived from prediction information of the base layer and wherein the enhancement block is derived from one or more base mode blocks of a spatially corresponding region of the base mode prediction image; and

a GRILP prediction mode including: obtaining a block predictor candidate for predicting the enhancement block within the enhancement original INTER image and an associated enhancement-layer residual block corresponding to said prediction; determining a block predictor in the base layer co-located with the determined block predictor candidate within the enhancement original INTER image; determining a base-layer residual block associated with the enhancement block in the base layer that is co-located with the enhancement block in the enhancement original INTER image, as the difference between the co-located enhancement block in the enhancement original INTER image and the determined block predictor in the base layer; determining, for the enhancement block of the enhancement original INTER image, a further residual block corresponding, at least partly, to the difference between the enhancement-layer residual block and the base-layer residual block;

a module for obtaining a prediction block from the selected prediction mode and subtracting the prediction block from the enhancement block of the enhancement original INTER image to obtain a residual block;

a module for transforming pixels values of the residual block to obtain transformed coefficients;

a module for quantizing at least one of the transformed coefficients to obtain quantized symbols;

a module for encoding the quantized symbols into encoded data.

165. The video encoder according to claim 164, wherein the plurality of prediction modes includes an inter difference mode in addition or in replacement of the GRILP mode including: performing a motion estimation on a current block of a current Enhancement Layer (EL) image to obtain a motion vector designating a reference block in a EL reference image; computing a difference image between the EL reference image and an upsampled version of the image of the base layer temporally co-located with the EL reference image; performing a motion compensation on the block of the obtained difference image pointed by the motion vector determined during the motion estimation step to obtain a residual block; adding the obtained residual block to the reference block to obtain a final block predictor used to predict the current block.

166. The video encoder according to Claim 164 or 165, wherein the plurality of prediction modes includes the following prediction modes:

a motion compensated temporal prediction mode within the enhancement layer;

an intra base layer prediction mode where the prediction block is taken from a base block co-located with the enhancement block in an up-sampled decoded version of the corresponding base image;

the base mode prediction mode, wherein each base mode block of the base mode prediction image derives from the co-located base block in the corresponding base image when the co-located base block is intra coded or derives, when the co-located base block is inter coded into a base residual using prediction information, from the block in the enhancement layer that is obtained by applying an up-sampled version of a motion vector of the prediction information onto the base mode block and from an up-sampled decoded version of the base residual;

the GRILP prediction mode and/or the inter difference prediction mode; and

a difference INTRA coding mode.

167. The video encoder according to Claim 164, 165 or 166, wherein the Inter encoding module is configured to, in the GRILP prediction mode, when an image of residual data used for the encoding of the base layer is available,

determine the base-layer residual block in the base layer by:

determining the overlap in the image of residual data between the determined block predictor and the block predictor used in the encoding of the block co-located with the enhancement block in the base layer; and

using the part in the image of residual data corresponding to this overlap if any to compute a part of said further residual block of the

enhancement original INTER image, wherein the samples of said further residual block of the enhancement original INTER image corresponding to this overlap each corresponds to a difference between a sample of the enhancement-layer residual block and a corresponding sample of the base-layer residual block.

168. The video encoder according to any of Claims 164 to 167, wherein, in the GRILP prediction mode, the determination of a predictor of the enhancement block is made using a cost function adapted to take into account the prediction of the enhancement-layer residual block to determine a rate distortion cost.

169. The video encoder according to any of Claims 164 to 168, comprising a module for de-blocking filtering the base mode prediction image before it is used to provide prediction blocks.

170. The video encoder according to Claim 169, wherein the de-blocking filtering module is configured to applied the deblocking-filtering to the boundaries of the base mode blocks of the base mode prediction image.

171. The video encoder according to Claim 169 or 170, further comprising a module for deriving the organisation of transform units of base blocks in the base layer towards the enhancement layer wherein the de-blocking filtering is applied to the boundaries of the transform units derived from the base layer.

172. The video encoder according to any of Claims 164 to 171, wherein the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained; and

the Inter encoding module comprises a module for encoding the motion information using a motion information predictor taken from a set including motion information, if any, associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

173. The video encoder according to Claim 172, wherein other motion information of the set is derived from the motion information by adding respective spatial offsets.

174. The video encoder according to Claim 172 or 173, wherein the set includes motion information, if any, associated with blocks spatially neighbouring the enhancement block in the enhancement original INTER image, and if no motion information exist for a given neighbouring block, motion information, if any, associated

with the base block spatially corresponding to the given neighbouring block in the corresponding base image.

175. The video encoder according to any of Claims 164 to 174, wherein the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained;

the Inter encoding module further comprises a module for encoding the motion information using a motion information predictor taken from a set including more motion information predictors than another set usable for predicting motion information associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

176. The video encoder according to Claim 175, wherein the set of motion information predictors for the enhancement original INTER image includes at least one motion information predictor generated based on motion information from the base layer.

177. The video encoder according to any of Claims 164 to 176, wherein the set of vector predictors comprises at least one temporal motion information predictor and at least one spatial motion information predictor, the at least one temporal motion information predictor being positioned before the at least one spatial motion information predictor.

178. The video encoder according to any of Claims 164 to 177, , wherein in case of spatial scalability between the base layer and the enhancement layer with the base layer having a lower spatial resolution than the enhancement layer; and

prediction information, such as a motion vector, is derived or up-sampled for a processing block of size 2Nx2N in the enhancement original INTER image from the base layer of lower spatial resolution, the derivation or up-sampling comprises:

determining whether or not the region of the base layer, spatially corresponding to the processing block, is wholly located within one elementary prediction unit of the base layer; and

in the case where the region of the base layer spatially corresponding to the processing block is fully located within one elementary prediction unit of the base layer, deriving prediction information for that processing block from the base layer prediction information of the said one elementary prediction unit;

otherwise in the case where the region of the base layer spatially corresponding to the processing block overlaps, at least partially, each of a plurality of elementary prediction units,

dividing the processing block into a plurality of sub-processing blocks , each of size NxN such that the region of the base layer spatially corresponding to each sub-processing block is wholly located within one elementary prediction unit of the base layer; and

deriving the prediction information for each sub-processing block from the base layer prediction information of the spatially corresponding elementary prediction unit.

179. The video encoder according to Claim 178, wherein in the case where the corresponding elementary prediction unit of the base layer is Intra-coded then the processing block is predicted from the elementary prediction unit reconstructed and resampled to the enhancement layer resolution.

180. The video encoder according to Claim 179, wherein in the case where the corresponding elementary prediction unit is Inter-coded then the processing block is temporally predicted using motion information derived from the said corresponding elementary prediction unit of the base layer.

181. The video encoder according to any of Claims 178 to 180, wherein the processing block is temporally predicted further using a decoded temporal residual of the corresponding elementary prediction unit of the base layer, said temporal residual being computed between base layer images, as a function of the motion information of the elementary prediction unit.

182. The encoding method according to the preceding claim, wherein the spatial scaling between an image of the enhancement layer and a corresponding image of the base layer is a non-integer ratio

183. The encoding method according to the preceding claim, wherein the non-integer ratio is 1.5.

184. The video encoder according to any of Claims 164 to 181, wherein the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the video encoder comprises:

a module for obtaining a first set of quantization offsets to be applied to a group of enhancement images, a different quantization offset being obtained for at least two enhancement INTER images belonging to the same temporal depth; and

a module for determining a quantization parameter used to encode the enhancement INTER images based on the obtained first set of quantization parameters.

185. The video encoder according to any of Claims 164 to 181, wherein the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the video encoder comprises:

a module for obtaining a first set of quantization offsets to be applied to a group of enhancement images, the same quantization offset being obtained for at least two enhancement INTER images belonging to different temporal depths; and

a module for determining a quantization parameter used to encode the enhancement INTER images based on the obtained first set of quantization parameters.

186. The video encoder according to any of Claims 184 to **Erreur ! Source du renvoi introuvable.**, configured to obtain, among the at least two enhancement INTER images that belong to the same temporal depth, a first offset for a first enhancement INTER image having a reference image of a first quality and to obtain a second offset, larger than said first offset, for a second enhancement INTER image having a reference image of a second quality lower than said first quality.

187. The video encoder according to any of Claims 184 to 186, wherein the quantization offset obtained for an enhancement INTER image takes into account:

the number of enhancement images using the enhancement INTER image as a reference image for temporal prediction;

the temporal distance of the enhancement INTER image to its reference image having the lowest quantization offset; and

the value of the quantization parameter applied to this reference image.

188. The video encoder according to any of Claims 184 to 187, wherein the quantization offset obtained for an enhancement INTER image is equal or larger than the quantization offset of its reference image having the lowest quantization offset.

189. The video encoder according to any of Claims 184 to 188, further comprising a module for encoding, using a second set of different quantization offsets, a group of base images in the base layer that temporally coincides with the group of enhancement images ;

wherein the second set is obtained based on a temporal depth each base image belongs to.

190. The video encoder according to any of Claims 166 to 189, wherein the Inter encoding module further comprises a module for encoding, in the bit-stream, quad-trees representing a segmentation of respective portions of the enhancement original INTER image and coding modes associated with blocks in the segmentation;

wherein the coding mode associated with a given block is encoded through

a first syntax element that indicates whether the coding mode associated with the given block is based on temporal/Inter prediction or not,

a second coding mode syntax element that indicates whether a prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information is used or not for encoding the block if the first coding mode syntax element refers to temporal/Inter prediction, or indicates whether the coding mode associated with the given block is a conventional Intra prediction or based on Inter-layer prediction if the first coding mode syntax element refers to non temporal/Inter prediction, and

in the case in which the coding mode associated with the given block is based on Inter-layer prediction , a third coding mode syntax element that indicates whether the coding mode associated with the given block is the intra base layer mode or the base mode prediction mode.

191. The video encoder according to claim 190 wherein the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the GRILP prediction sub-mode.

192. The video encoder according to claim 190 wherein the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the inter difference prediction sub-mode.

193. The video encoder according to claim 191 or 192 wherein, a fourth coding mode syntax element indicates whether the inter difference block is used or not for encoding the block if the second coding mode syntax element indicates that the GRILP mode is not used or whether the GRILP mode is used or not for encoding the block if the second coding mode syntax element indicates that the inter difference mode is not used.

194. The video encoder according to any claim 191 to 193 wherein, at least one high level syntax element indicates which of the coding mode syntax elements are present or not in the bit-stream.

195. The video encoder according to claim 194 wherein, when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding mode syntax element following the removed coding mode syntax element replaces the removed coding mode syntax element and the coding order of the remaining coding mode syntax elements is kept.

196. The video encoder according to claim 195 wherein, when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding order of the remaining coding mode syntax elements is modified.

197. The video encoder according to claim 196 wherein the modification of the coding order of the remaining coding mode syntax elements takes into account the probability of occurrence of coding modes represented by remaining coding mode syntax elements.

198. A video encoder for encoding a video sequence of images of pixels into a scalable video bit-stream according to a scalable encoding scheme, combining the video encoder according to any of Claims 132 to 163 and the video encoder according to any of Claims 164 to 190.

199. The video encoder according to Claim 198, wherein

the Intra encoding module comprises a module for selecting quantizers from the predetermined block merit to quantize the selected coefficients, the predetermined block merit deriving from a frame merit;

the Inter encoding module comprises a module for selecting quantizers from a quantization parameter to quantize the transformed coefficients; and

the frame merit and the quantization parameter are computed from a user-specified quality parameter and are linked together with a balancing parameter.

200. The video encoder according to any of the preceding claims, implemented by a computer, wherein data, such as image samples, in the base layer are processed using 8-bit words and data, such as image samples, in the enhancement layer are processed using 10-bit words.

201. The video encoder according to Claim 200, wherein data, such as image samples, from the base layer are up-scaled to 10-bit words by multiplying each value by 4, when this data from the base layer is processed for the encoding of the enhancement layer.

202. A video decoder for decoding a scalable video bit-stream, comprising:

a base layer decoding module decoding a base layer made of base images;

an enhancement layer decoding module decoding an enhancement layer made of enhancement images, including an Intra decoding module for decoding data representing at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame decoding; and an Inter decoding module for decoding data representing at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame decoding;

wherein the Intra decoding module for decoding data representing at least one block of pixels in the enhancement original INTRA image, comprises :

a module for receiving said data and parameters each representative of a probabilistic distribution of a coefficient type;

a module for decoding said data into symbols;

a module for selecting coefficient types for which a coefficient encoding merit prior to encoding, estimated based on the parameter associated with the concerned coefficient type, is greater than a predetermined block merit;

a module for dequantizing, for selected coefficient types, symbols into dequantized coefficients having a coefficient type among the selected coefficient types;

a module for transforming dequantized coefficients into pixel values in the spatial domain for said block.

203. The video decoder according to Claim 202, comprising a module for determining the predetermined block merit based on a predetermined frame merit and on a number of blocks of a block type of the block per area unit.

204. The video decoder according to Claim 203, wherein the module for determining the predetermined frame merit is configured to determine a frame merit and a distortion at the image level using a balancing parameter such that a video merit, computed based on said distortion and said frame merit, corresponds to a target video merit.

205. The video decoder according to Claim 204, wherein the module for determining the predetermined frame merit is configured to determine a frame merit and a distortion at the image level such that a product of the determined distortion at the image level and of the target video merit essentially equals a product of the balancing parameter and the determined frame merit.

206. The video decoder according to any of Claims 202 to 205, wherein the predetermined frame merit is decoded from the bit-stream.

207. The video decoder according to any of Claims 202 to 206, wherein the enhancement original INTRA image is a luminance image, the enhancement layer comprises at least one corresponding colour image comprising a plurality of colour blocks; and the video decoder comprises :

a module for determining a colour frame merit;

a module for decoding data associated with a colour block among said plurality of colour blocks into a set of symbols each corresponding to a coefficient type, said block having a particular block type;

a module for determining a colour block merit based on the colour frame merit and on a number of blocks of the particular block type per area unit;

a module for selecting coefficient types based, for each coefficient type, on a coefficient encoding merit prior to encoding, for said coefficient type, and on the colour block merit;

a module for dequantizing, for selected coefficient types, symbols into dequantized coefficients having a coefficient type among the selected coefficient types;

a module for transforming dequantized coefficients into pixel values in the spatial domain for said colour block.

208. The video decoder according to Claim 207, wherein the module for determining the colour frame merit uses a balancing parameter.

209. The video decoder according to Claim 208, wherein the module for determining the predetermined frame merit is configured to determine a frame merit and a distortion at the image level such that a product of the determined distortion at the image level and of a target video merit essentially equals the determined frame merit and the module for determining the colour frame merit is configured to determine the colour frame merit such that a product of a corresponding distortion for the colour frame and of the target video merit essentially equals a product of the balancing parameter and the determined colour frame merit.

210. The video decoder according to any of Claims 202 to 209, wherein the coefficient encoding merit prior to encoding for a given coefficient type estimates a ratio between a distortion variation provided by encoding a coefficient having the given type and a rate increase resulting from encoding said coefficient.

211. The video decoder according to any of Claims 202 to 210, wherein the Intra decoding module comprises a quantizer selector for selecting, for each coefficient for

which the coefficient encoding merit prior to encoding is greater than the predetermined block merit, a quantizer depending on the received parameter associated with the concerned coefficient type and on the predetermined block merit, wherein the module for dequantizing symbols uses the selected quantizer.

212. The video decoder according to Claim 211, wherein the Intra decoding module comprises a module for determining the coefficient encoding merit prior to encoding for given coefficient type and block type based on the received parameters for the given coefficient type and block type.

213. The video decoder of any of Claims 202 to 212, wherein a parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type previously received for a previous enhancement INTRA image is reused as the at least one parameter representative of a probabilistic distribution of coefficients having the concerned coefficient type in the enhancement original INTRA image being decoded.

214. The video decoder according to any of Claims 202 to 213, wherein the selected coefficient types of the enhancement original INTRA image being decoded belong to a first group; and

the video decoder further comprises :

a module for receiving encoded coefficients relating to a second enhancement original INTRA image of the enhancement layer and having coefficient types in a second group;

a module for receiving parameters associated with coefficient types of the second group not included in the first group;

a module for decoding the received coefficients relating to the second enhancement original INTRA image, wherein decoding a received coefficient having a given coefficient type in the first group includes a step of dequantizing using a dequantizer selected based on the previously received parameter associated with the given coefficient type;

a module for transforming the decoded coefficients into pixel values for the second enhancement original INTRA image.

215. The video decoder according to Claim 214, wherein the parameters associated with coefficient types of the first group are received in a first transport unit and wherein the parameters associated with coefficient types of the second group not in the first group are received in a second transport unit, distinct from the first transport unit.

216. The video decoder according to the preceding claim, wherein the information supplied to the decoder for said second image does not include information about the reused parameter(s).

217. The video decoder according to the preceding claim, wherein such a parametric probabilistic model is obtained for each type of encoded DCT coefficient in said first image.

218. The video decoder according to the preceding claim, wherein parameters of the first-image parametric probabilistic model obtained for at least one said DCT coefficient type are reused for said second image.

219. The video decoder according to Claim 214 or 215, comprising a step of receiving encoded coefficients relating to a third enhancement original INTRA image of the enhancement layer and a flag indicating whether previously received parameters are valid,

the video decoder is configured if the received flag indicate that the previously received parameters are valid, to:

receive parameters associated with coefficient types of a third group not included in the first and second groups;

decode the received coefficients relating to the third enhancement original INTRA image, wherein decoding a received coefficient having a given coefficient type in the first or second group includes a step of dequantizing using a dequantizer selected based on the previously received parameter associated with the given coefficient type;

transform the decoded coefficients into pixel values for the third enhancement original INTRA image;

220. The video decoder according to Claim 219, configured if the received flag indicate that the previously received parameters are no longer valid, to:

receive encoded coefficients relating to the third enhancement original INTRA image and having coefficient types in a first group;

receive new parameters associated with coefficient types of encoded coefficients relating to the third enhancement original INTRA image;

decode the received coefficients relating to the third enhancement original INTRA image, wherein decoding a received coefficient having a given coefficient type includes a step of dequantizing using a dequantizer selected based on the received new parameter associated with the given coefficient type;

transform the decoded coefficients into pixel values for the third enhancement original INTRA image.

221. The video decoder according to any of Claims 202 to 220, further comprising a module for decoding, from the bit-stream, a quad-tree representing a segmentation of the enhancement original INTRA image said plurality of blocks of pixels, each block having a block type, the quad tree having a plurality of levels, each associated with a block size, and leaves associated with blocks and having a value indicating either a label for the concerned block or a subdivision of the concerned block.

222. The video decoder according to the preceding claim, wherein the module for decoding the quad tree uses an arithmetic entropy decoding that uses, when decoding the segmentation relating to a given block, conditional probabilities for the various possible leaf values depending on a state of a block in the base layer co-located with said given block.

223. The video decoder according to Claim 202 to 222, comprising:

a module for receiving video data of the base layer, video data of the enhancement layer, a table of conditional probabilities and a coded quad-tree representing, by leaf values, an image segmentation into blocks for the enhancement original INTRA image;

a module for decoding video data of the base layer to generate decoded base layer video data having a second resolution, lower than a first resolution, and up-sampling the decoded base layer video data to generate up-sampled video data having the first resolution;

a module for determining, for at least one block represented in the quad-tree, the probabilities respectively associated with the possible leaf values based on the received table and depending on a state of a block in the base layer co-located with said block;

a module for decoding the coded quad-tree to obtain the segmentation, including arithmetic entropy decoding the leaf value associated with said block using the determined probabilities;

a module for decoding, using the obtained segmentation, video data of the enhancement layer to generate residual data having the first resolution;

a module for forming a sum of the up-sampled video data and the residual data to generate enhanced video data.

224. The video decoder according to Claim 202 to 223, comprising a module for determining the parameter input to the adaptive post-filter based on the predetermined frame merit.

225. A video decoder for decoding a scalable video bit-stream, comprising:

a base layer decoding module decoding a base layer made of base images;

an enhancement layer decoding module decoding an enhancement layer made of enhancement images, including an Intra decoding module for decoding data representing at least one enhancement image, referred to as enhancement original INTRA image, using intra-frame decoding; and an Inter decoding module for decoding data representing at least one other enhancement image, referred to as enhancement original INTER image, using inter-frame decoding;

wherein the Inter decoding module for decoding data representing the enhancement original INTER image comprising a plurality of blocks of pixels, each block having a block type, comprises:

a module for decoding prediction mode information from the bit-stream for at least one enhancement block of the enhancement original INTER image to obtain a prediction mode having been selected from among a plurality of prediction modes, wherein the plurality of prediction modes includes at least one of:

a base mode prediction mode involving computation, from the base layer, of a base mode prediction image corresponding to the enhancement original INTER image, the base mode prediction image being composed of base mode blocks obtained using prediction information derived from prediction information of the base layer and wherein the enhancement block is derived from one or more base mode blocks of a spatially corresponding region of the base mode prediction image; and

a GRILP prediction mode including: obtaining from the bit-stream the location of a block predictor of the enhancement block within the enhancement original INTER image to be decoded and a residual block comprising difference information between enhancement image residual information and base layer residual information; determining a block predictor in the base layer co-located with the block predictor in the enhancement original INTER image; determining a base-layer residual block corresponding to the difference between the block of the base layer co-located with the enhancement block to be decoded and the determined block predictor in the base layer;

reconstructing an enhancement-layer residual block using the determined base-layer residual block and said residual block obtained from the bit stream; reconstructing the enhancement block using the block predictor and the enhancement-layer residual block;

a module for obtaining a prediction block from the selected prediction mode and adding the prediction block to a decoded enhancement residual block to obtain the enhancement block, said enhancement residual block comprising quantized symbols, the decoding of the enhancement original INTER image comprising inverse quantizing these quantized symbols to obtain transformed coefficients.

226. The video decoder according to Claim 225, wherein the plurality of prediction modes includes an inter difference mode in addition or in replacement of the GRILP mode including for a block to decode: obtaining a motion vector designating a reference block in a EL reference image; computing a difference image between the EL reference image and an upsampled version of the image of the base layer temporally co-located with the EL reference image; performing a motion compensation on the block of the obtained difference image pointed by the motion vector determined during the motion estimation step to obtain a residual block; adding the obtained residual block to the reference block to obtain a final block predictor used to decode the current block.

227. The video decoder of Claim 225, wherein the plurality of prediction modes includes the following prediction modes:

a motion compensated temporal prediction mode within the enhancement layer;

an intra base layer prediction mode where the prediction block is taken from a base block co-located with the enhancement block in an up-sampled decoded version of the corresponding base image;

the base mode prediction mode, wherein each base mode block of the base mode prediction image derives from the co-located base block in the corresponding base image when the co-located base block is intra coded or derives, when the co-located base block is inter coded into a base residual using prediction information, from the block in the enhancement layer that is obtained by applying an up-sampled version of a motion vector of the prediction information onto the base mode block and from an up-sampled decoded version of the base residual;

the GRILP prediction mode and/or the inter difference prediction mode; and

a difference INTRA coding mode.

228. The video decoder according to Claim 225 or 227, configured, in the GRILP prediction mode, when an image of residual data of the base layer is available to,

determine the base-layer block residual in the base layer by:

determining the overlap in the image of residual data between the obtained block predictor and the block predictor used in the encoding of the block co-located with the enhancement block in the base layer;

using the part in the image of residual data corresponding to this overlap if any to reconstruct a part of the enhancement-layer residual block, wherein the samples of the enhancement-layer residual block corresponding to this overlap each involves an addition of a sample of the obtained residual block and a corresponding sample of the base-layer residual block.

229. The video encoder according to any of Claims 225 to 228, comprising a module for de-blocking filtering the base mode prediction image before it is used to provide prediction blocks.

230. The video encoder according to Claim 229, wherein the de-blocking filtering module applies de-blocking filtering to the boundaries of the base mode blocks of the base mode prediction image.

231. The video encoder according to Claim 229 or 230, further comprising a module for deriving the organisation of transform units of base blocks in the base layer towards the enhancement layer wherein the de-blocking filtering module applies de-blocking filtering to the boundaries of the transform units derived from the base layer.

232. The video decoder according to any of Claims 225 to 231, wherein the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained; and

the Intra decoding module further comprises a module for decoding the motion information using a motion information predictor taken from a set including motion information, if any, associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

233. The video encoder according to Claim 232, wherein other motion information of the set is derived from the motion information by adding respective spatial offsets.

234. The video encoder according to Claim 232 or 233, wherein the set includes motion information, if any, associated with blocks spatially neighbouring the

enhancement block in the enhancement original INTER image, and if no motion information exist for a given neighbouring block, motion information, if any, associated with the base block spatially corresponding to the given neighbouring block in the corresponding base image.

235. The video decoder according to any of Claims 225 to 234, wherein the plurality of prediction modes includes a motion compensated temporal prediction mode within the enhancement layer;

the motion compensated temporal prediction mode being selected for an enhancement block, motion information including a motion vector is obtained;

the Intra decoding module further comprises a module for decoding the motion information using a motion information predictor taken from a set including more motion information predictors than another set usable for predicting motion information associated with the base block spatially corresponding to the enhancement block in the corresponding base image.

236. The video encoder according to Claim 235, wherein the set of motion information predictors for the enhancement original INTER image includes at least one motion information predictor generated based on motion information from the base layer.

237. The video decoder according to any of Claims 225 to 236, wherein the set of vector predictors comprises at least one temporal motion information predictor and at least one spatial motion information predictor, the at least one temporal motion information predictor being positioned before the at least one spatial motion information predictor.

238. The video decoder according to any of Claims 225 to 237, wherein in case of spatial scalability between the base layer and the enhancement layer with the base layer having a lower spatial resolution than the enhancement layer; and

prediction information, such as a motion vector, is derived or up-sampled for a processing block of size 2Nx2N in the enhancement original INTER image from the base layer of lower spatial resolution, the derivation or up-sampling comprises:

determining whether or not the region of the base layer, spatially corresponding to the processing block, is wholly located within one elementary prediction unit of the base layer; and

in the case where the region of the base layer spatially corresponding to the processing block is fully located within one elementary prediction unit of the base

layer, deriving prediction information for that processing block from the base layer prediction information of the said one elementary prediction unit;

otherwise in the case where the region of the base layer spatially corresponding to the processing block overlaps, at least partially, each of a plurality of elementary prediction units,

dividing the processing block into a plurality of sub-processing blocks , each of size NxN such that the region of the base layer spatially corresponding to each sub-processing block is wholly located within one elementary prediction unit of the base layer; and

deriving the prediction information for each sub-processing block from the base layer prediction information of the spatially corresponding elementary prediction unit.

239. The video decoder according to Claim 238, wherein in the case where the corresponding elementary prediction unit of the base layer is Intra-coded then the processing block is predicted from the elementary prediction unit reconstructed and resampled to the enhancement layer resolution.

240. The video decoder according to Claim 238, wherein in the case where the corresponding elementary prediction unit is Inter-coded then the processing block is temporally predicted using motion information derived from the said corresponding elementary prediction unit of the base layer.

241. The video decoder according to any of Claims 238 to 240, wherein the processing block is temporally predicted further using a decoded temporal residual of the corresponding elementary prediction unit of the base layer, said temporal residual being computed between base layer images, as a function of the motion information of the elementary prediction unit.

242. The encoding method according to the preceding claim, wherein the spatial scaling between an image of the enhancement layer and a corresponding image of the base layer is a non-integer ratio

243. The encoding method according to the preceding claim, wherein the non-integer ratio is 1.5.

244. The video decoder according to any of Claims 225 to 241, wherein the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the video decoder comprises:

a module for obtaining a first set of quantization offsets to be applied to a group of enhancement images, a different quantization offset being obtained for at least two enhancement INTER images belonging to the same temporal depth; and

a module for determining a quantization parameter used for inverse quantization to decode the enhancement INTER images based on the obtained first set of quantization parameters.

245. The video decoder according to any of Claims 225 to 244, wherein the enhancement layer is divided into groups of enhancement images comprising a plurality of temporal depths associated with the enhancement images, and the video decoder comprises:

a module for obtaining a first set of quantization offsets to be applied to a group of enhancement images, the same quantization offset being obtained for at least two enhancement INTER images belonging to different temporal depths; and

a module for determining a quantization parameter used for inverse quantization to decode the enhancement INTER images based on the obtained first set of quantization parameters.

246. The video decoder according to any of Claims 244 to **Erreur ! Source du renvoi introuvable.**, configured to obtain, among the at least two enhancement INTER images that belong to the same temporal depth, a first offset for a first enhancement INTER image having a reference image of a first quality and to obtain a second offset, larger than said first offset, for a second enhancement INTER image having a reference image of a second quality lower than said first quality.

247. The video decoder according to any of Claims 244 to 246, wherein the quantization offset obtained for an enhancement INTER image takes into account:

the number of enhancement images using the enhancement INTER image as a reference image for temporal prediction;

the temporal distance of the enhancement INTER image to its reference image having the lowest quantization offset; and

the value of the quantization parameter applied to this reference image.

248. The video decoder according to any of Claims 244 to 247, wherein the quantization offset obtained for an enhancement INTER image is equal or larger than the quantization offset of its reference image having the lowest quantization offset.

249. The video decoder according to any of Claims 244 to 248, further comprising a module for encoding, using a second set of different quantization offsets,

a group of base images in the base layer that temporally coincides with the group of enhancement images ;

wherein the second set is obtained based on a temporal depth each base image belongs to.

250. The video decoder according to any of Claims 225 to 249, wherein the Inter decoding module further comprises a module for decoding, from the bit-stream, quad-trees representing a segmentation of respective portions of the enhancement original INTER image and coding modes associated with blocks in the segmentation;

wherein the module for decoding quad-trees is configured to decode, from a received code associated with a block in the segmentation,

a first coding mode syntax element that indicates whether the coding mode associated with the block is based on temporal/Inter prediction or not,

a second coding mode syntax element that indicates whether a prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information is used or not for encoding the block prediction sub-mode is activated or not if the first coding mode syntax element refers to temporal/Inter prediction, or indicates whether the coding mode associated with the block is a conventional Intra prediction or based on Inter-layer prediction if the first coding mode syntax element refers to non temporal/Inter prediction, and

in the case in which the coding mode associated with the given block is based on Inter-layer prediction , a third coding mode syntax element that indicates whether the coding mode associated with the given block is the intra base layer mode or the base mode prediction mode.

251. The video decoder according to claim 250 wherein the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the GRILP prediction sub-mode.

252. The video decoder according to claim 250 wherein the prediction sub-mode comprising inter layer residual prediction with a residual predictor obtained using enhancement layer motion information concerned by the second coding mode syntax element is the inter difference prediction sub-mode.

253. The video decoder according to claim 251 or 252 wherein, a fourth coding mode syntax element indicates whether the inter difference block was used or not for

encoding the block if the second coding mode syntax element indicates that the GRILP mode was not used or whether the GRILP mode was used or not for encoding the block if the second coding mode syntax element indicates that the inter difference mode was not used.

254. The video decoder according to any claim 250 to 253 wherein, at least one high level syntax element indicates which of the coding mode syntax elements are present or not in the bit-stream.

255. The video decoder according to claim 254 wherein, when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding mode syntax element following the removed coding mode syntax element replaces the removed coding mode syntax element and the coding order of the remaining coding mode syntax elements is kept.

256. The video decoder according to claim 255 wherein, when a high level syntax element indicates that a coding mode syntax element is not present in the bit-stream, the coding order of the remaining coding mode syntax elements is modified.

257. The video decoder according to claim 256 wherein the modification of the coding order of the remaining coding mode syntax elements takes into account the probability of occurrence of coding modes represented by remaining coding mode syntax elements.

258. A video decoder for decoding a scalable video bit-stream made, combining the video decoder according to any of Claims 202 to 224 and the video decoder according to any of Claims 225 to 250.

259. The video decoder according to Claim 258, wherein

the Intra decoding module comprises a module for selecting quantizers from the predetermined block merit to dequantize symbols of the selected coefficient types, the predetermined block merit deriving from a frame merit;

the Inter decoding module comprises a module for selecting quantizers from a quantization parameter to inverse quantize the quantized symbols; and

the frame merit and the quantization parameter are computed from a received quality parameter and are linked together with a balancing parameter..

260. The video decoder according to any of Claims 202 to 259, implemented by a computer, wherein data, such as image samples, in the base layer are processed using 8-bit words and data, such as image samples, in the enhancement layer are processed using 10-bit words.

261. The video decoder according to Claim 260, wherein data, such as image samples, from the base layer are up-scaled to 10-bit words by multiplying each value by 4, when this date from the base layer is processed for the decoding of the enhancement layer.

262. A non-transitory computer-readable medium storing a program which, when executed by a microprocessor or computer system in an apparatus, causes the apparatus to perform the steps of any of Claims 1 to 131.

263. An encoding device for encoding an image substantially as herein described with reference to, and as shown in, Figure 7; Figures 7 and 28; Figures 7, 28 and 42; Figures 7, 28, 42 and at least one from Figures 33, 34, 35, 36A, 38, 39 and 44; Figure 9; Figures 9 and 11; or Figures 9, 11 and at least one from Figures 21, 21A, 21B, 22, 24 and 25 of the accompanying drawings.

264. A decoding device for decoding a scalable video bit-stream substantially as herein described with reference to, and as shown in, Figure 8; Figures 8 and 29; Figures 8, 29 and 43; Figures 8, 29, 43 and at least one from Figures 33, 34, 35, 36A, 38, 39 and 44; Figure 10; Figures 10 and 12; or Figures 10, 12 and at least one from Figures 21, 21A, 21B, 24A and 25A of the accompanying drawings.

265. An encoding method for encoding an image substantially as herein described with reference to, and as shown in, Figure 7; Figures 7 and 28; Figures 7, 28 and 42; Figures 7, 28, 42 and at least one from Figures 33, 34, 35, 36A, 38, 39 and 44; Figure 9; Figures 9 and 11; or Figures 9, 11 and at least one from Figures 21, 21A, 21B, 22, 24 and 25 of the accompanying drawings.

266. A decoding method for decoding a scalable video bit-stream substantially as herein described with reference to, and as shown in, Figure 8; Figures 8 and 29; Figures 8, 29 and 43; Figures 8, 29, 43 and at least one from Figures 33, 34, 35, 36A, 38, 39 and 44; Figure 10; Figures 10 and 12; or Figures 10, 12 and at least one from Figures 21, 21A, 21B, 24A and 25Aof the accompanying drawings.

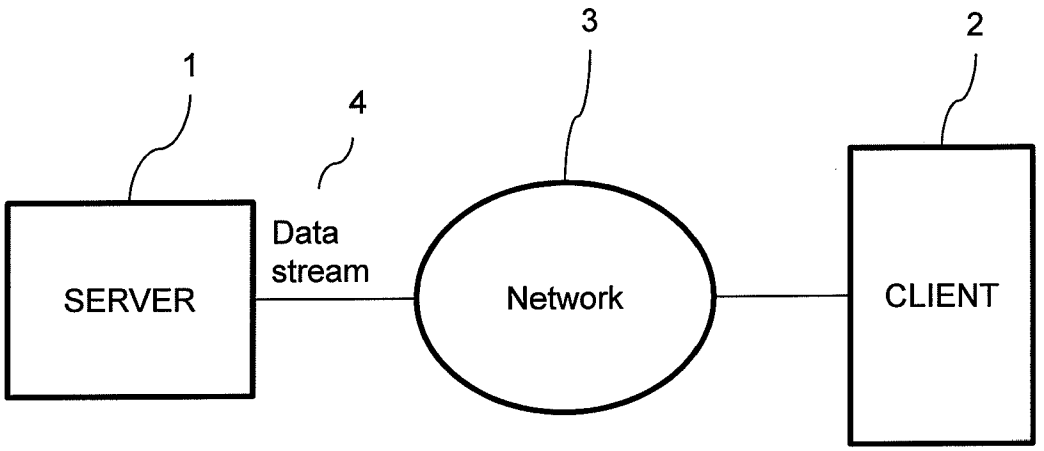

1
4
3
2
SERVER
Data stream
Network
CLIENT

Fig. 1A

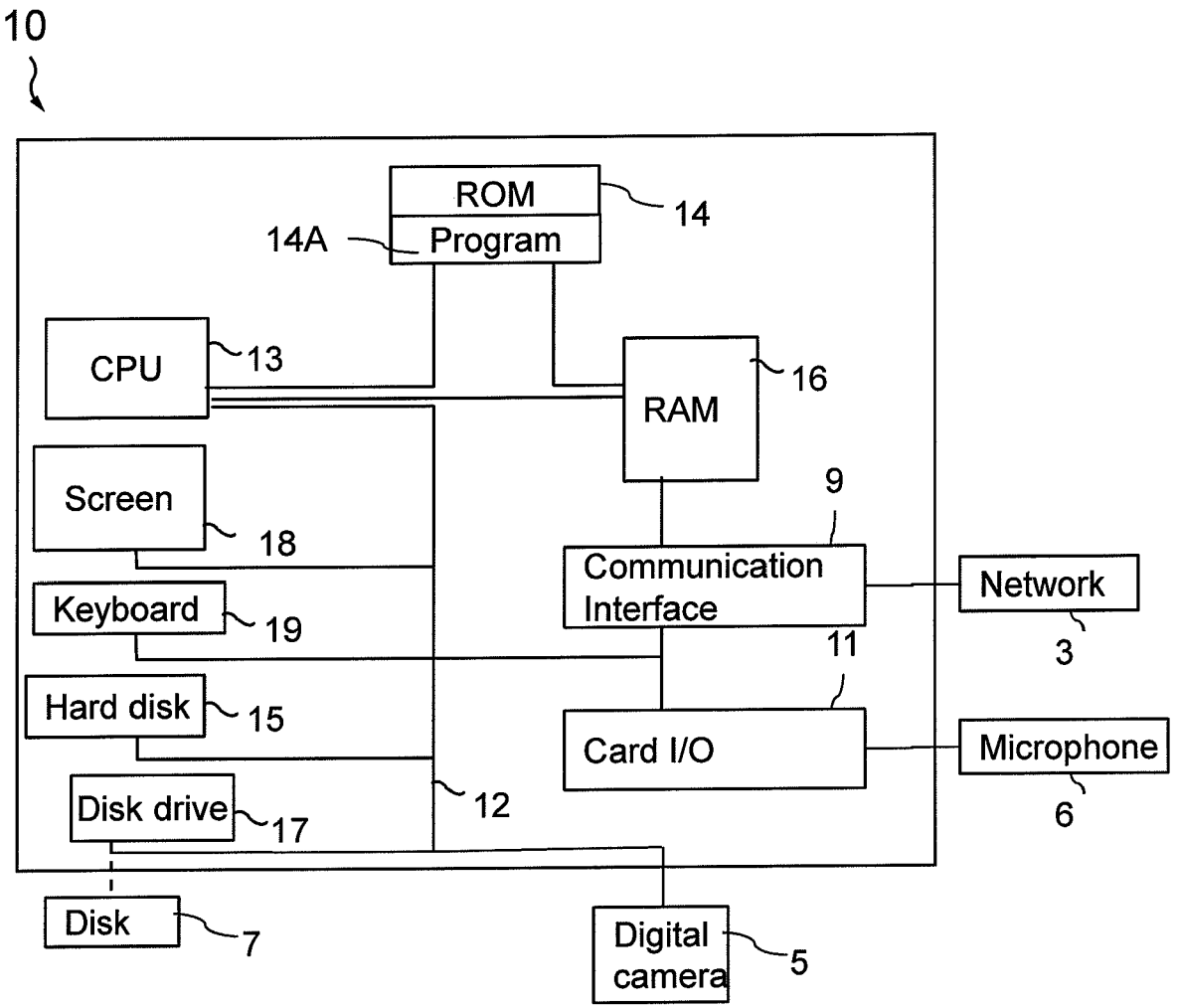

10
ROM
14
14A
Program
CPU
13
RAM
16
Screen
18
9
Communication Interface
Network
3
Keyboard
19
11
Hard disk
15
Card I/O
Microphone
6
Disk drive
17
12
Disk
7
Digital camera
5

Fig. 1B

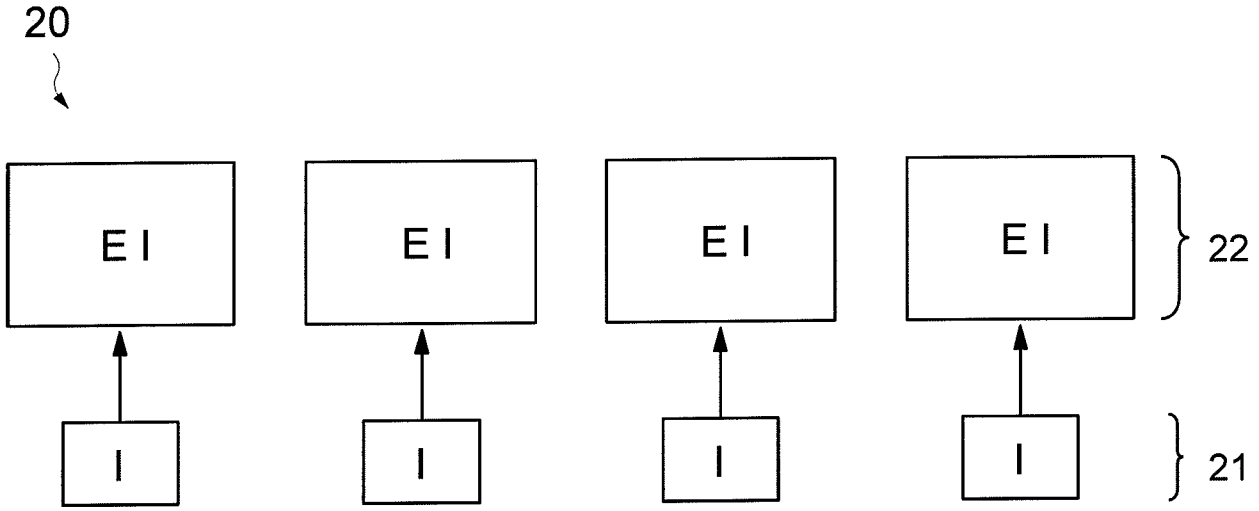

20
E I
E I
E I
E I
22
I
I
I
I
21

Fig. 2

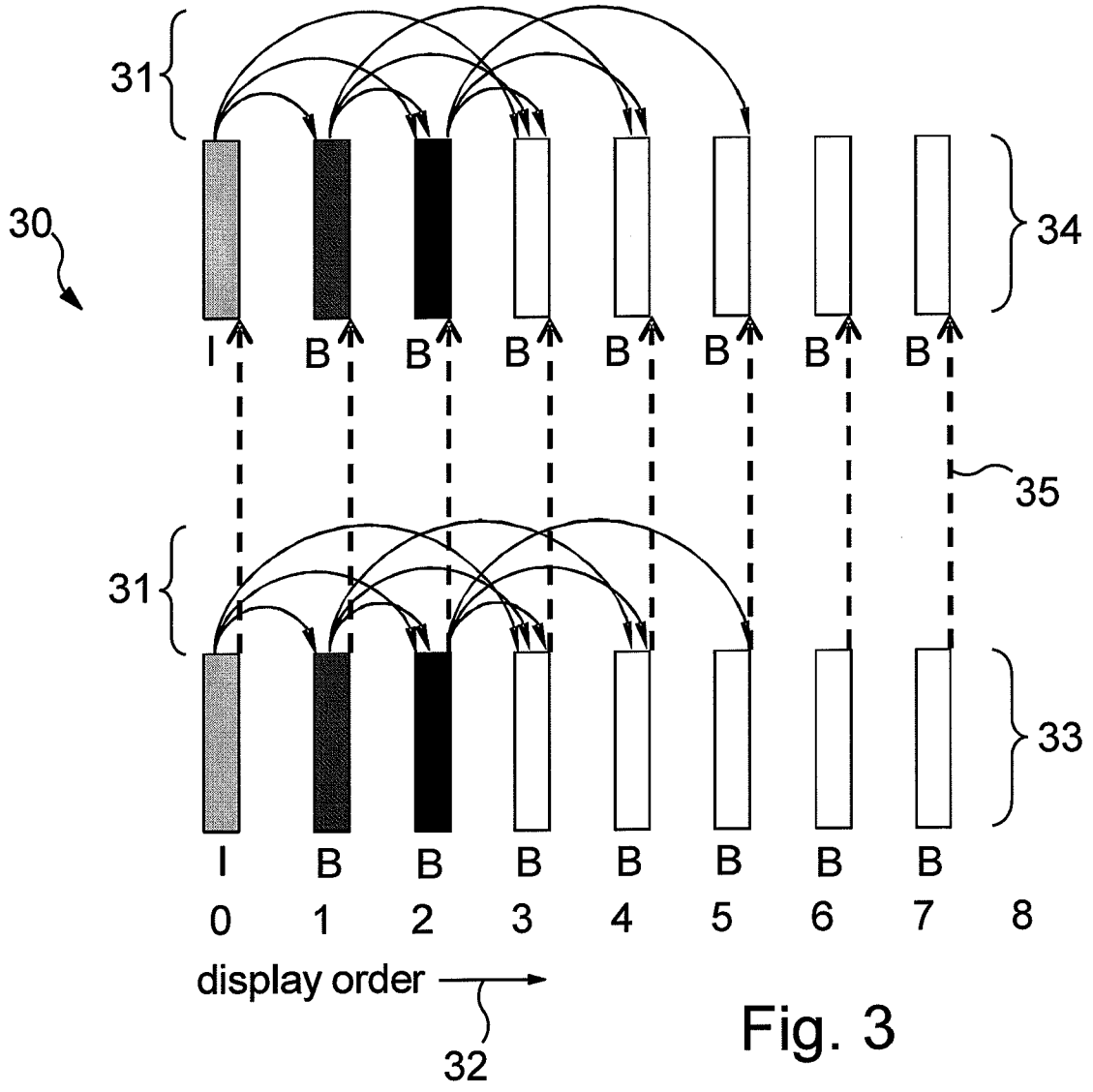

31
30
34
I B B B B B B B
35
31
33
I B B B B B B B
0 1 2 3 4 5 6 7 8
display order
32

Fig. 3

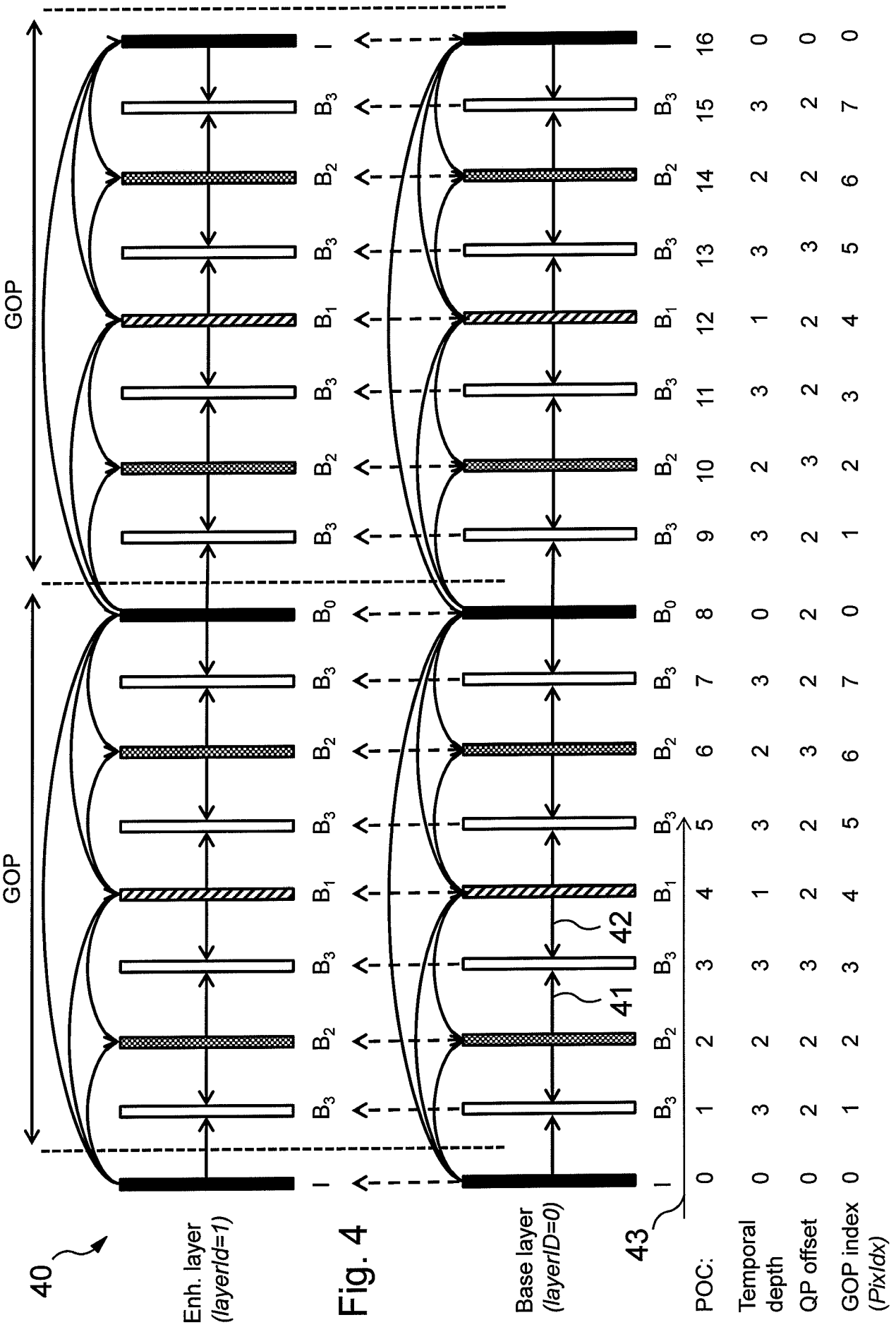

40
GOP
GOP
Enh. layer (layerId=1)
Base layer (layerID=0)
I
B3
B2
B3
B1
B3
B2
B3
B0
B3
B2
B3
B1
B3
B2
B3
I
42
41
43
POC: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Temporal depth 0 3 2 3 1 3 2 3 0 3 2 3 1 3 2 3 0
QP offset 0 2 2 3 2 2 3 2 2 2 3 2 2 3 2 2 0
GOP index (PixIdx) 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0

Fig. 4

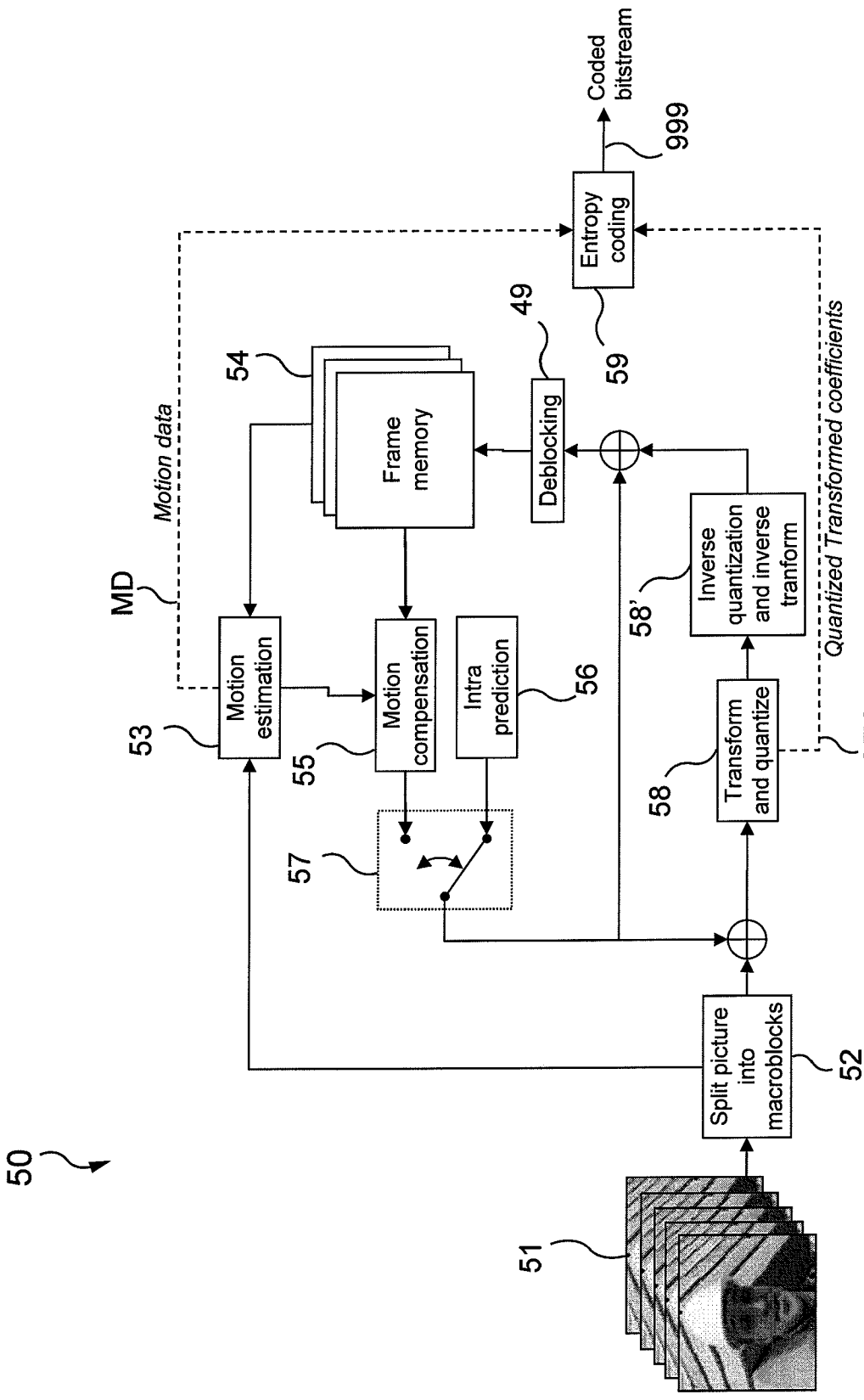
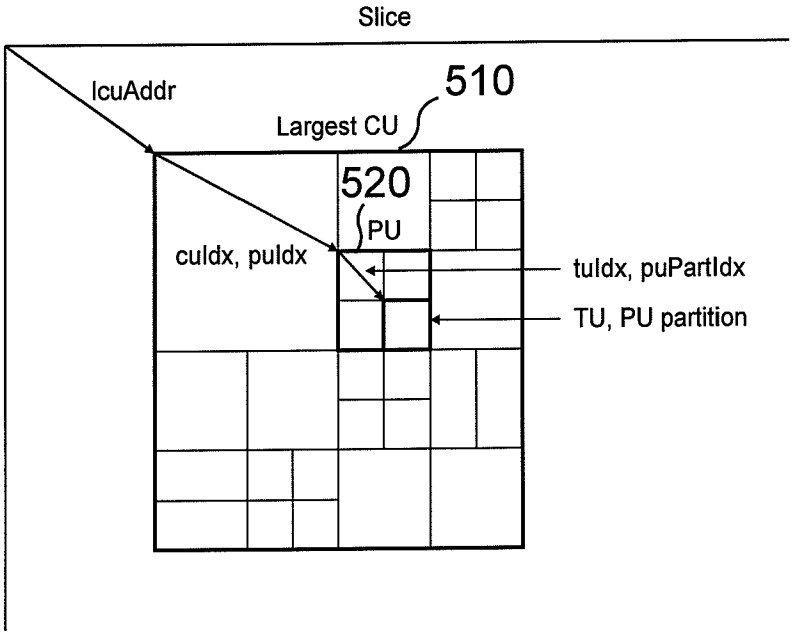

50
51
Split picture into macroblocks
52
Transform and quantize
58
QTC
Quantized Transformed coefficients
Inverse quantization and inverse tranform
58'
Entropy coding
59
Coded bitstream
999
Deblocking
49
Frame memory
54
Motion data
MD
Motion estimation
53
Motion compensation
55
Intra prediction
56
57

Fig. 5


Slice
lcuAddr
510
Largest CU
520
PU
culdx, puIdx
tuIdx, puPartIdx
TU, PU partition

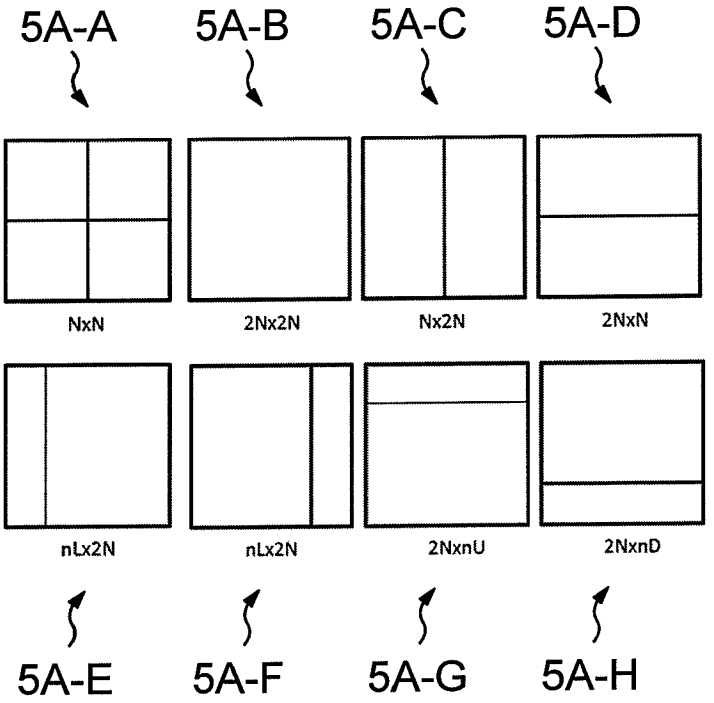

5A-A
5A-B
5A-C
5A-D
NxN
2Nx2N
Nx2N
2NxN
nLx2N
nLx2N
2NxnU
2NxnD
5A-E
5A-F
5A-G
5A-H

Fig. 5A

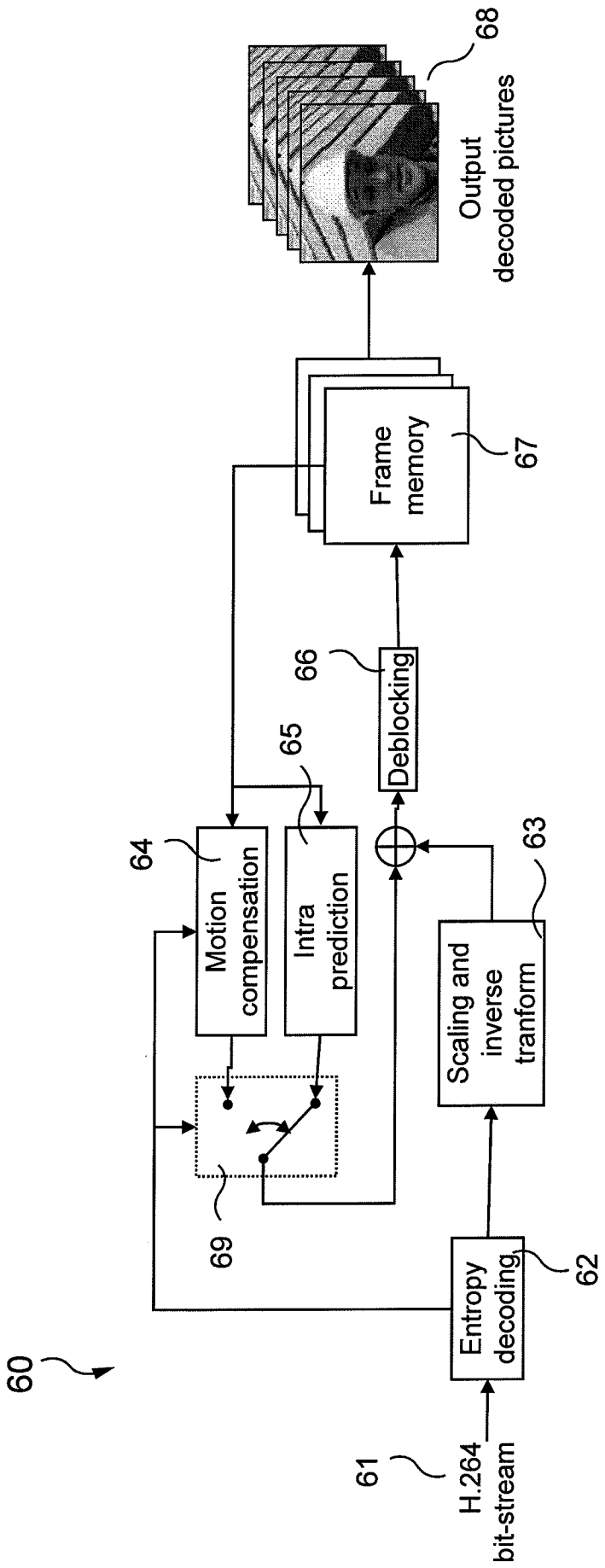

60
61
H.264
bit-stream
Entropy decoding
62
Scaling and inverse tranform
63
Motion compensation
64
Intra prediction
65
Deblocking
66
Frame memory
67
Output decoded pictures
68
69

Fig. 6

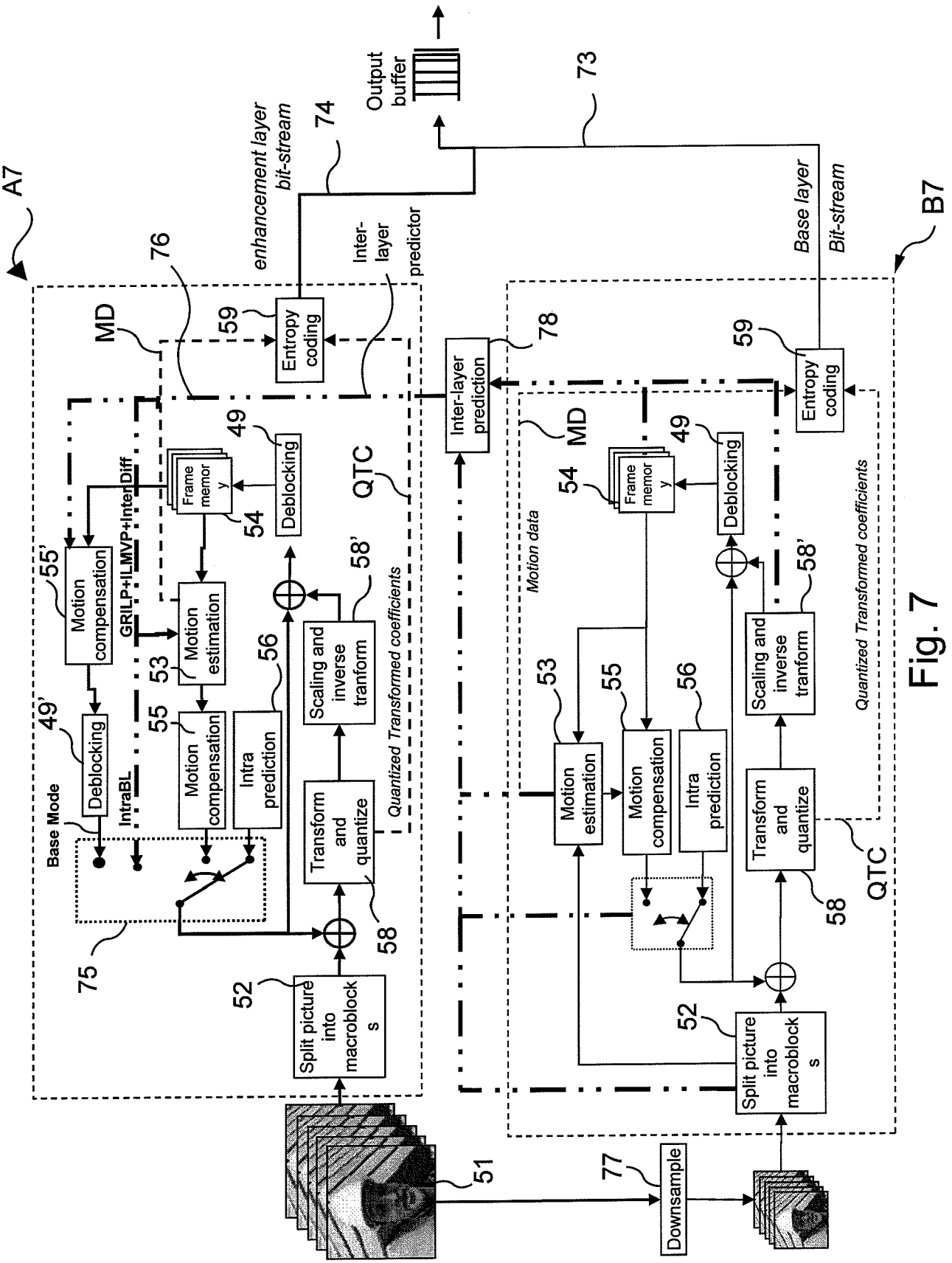

A7
B7
Output buffer
73
enhancement layer bit-stream
74
Base layer Bit-stream
Inter-layer predictor
76
MD
59
Entropy coding
78
Inter-layer prediction
49
Deblocking
Frame memory
54
QTC
Quantized Transformed coefficients
GRILP+ILMVP+InterDiff
55'
Motion compensation
Motion estimation
53
56
Scaling and inverse tranform
58'
49'
Deblocking
IntraBL
55
Motion compensation
Intra prediction
Transform and quantize
58
Base Mode
75
52
Split picture into macroblocks
51
77
Downsample
Motion data
Quantized Transformed coefficients

Fig. 7

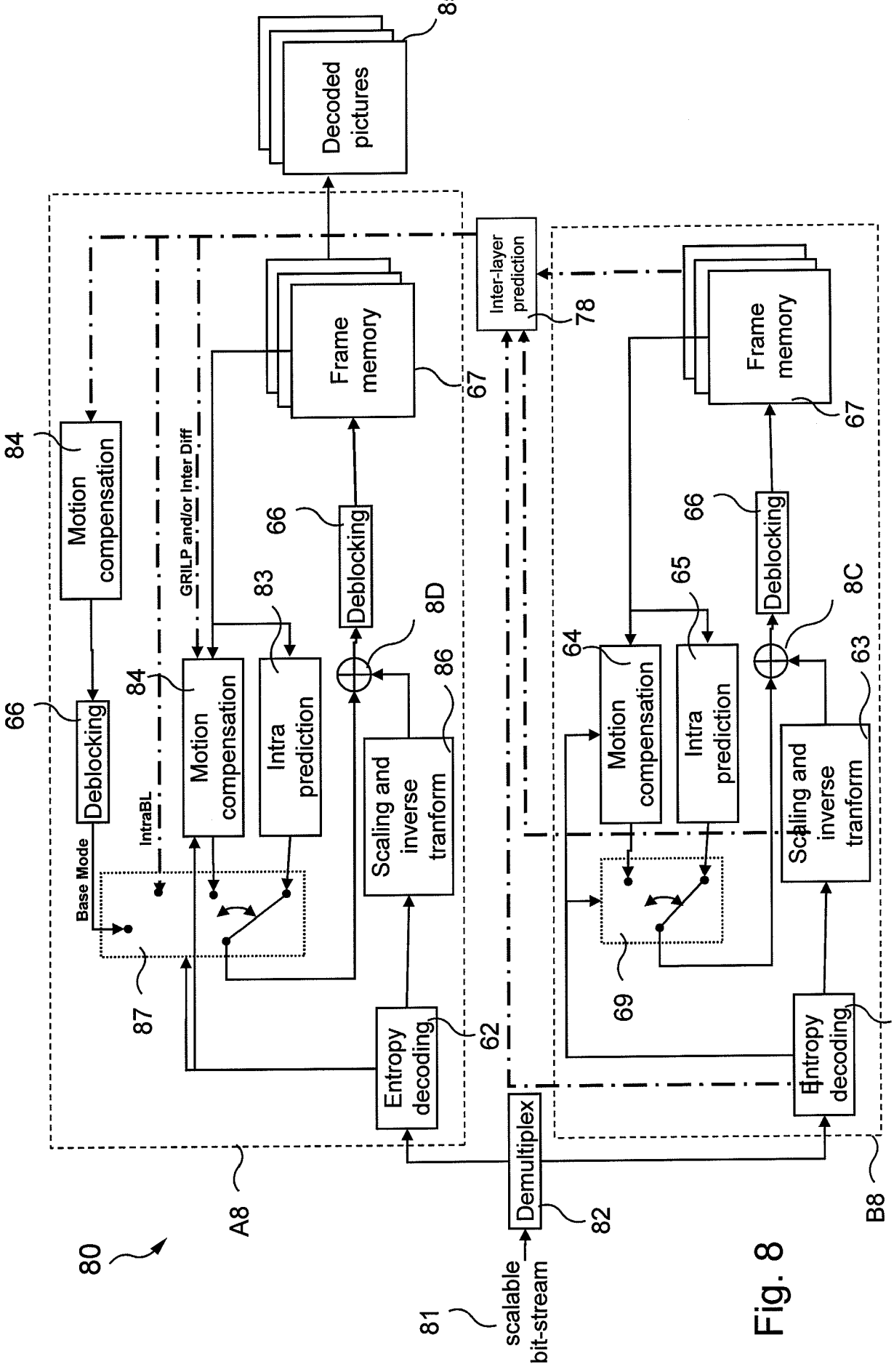

Decoded pictures
85
Frame memory
67
Inter-layer prediction
78
Motion compensation
84
GRILP and/or Inter Diff
Deblocking
66
83
Intra prediction
8D
Scaling and inverse tranform
86
IntraBL
Base Mode
87
Entropy decoding
62
Demultiplex
82
scalable bit-stream
81
80
A8
B8
65
64
63
8C
69

Fig. 8

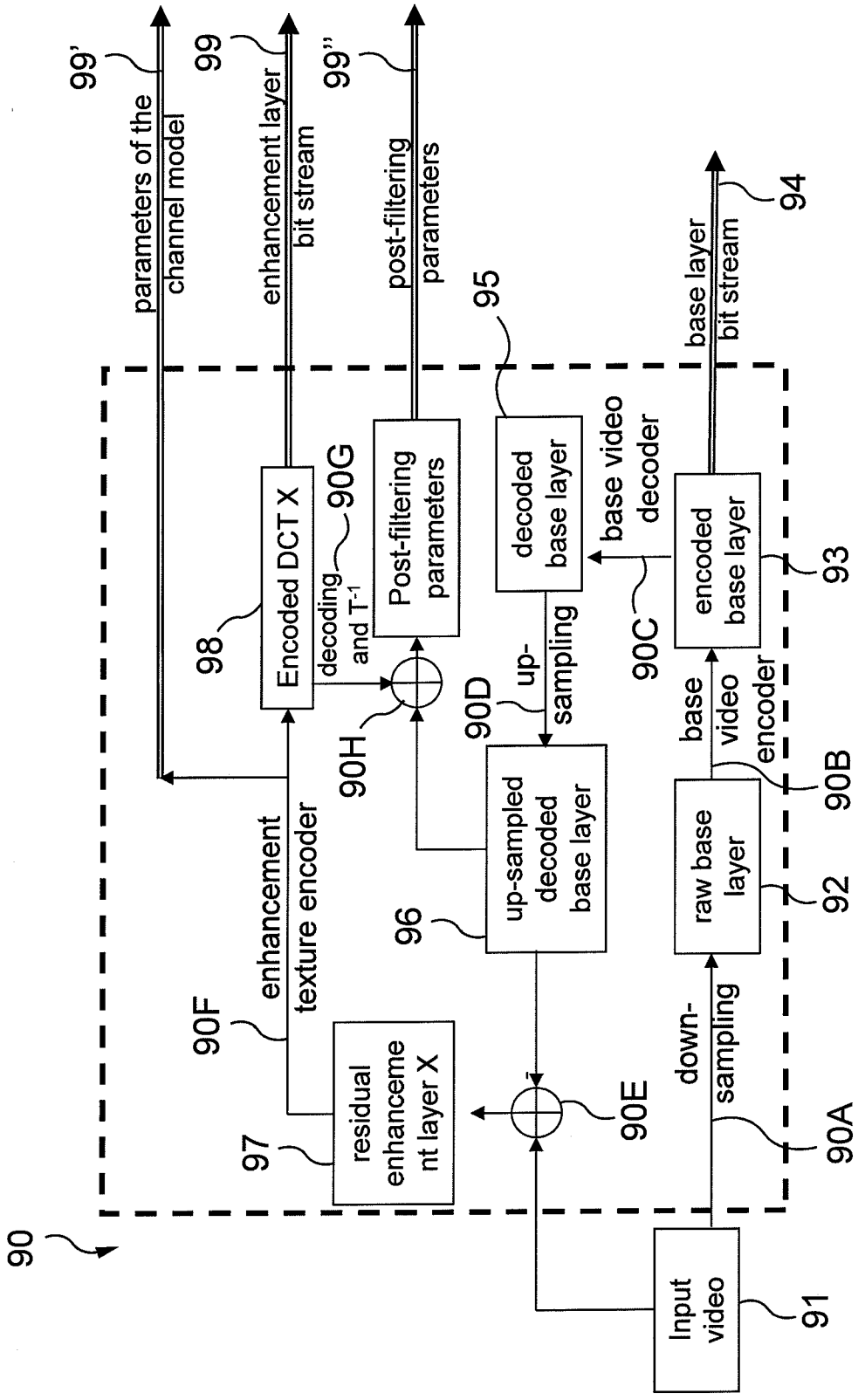

90
Input video
91
down-sampling
90A
raw base layer
92
base video encoder
90B
encoded base layer
93
base layer bit stream
94
base video decoder
90C
decoded base layer
95
up-sampling
90D
up-sampled decoded base layer
96
90E
residual enhancement layer X
97
enhancement texture encoder
90F
Encoded DCT X
98
decoding and T-1
90G
90H
Post-filtering parameters
post-filtering parameters
99''
enhancement layer bit stream
99
parameters of the channel model
99'

Fig. 9

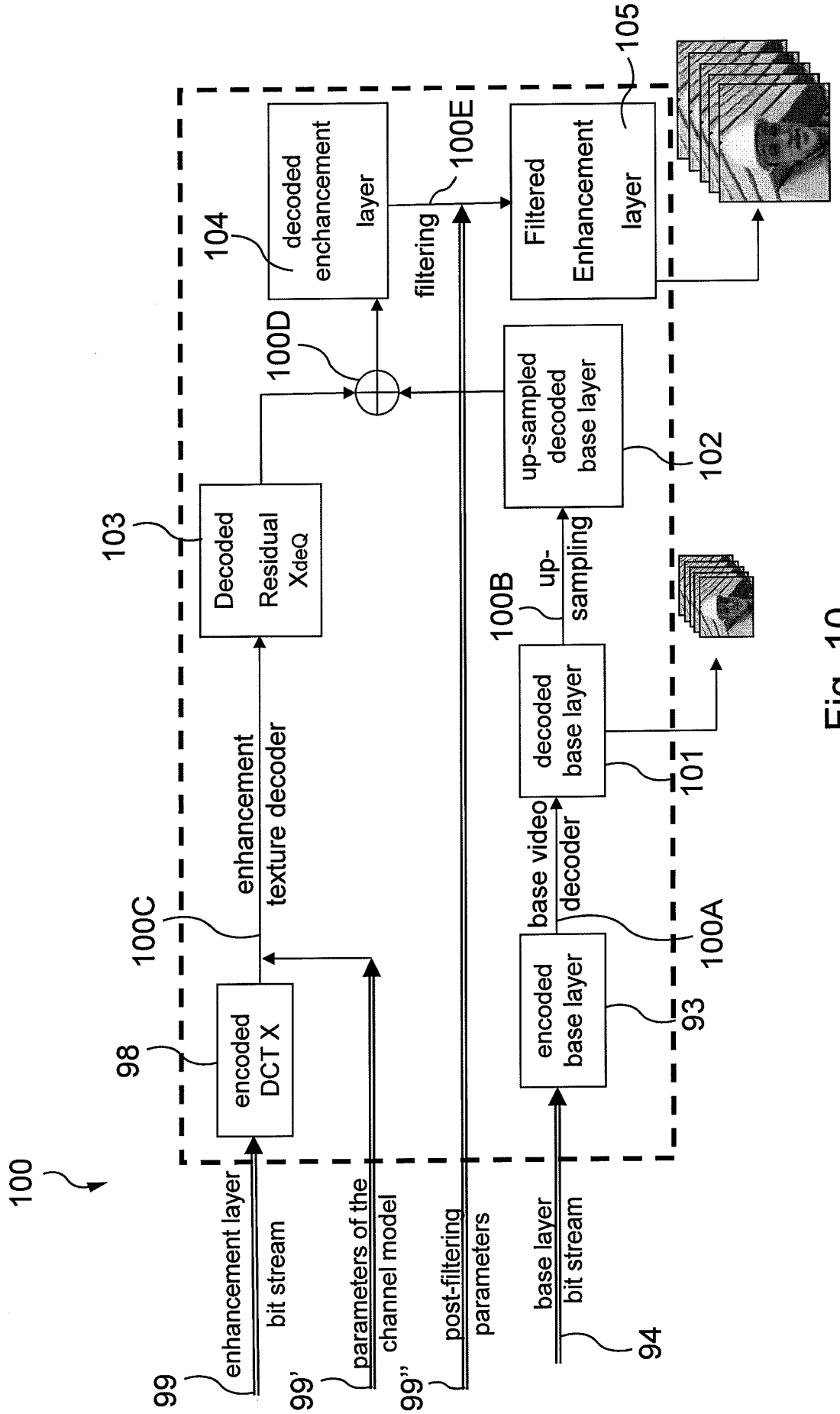

100
enhancement layer bit stream
99
parameters of the channel model
99'
post-filtering parameters
99''
base layer bit stream
94
encoded DCT X
98
Decoded Residual X_{deQ}
103
enhancement texture decoder
100C
encoded base layer
93
base video decoder
100A
decoded base layer
101
up-sampling
100B
up-sampled decoded base layer
102
decoded enchancement layer
104
filtering
100E
Filtered Enhancement layer
105
100D

Fig. 10

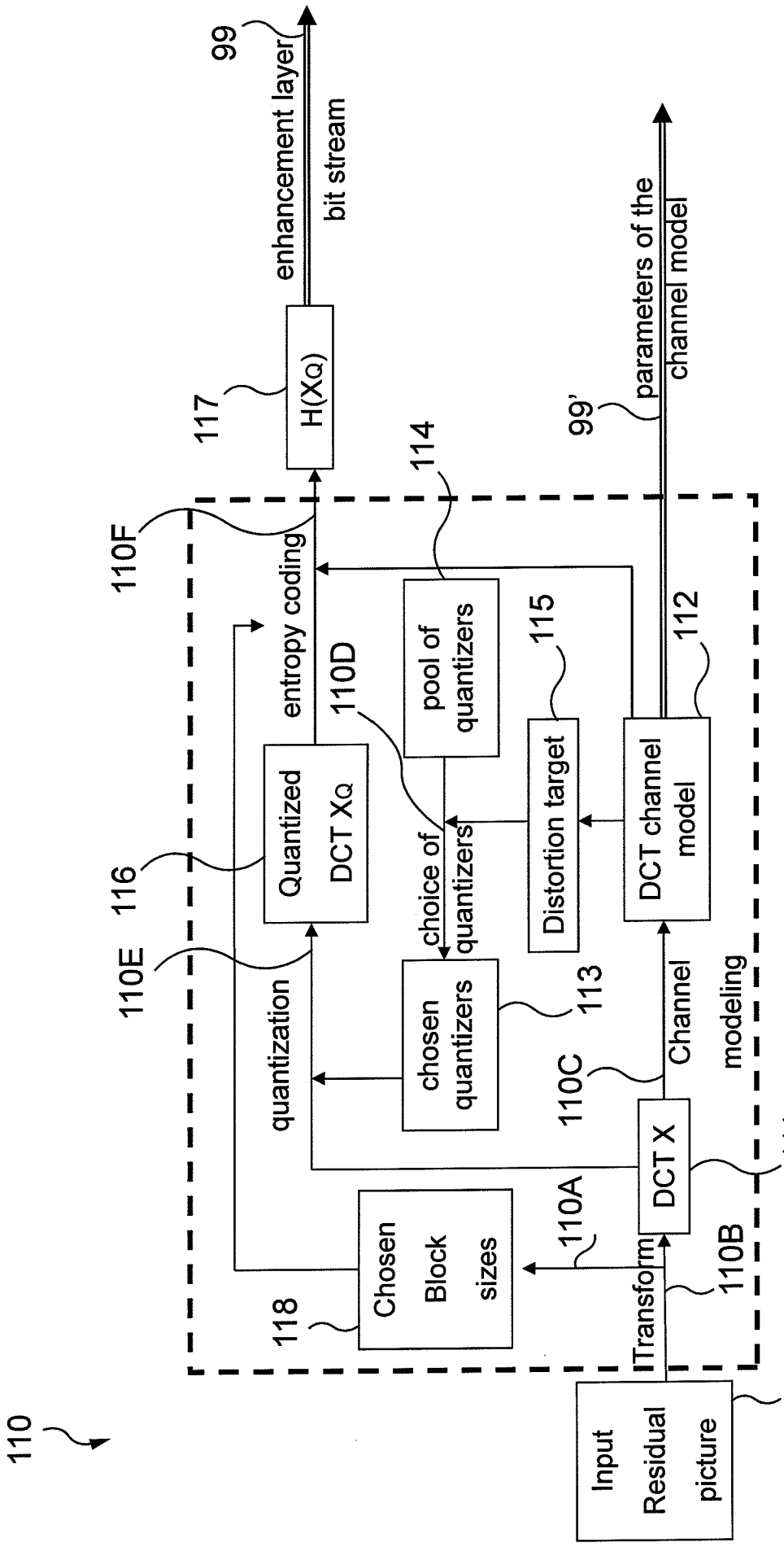

110
Input Residual picture
97
Transform
110B
DCT X
111
110A
Chosen Block sizes
118
quantization
110E
chosen quantizers
113
choice of quantizers
110C
Channel modeling
Quantized DCT XQ
116
entropy coding
110D
pool of quantizers
115
Distortion target
DCT channel model
112
110F
114
H(XQ)
117
99
enhancement layer bit stream
99'
parameters of the channel model

Fig. 11

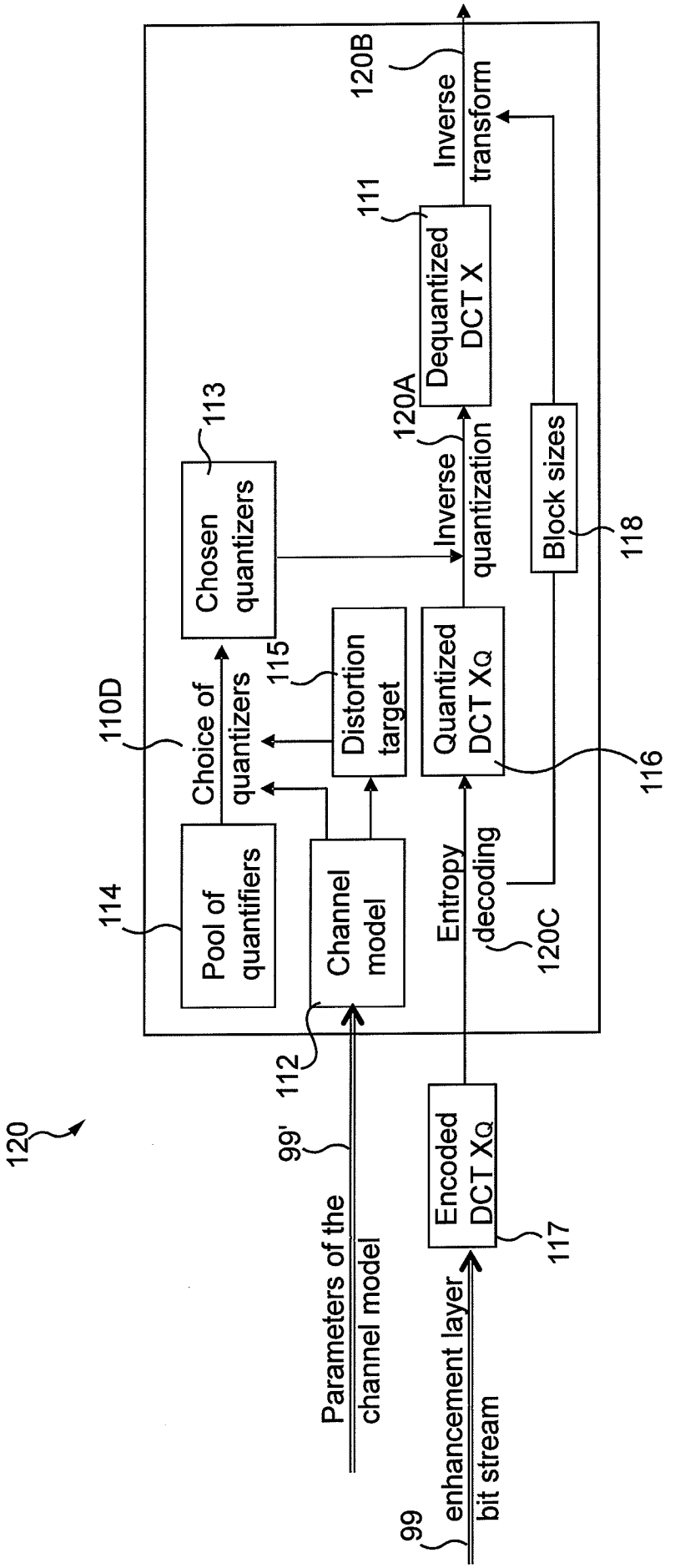

120
99
enhancement layer bit stream
99'
Parameters of the channel model
112
Encoded DCT XQ
117
Pool of quantifiers
114
Channel model
Entropy decoding
120C
Choice of quantizers
110D
Distortion target
115
Chosen quantizers
113
Quantized DCT XQ
116
Inverse quantization
120A
Block sizes
118
Dequantized DCT X
111
Inverse transform
120B

Fig. 12

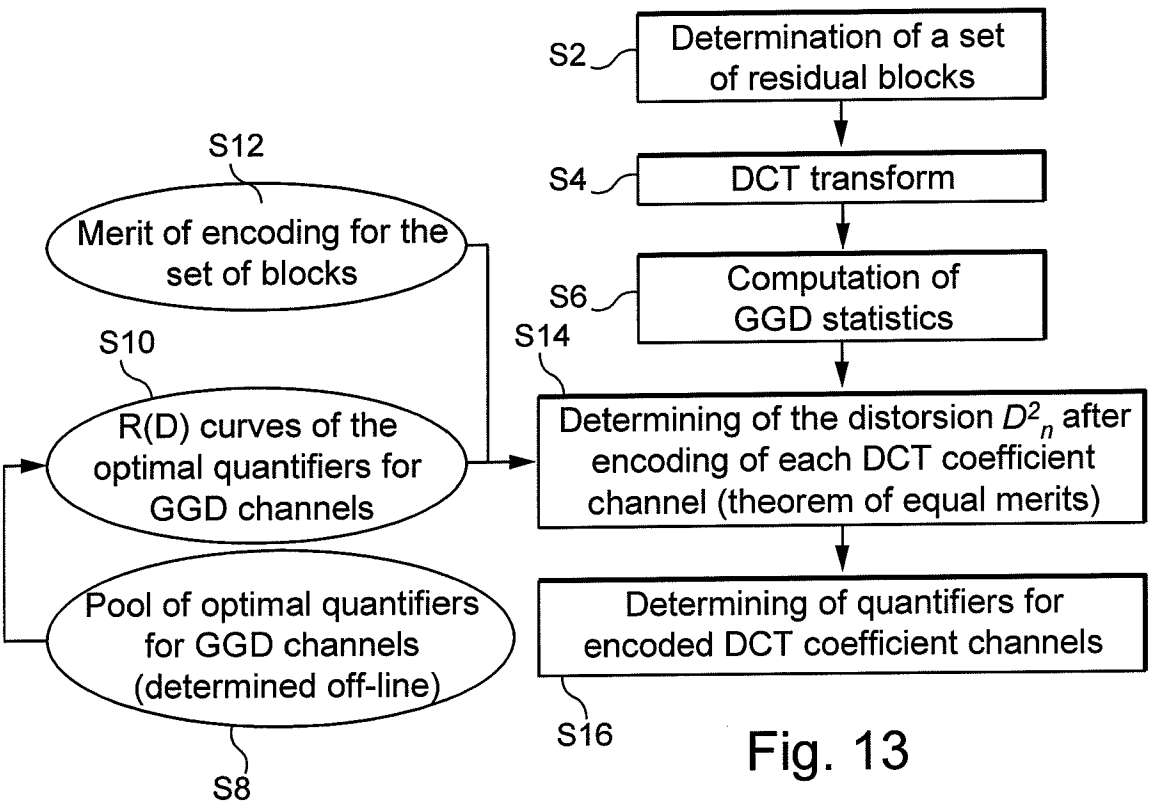

S2
Determination of a set of residual blocks
S4
DCT transform
S6
Computation of GGD statistics
S14
Determining of the distorsion D^2_n after encoding of each DCT coefficient channel (theorem of equal merits)
Determining of quantifiers for encoded DCT coefficient channels
S16
S12
Merit of encoding for the set of blocks
S10
R(D) curves of the optimal quantifiers for GGD channels
Pool of optimal quantifiers for GGD channels (determined off-line)
S8

Fig. 13

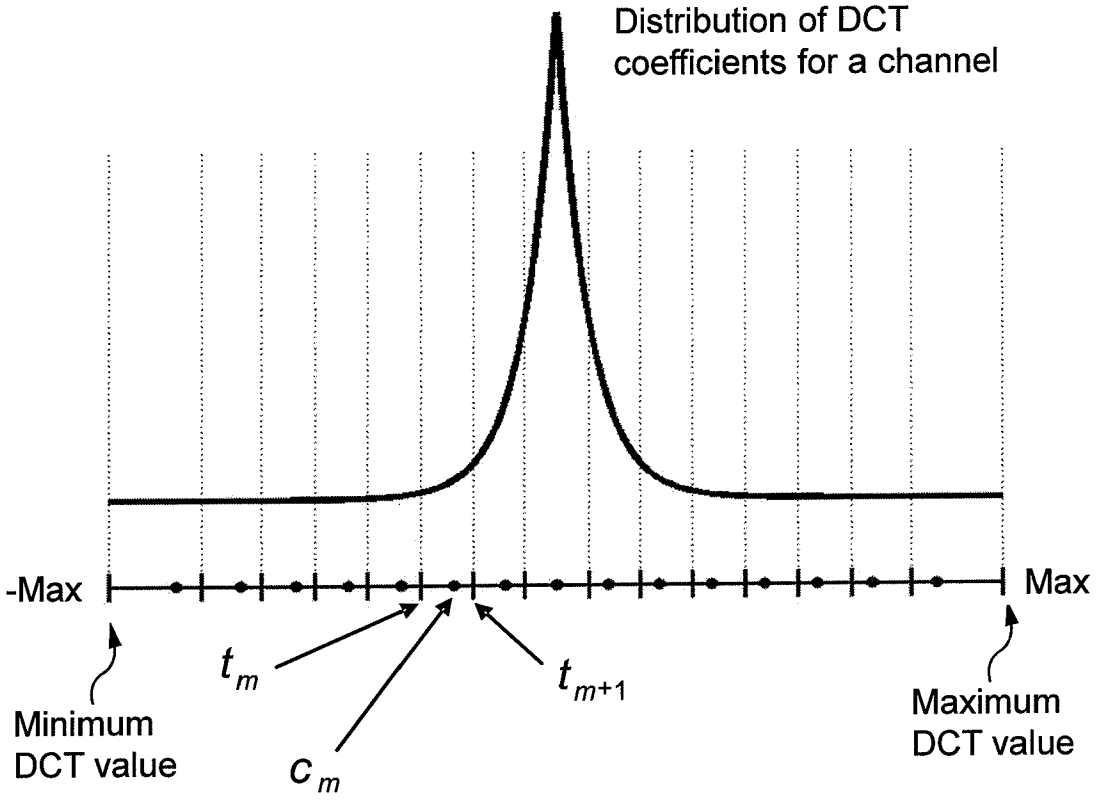

Distribution of DCT coefficients for a channel
-Max
Max
t_m
t_{m+1}
c_m
Minimum DCT value
Maximum DCT value

Fig. 14

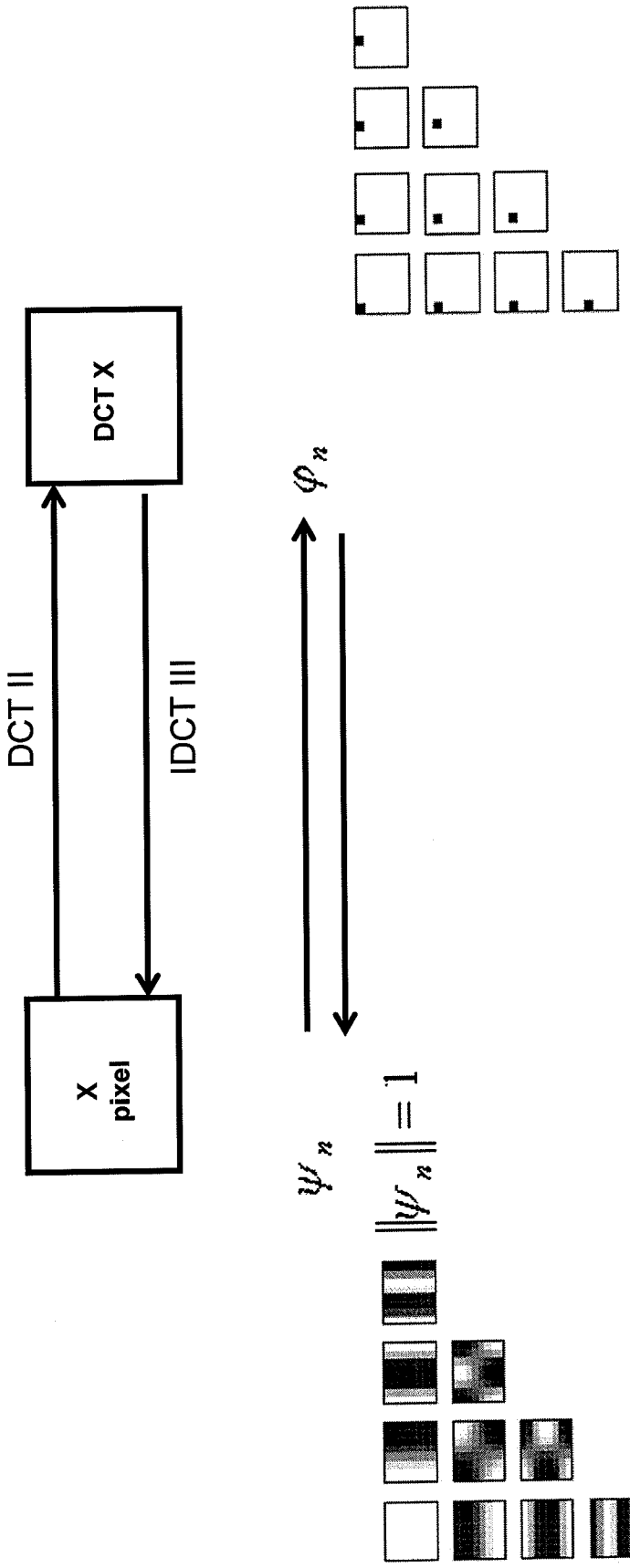


Fig. 15

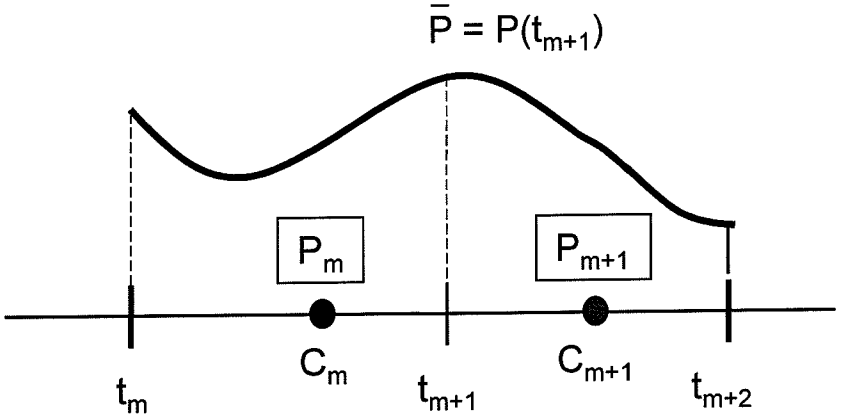


Fig. 16

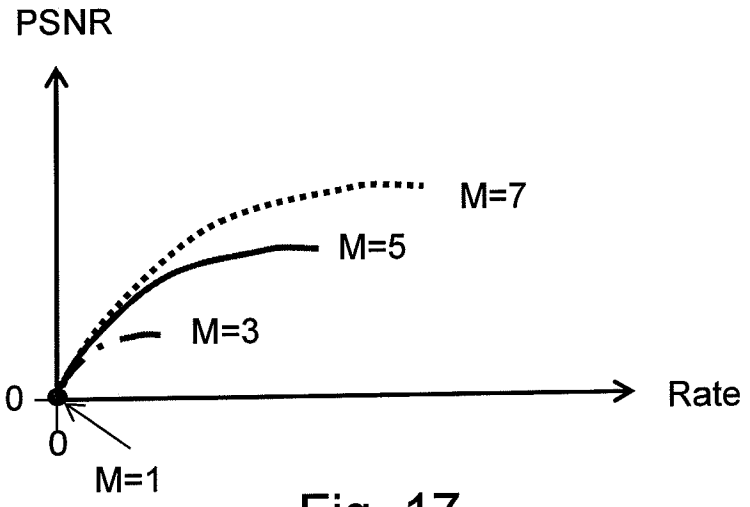


Fig. 17

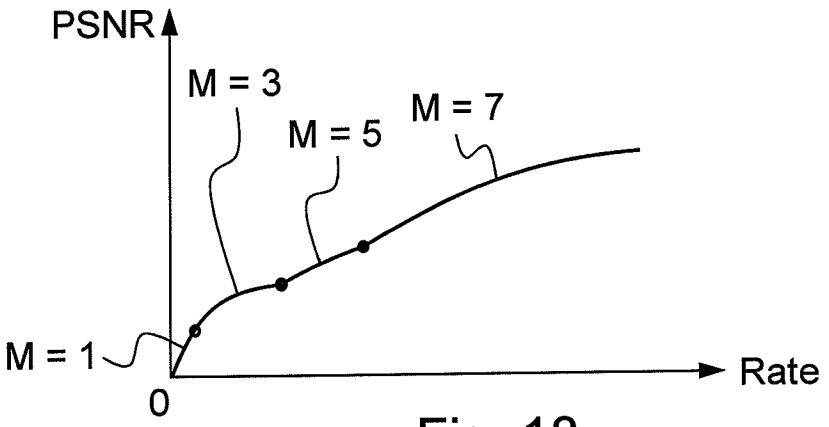


Fig. 18

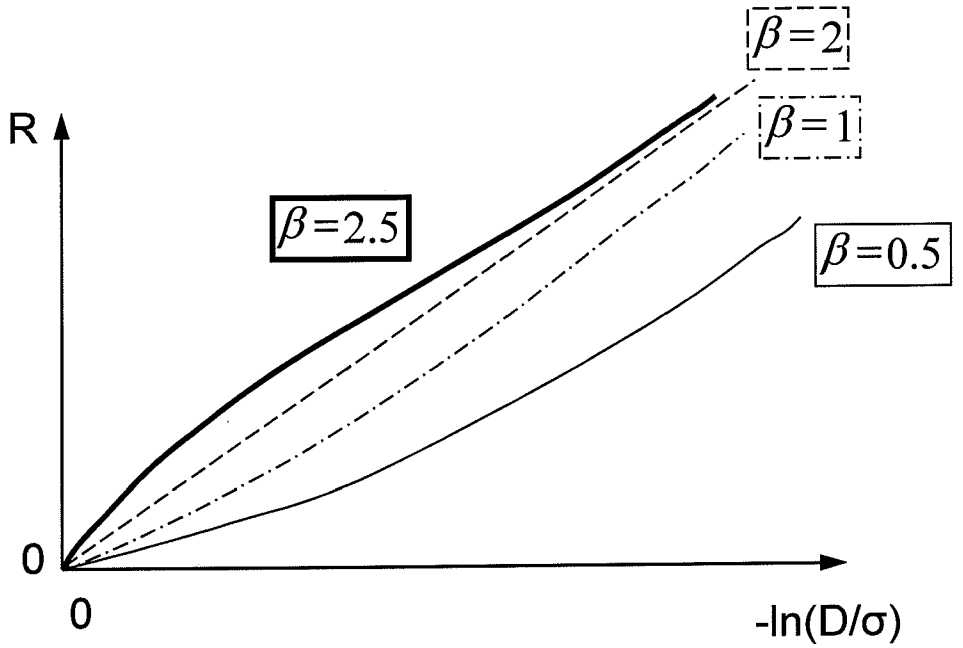

β=2
β=1
β=2.5
β=0.5
R
0
0
-ln(D/σ)

Fig. 19

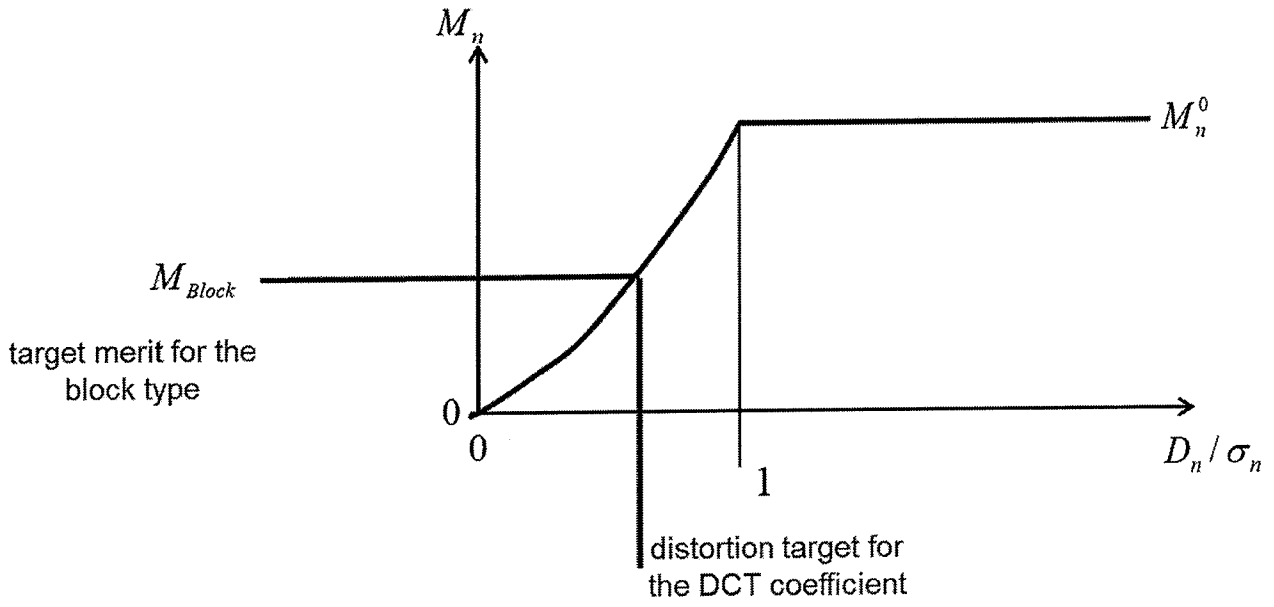

M_n
M_n^0
M_{Block}
target merit for the block type
0
0
1
D_n / σ_n
distortion target for the DCT coefficient

Fig. 20

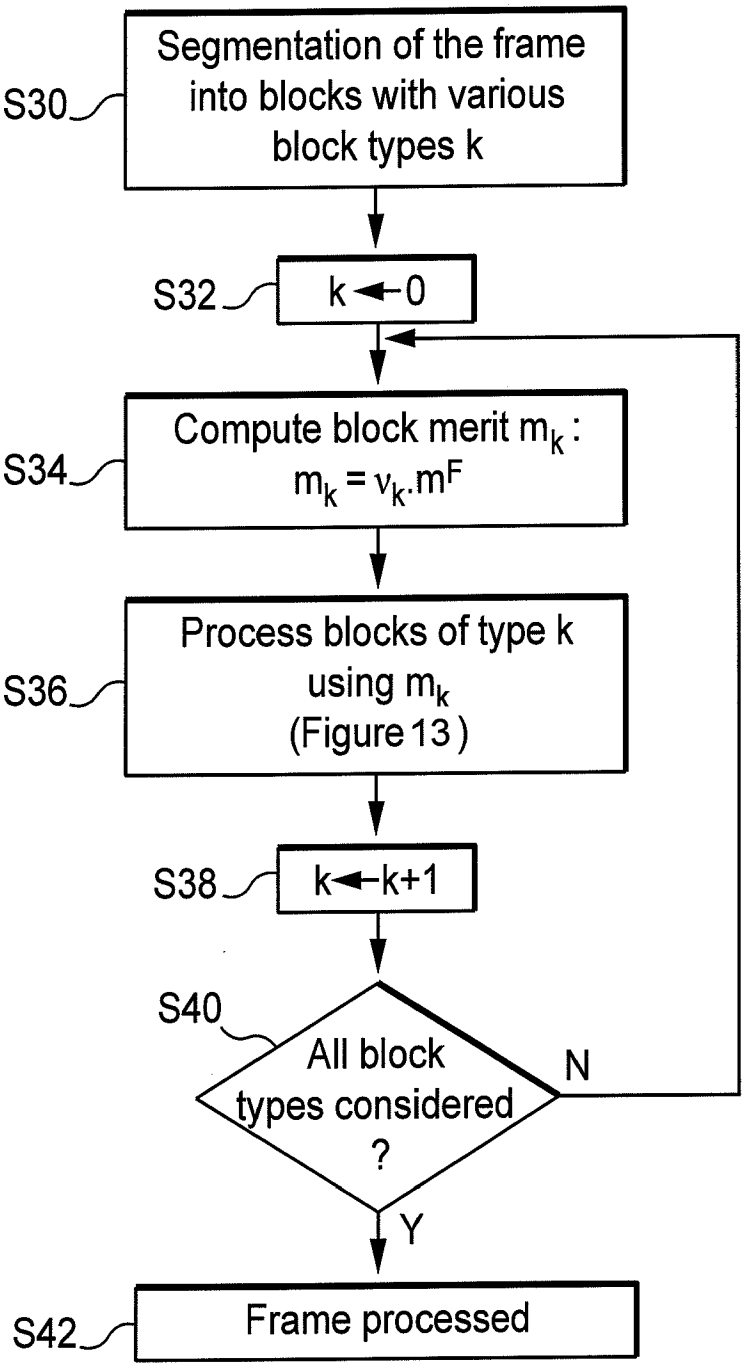

S30
Segmentation of the frame into blocks with various block types k
S32
k←0
S34
Compute block merit m_k : $m_k = \nu_k.m^F$
S36
Process blocks of type k using m_k (Figure 13)
S38
k←k+1
S40
All block types considered ?
N
Y
S42
Frame processed

Fig. 21

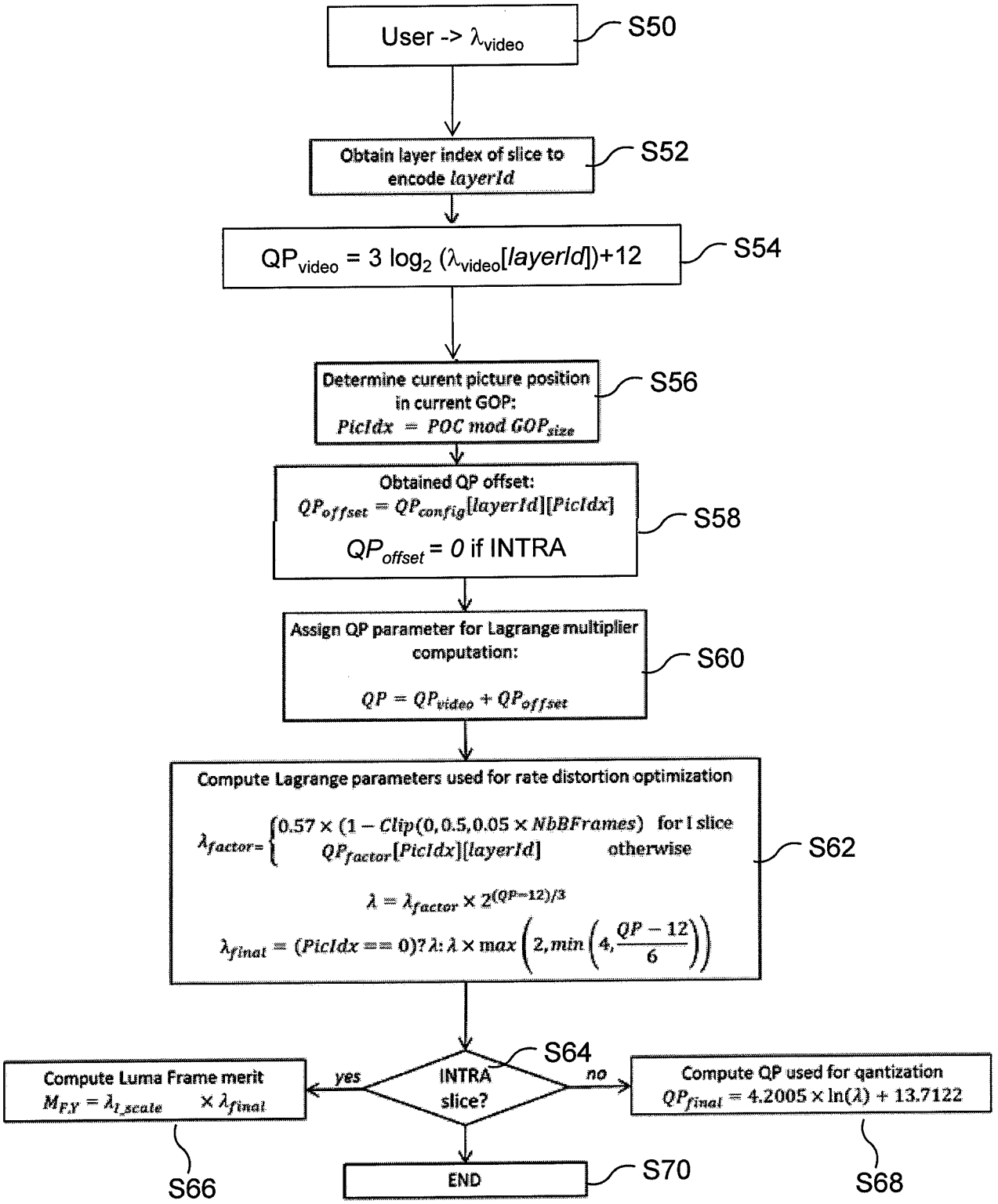

User -> λ_{video}
S50
Obtain layer index of slice to encode *layerId*
S52
$QP_{video} = 3 \log_2 (\lambda_{video}[layerId]) + 12$
S54
Determine curent picture position in current GOP:
$PicIdx = POC \bmod GOP_{size}$
S56
Obtained QP offset:
$QP_{offset} = QP_{config}[layerId][PicIdx]$
$QP_{offset} = 0$ if INTRA
S58
Assign QP parameter for Lagrange multiplier computation:
$QP = QP_{video} + QP_{offset}$
S60
Compute Lagrange parameters used for rate distortion optimization
$\lambda_{factor} = \begin{cases} 0.57 \times (1 - Clip(0, 0.5, 0.05 \times NbBFrames)) & \text{for I slice} \\ QP_{factor}[PicIdx][layerId] & \text{otherwise} \end{cases}$
$\lambda = \lambda_{factor} \times 2^{(QP-12)/3}$
$\lambda_{final} = (PicIdx == 0)\,?\,\lambda : \lambda \times max\left(2, min\left(4, \frac{QP-12}{6}\right)\right)$
S62
INTRA slice?
S64
yes
no
Compute Luma Frame merit
$M_{F,Y} = \lambda_{I_scale} \times \lambda_{final}$
S66
Compute QP used for qantization
$QP_{final} = 4.2005 \times \ln(\lambda) + 13.7122$
S68
END
S70

Fig. 21A

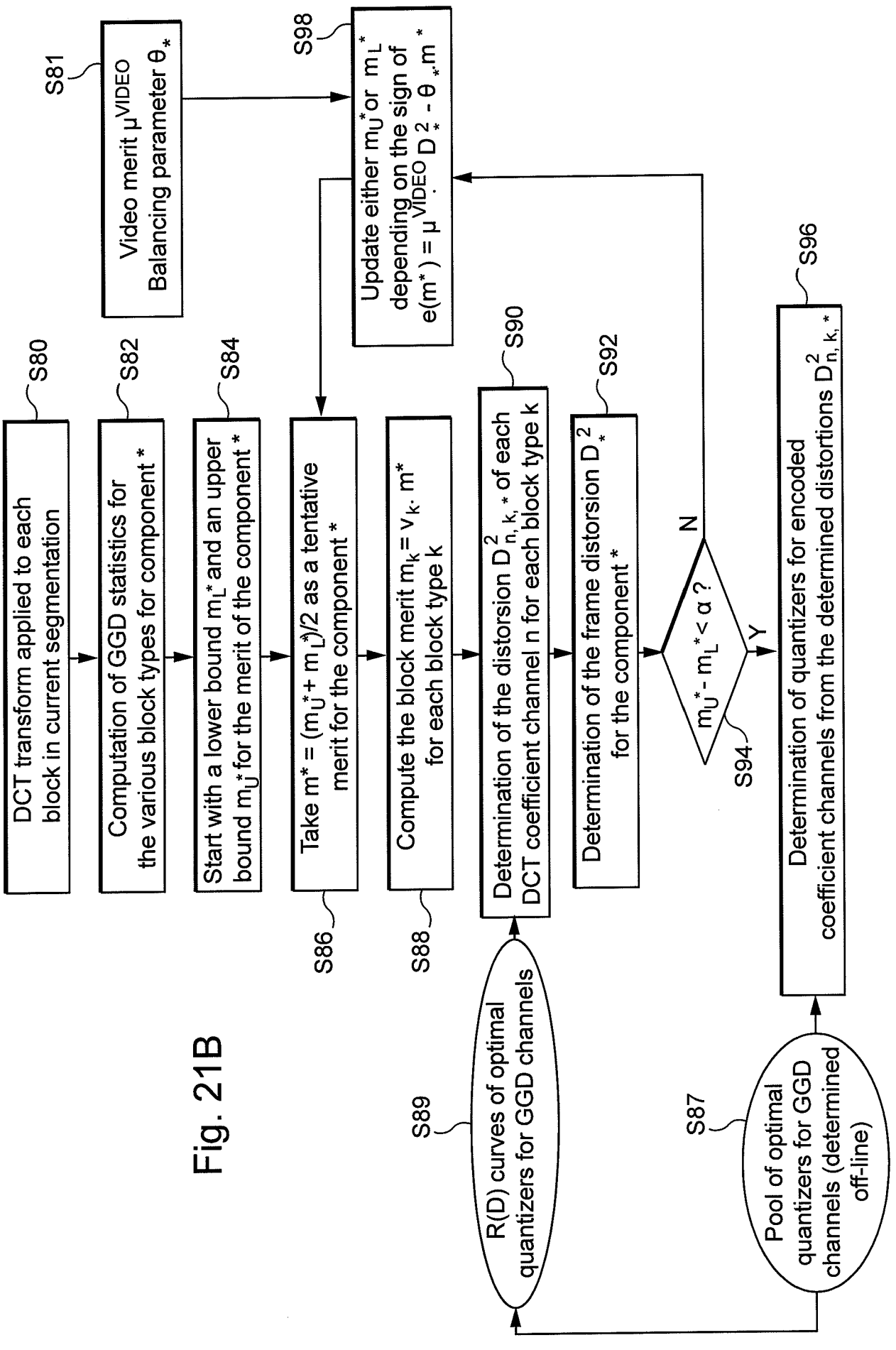

DCT transform applied to each block in current segmentation
S80
Computation of GGD statistics for the various block types for component *
S82
Start with a lower bound m_L^* and an upper bound m_U^* for the merit of the component *
S84
Take $m^* = (m_U^* + m_L^*)/2$ as a tentative merit for the component *
S86
Compute the block merit $m_k = v_k \cdot m^*$ for each block type k
S88
Determination of the distorsion $D^2_{n,k,*}$ of each DCT coefficient channel n for each block type k
S90
Determination of the frame distorsion D^2_* for the component *
S92
$m_U^* - m_L^* < \alpha$?
S94
N
Y
Determination of quantizers for encoded coefficient channels from the determined distortions $D^2_{n,k,*}$
S96
Update either m_U^* or m_L^* depending on the sign of $e(m^*) = \mu^{VIDEO} \cdot D^2_* - \theta_* \cdot m^*$
S98
Video merit μ^{VIDEO} Balancing parameter θ_*
S81
R(D) curves of optimal quantizers for GGD channels
S89
Pool of optimal quantizers for GGD channels (determined off-line)
S87

Fig. 21B

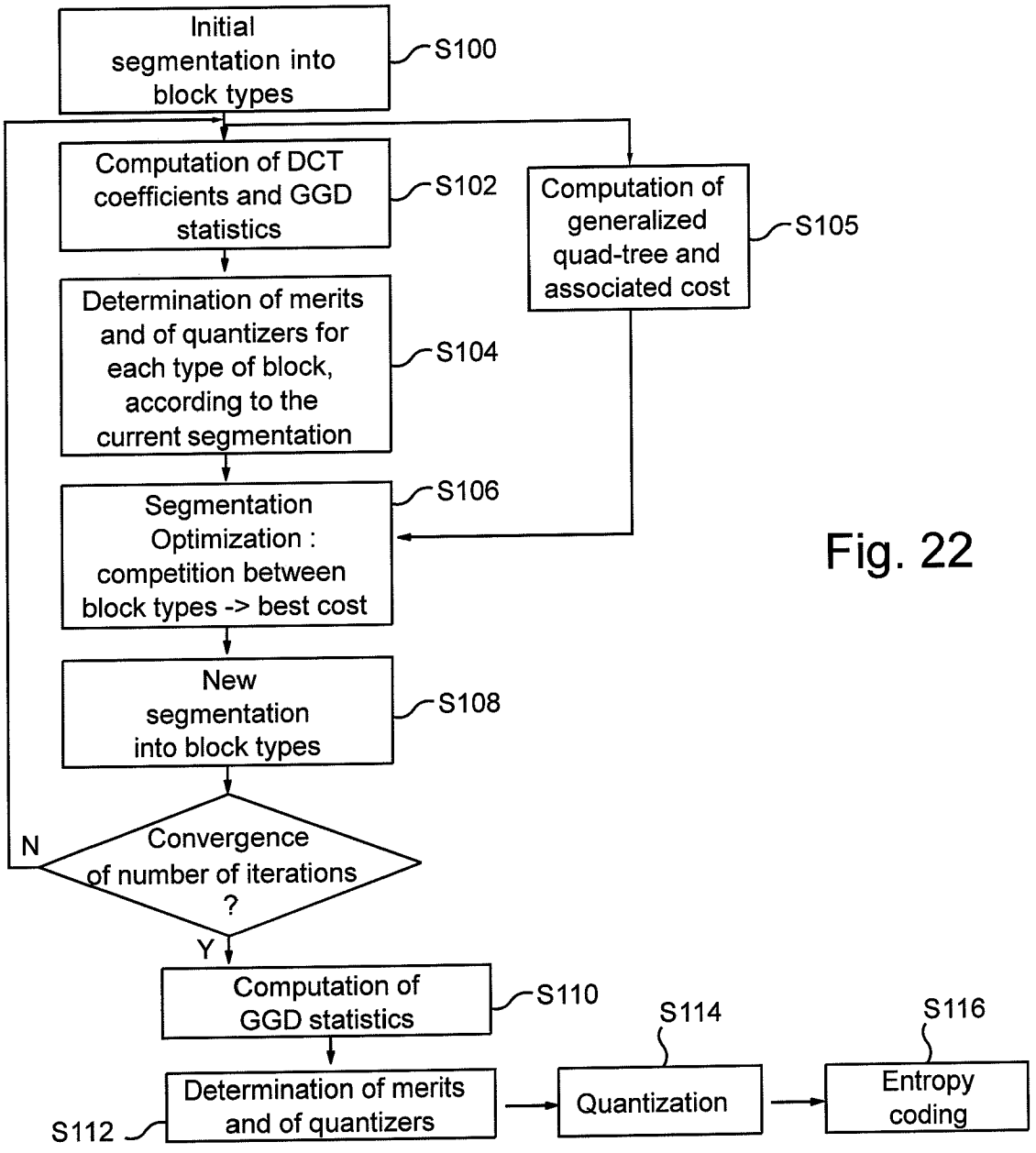

Initial segmentation into block types
S100
Computation of DCT coefficients and GGD statistics
S102
Computation of generalized quad-tree and associated cost
S105
Determination of merits and of quantizers for each type of block, according to the current segmentation
S104
Segmentation Optimization : competition between block types -> best cost
S106
New segmentation into block types
S108
Convergence of number of iterations ?
N
Y
Computation of GGD statistics
S110
Determination of merits and of quantizers
S112
S114
Quantization
S116
Entropy coding

Fig. 22

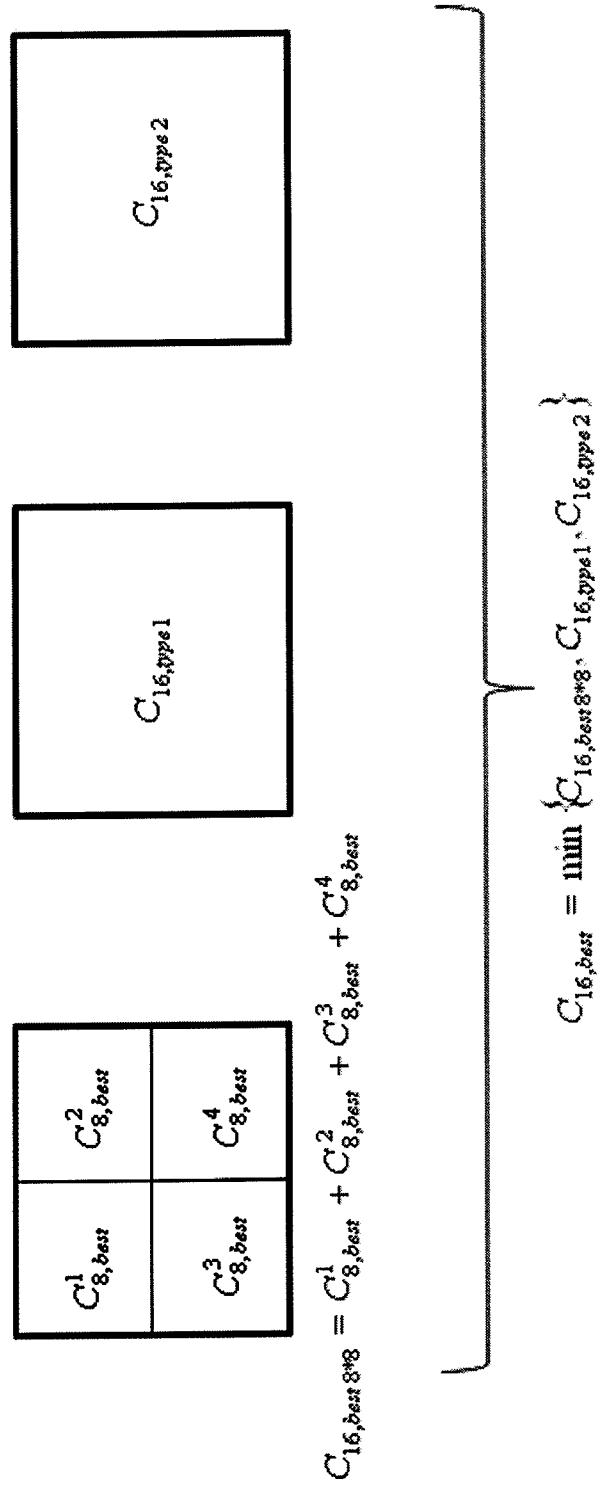

$C_{16,type2}$
$C_{16,type1}$
$C^1_{8,best}$ $C^2_{8,best}$ $C^3_{8,best}$ $C^4_{8,best}$
$C_{16,best8*8} = C^1_{8,best} + C^2_{8,best} + C^3_{8,best} + C^4_{8,best}$
$C_{16,best} = \min\{C_{16,best8*8}, C_{16,type1}, C_{16,type2}\}$

Fig. 23

Fig. 24

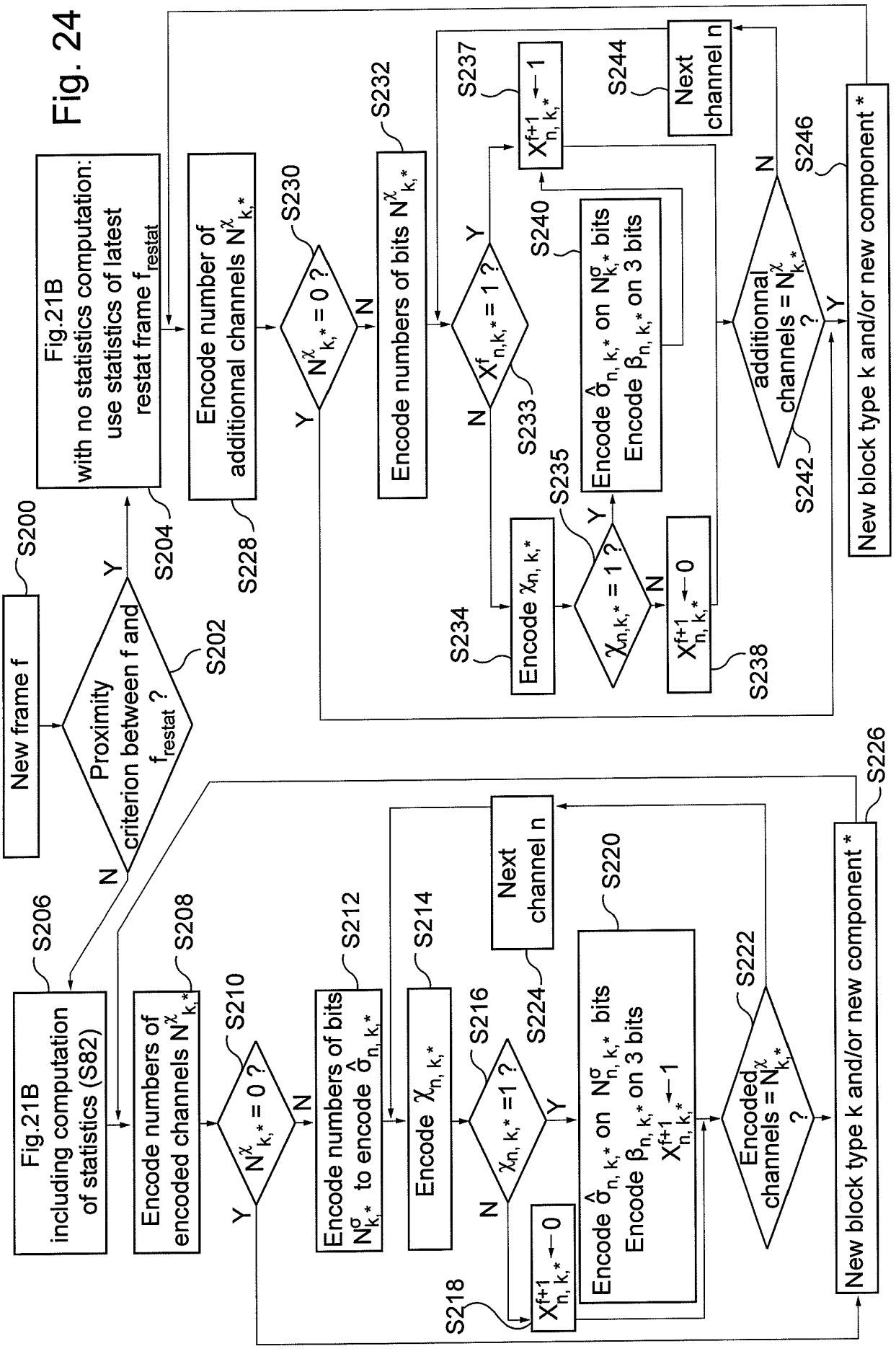
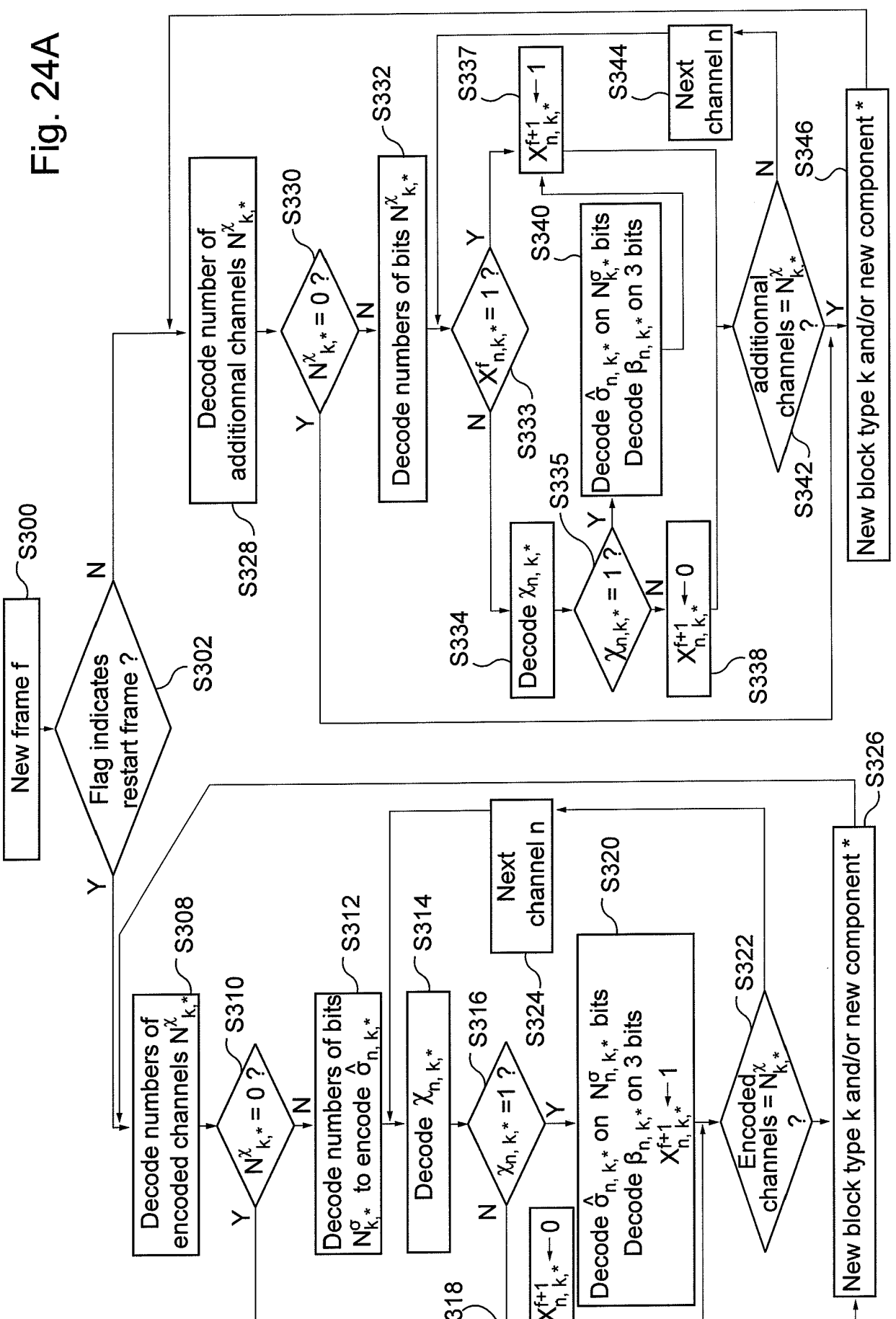

New frame f
S200
Proximity criterion between f and f_{restat} ?
S202
Y
N
S204
Fig.21B with no statistics computation: use statistics of latest restat frame f_{restat}
Encode number of additionnal channels $N^{\chi}_{k,*}$
S228
S230
$N^{\chi}_{k,*} = 0$?
Y
N
S232
Encode numbers of bits $N^{\chi}_{k,*}$
$X^{f}_{n,k,*} = 1$?
Y
N
S233
S237
$X^{f+1}_{n,k,*} \leftarrow 1$
S240
Encode $\hat{\sigma}_{n,k,*}$ on $N^{\sigma}_{k,*}$ bits
Encode $\beta_{n,k,*}$ on 3 bits
S244
Next channel n
S246
additionnal channels = $N^{\chi}_{k,*}$?
N
Y
S242
New block type k and/or new component *
S234
Encode $\chi_{n,k,*}$
S235
$\chi_{n,k,*} = 1$?
Y
N
$X^{f+1}_{n,k,*} \leftarrow 0$
S238
Fig.21B including computation of statistics (S82)
S206
Encode numbers of encoded channels $N^{\chi}_{k,*}$
S208
$N^{\chi}_{k,*} = 0$?
Y
N
S210
Encode numbers of bits $N^{\sigma}_{k,*}$ to encode $\hat{\sigma}_{n,k,*}$
S212
Encode $\chi_{n,k,*}$
S214
$\chi_{n,k,*} = 1$?
N
Y
S216
$X^{f+1}_{n,k,*} \leftarrow 0$
S218
Next channel n
S224
Encode $\hat{\sigma}_{n,k,*}$ on $N^{\sigma}_{n,k,*}$ bits
Encode $\beta_{n,k,*}$ on 3 bits
$X^{f+1}_{n,k,*} \leftarrow 1$
S220
Encoded channels = $N^{\chi}_{k,*}$?
S222
New block type k and/or new component *
S226

Fig. 24A


New frame f
S300
Flag indicates restart frame ?
S302
N
Y
Decode numbers of encoded channels $N^{\chi}_{k,*}$
S308
$N^{\chi}_{k,*}$ = 0 ?
S310
Y
N
Decode numbers of bits $N^{\sigma}_{k,*}$ to encode $\hat{\sigma}_{n,k,*}$
S312
Decode $\chi_{n,k,*}$
S314
$\chi_{n,k,*}$ =1 ?
S316
N
Y
$X^{f+1}_{n,k,*} \leftarrow 0$
S318
Next channel n
S324
Decode $\hat{\sigma}_{n,k,*}$ on $N^{\sigma}_{n,k,*}$ bits
Decode $\beta_{n,k,*}$ on 3 bits
$X^{f+1}_{n,k,*} \leftarrow 1$
S320
Encoded χ channels = $N^{\chi}_{k,*}$?
S322
New block type k and/or new component *
S326
Decode number of additionnal channels $N^{\chi}_{k,*}$
S328
$N^{\chi}_{k,*}$ = 0 ?
S330
Y
N
Decode numbers of bits $N^{\chi}_{k,*}$
S332
$X^{f}_{n,k,*}$ = 1 ?
S333
Y
N
$X^{f+1}_{n,k,*} \leftarrow 1$
S337
Next channel n
S344
Decode $\chi_{n,k,*}$
S334
$\chi_{n,k,*}$ = 1 ?
S335
Y
N
Decode $\hat{\sigma}_{n,k,*}$ on $N^{\sigma}_{k,*}$ bits
Decode $\beta_{n,k,*}$ on 3 bits
S340
$X^{f+1}_{n,k,*} \leftarrow 0$
S338
additionnal channels = $N^{\chi}_{k,*}$?
S342
N
Y
S346
New block type k and/or new component *

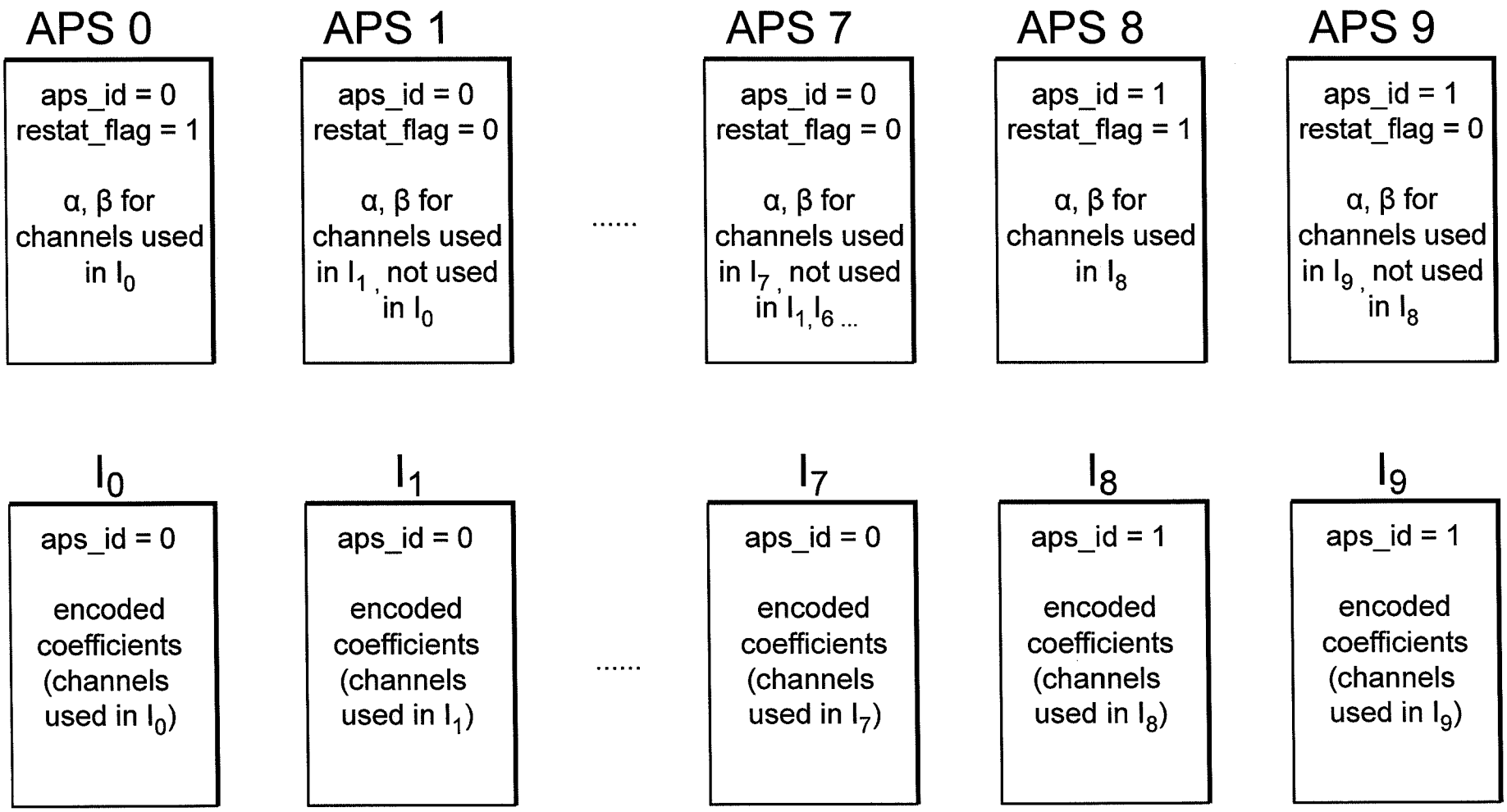

APS 0
aps_id = 0
restat_flag = 1
α, β for channels used in I_0
APS 1
aps_id = 0
restat_flag = 0
α, β for channels used in I_1, not used in I_0
.....
APS 7
aps_id = 0
restat_flag = 0
α, β for channels used in I_7, not used in I_1, I_6 ...
APS 8
aps_id = 1
restat_flag = 1
α, β for channels used in I_8
APS 9
aps_id = 1
restat_flag = 0
α, β for channels used in I_9, not used in I_8
I_0
aps_id = 0
encoded coefficients (channels used in I_0)
I_1
aps_id = 0
encoded coefficients (channels used in I_1)
.....
I_7
aps_id = 0
encoded coefficients (channels used in I_7)
I_8
aps_id = 1
encoded coefficients (channels used in I_8)
I_9
aps_id = 1
encoded coefficients (channels used in I_9)

Fig. 24B

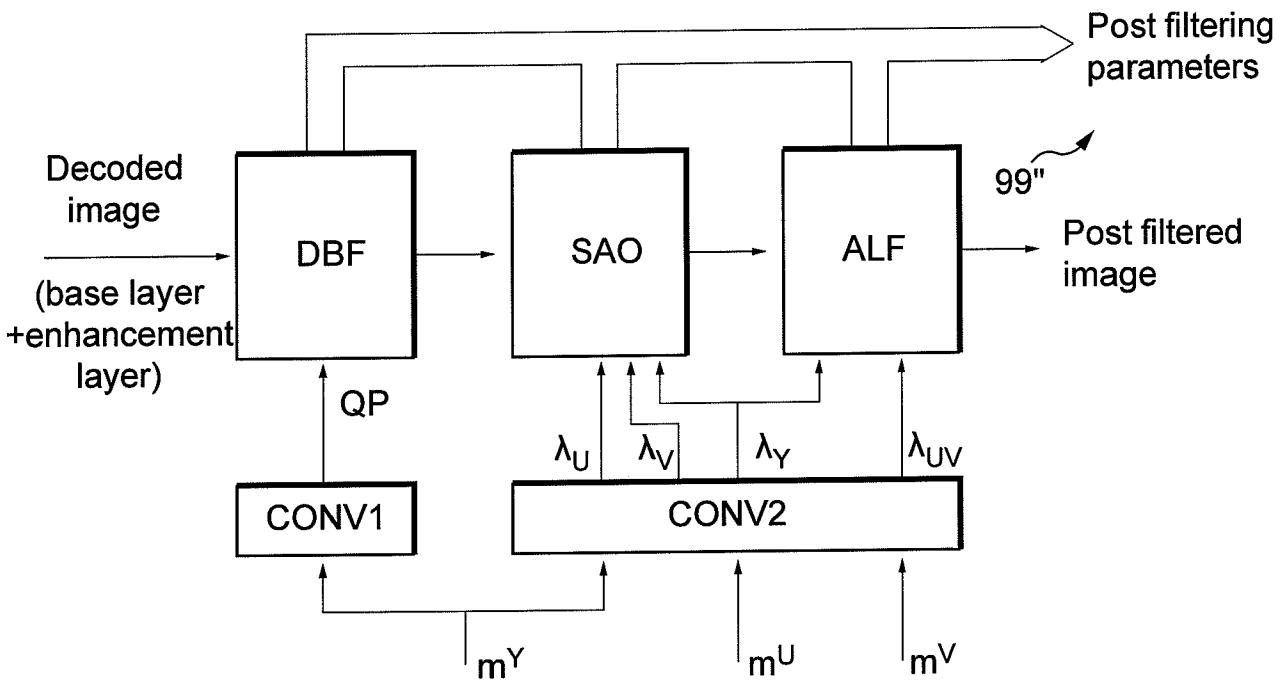


Fig. 25

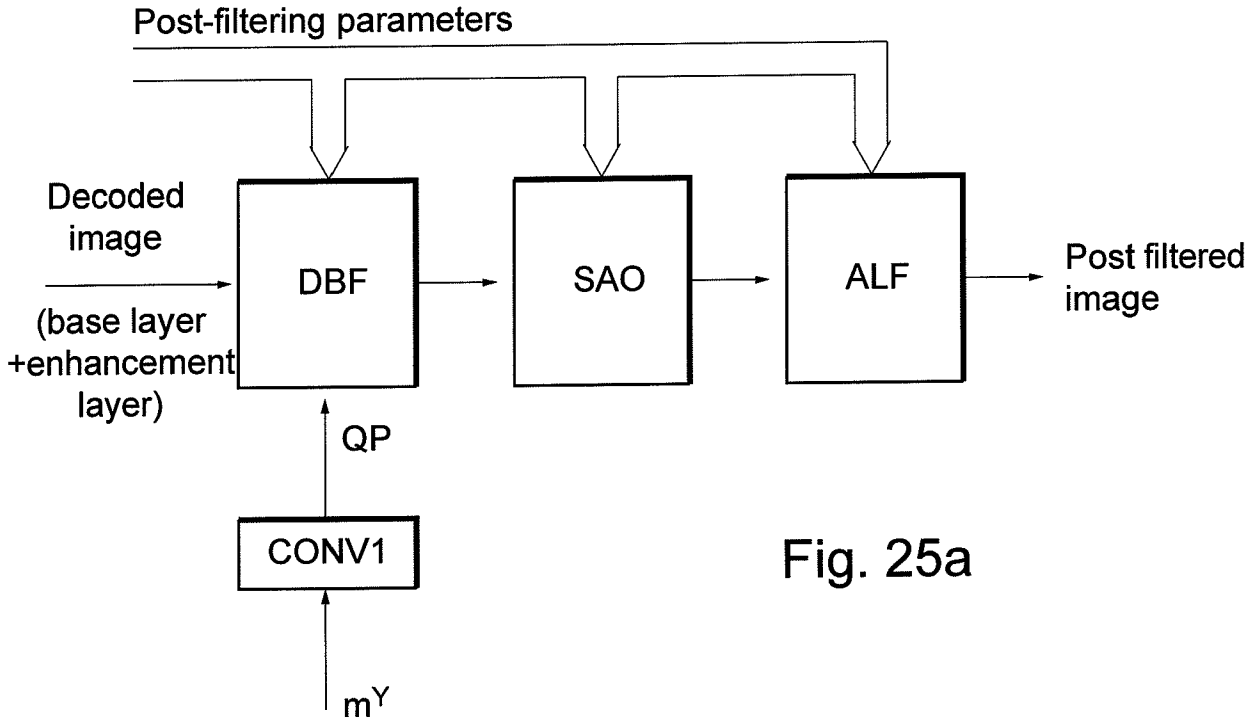


Fig. 25a

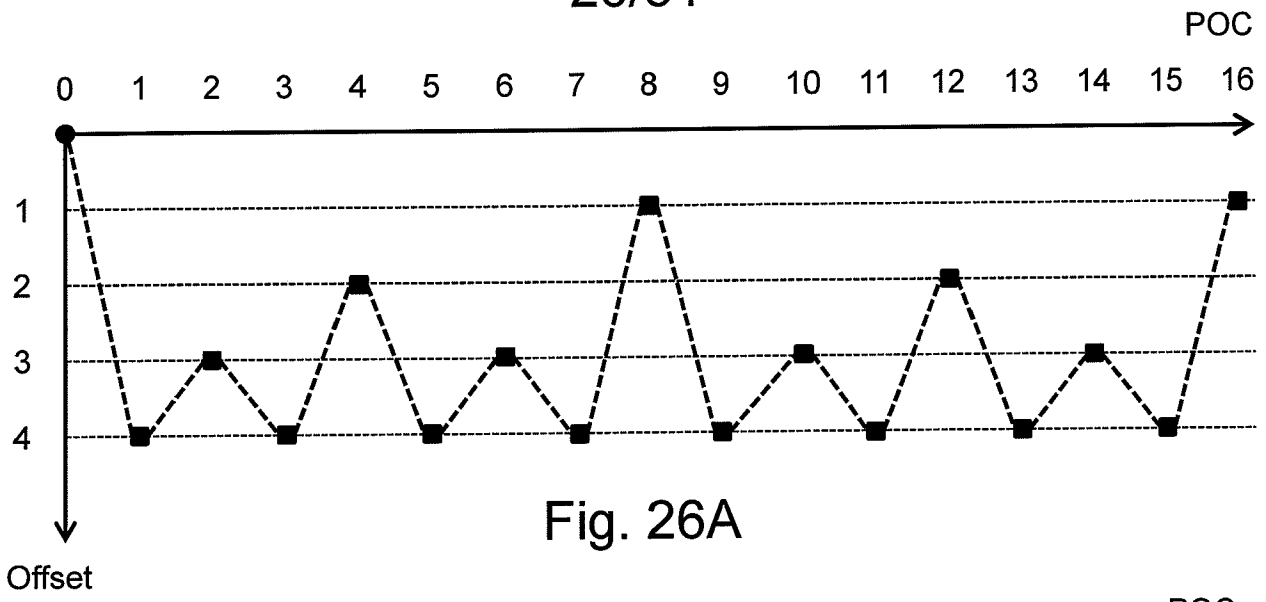


Fig. 26A

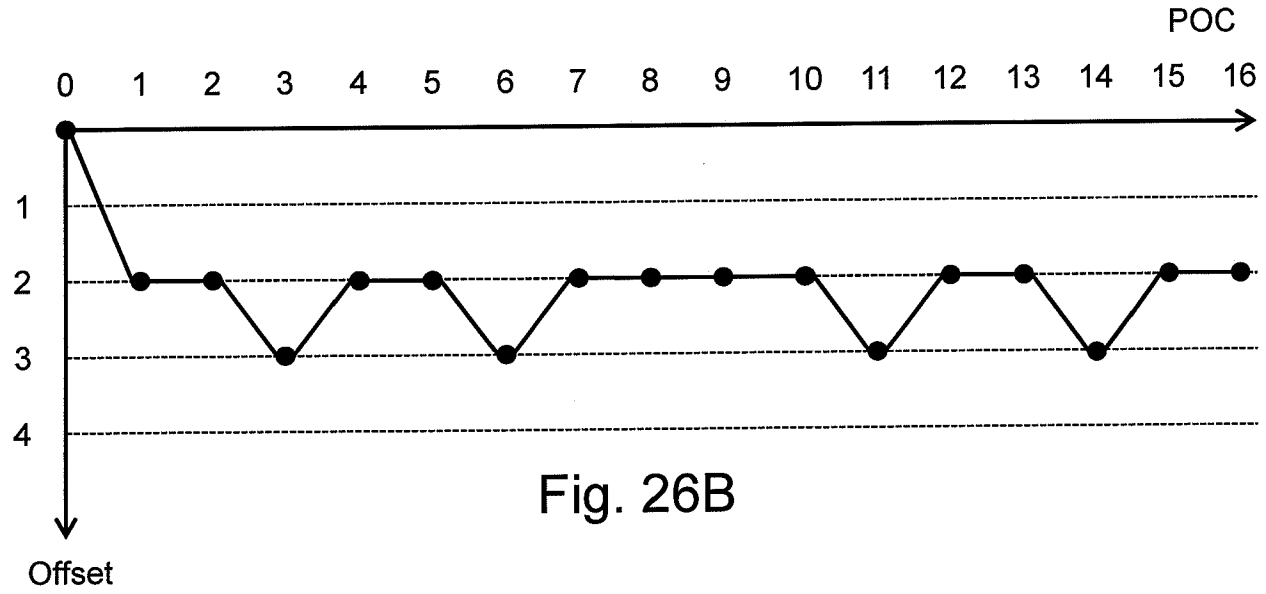


Fig. 26B

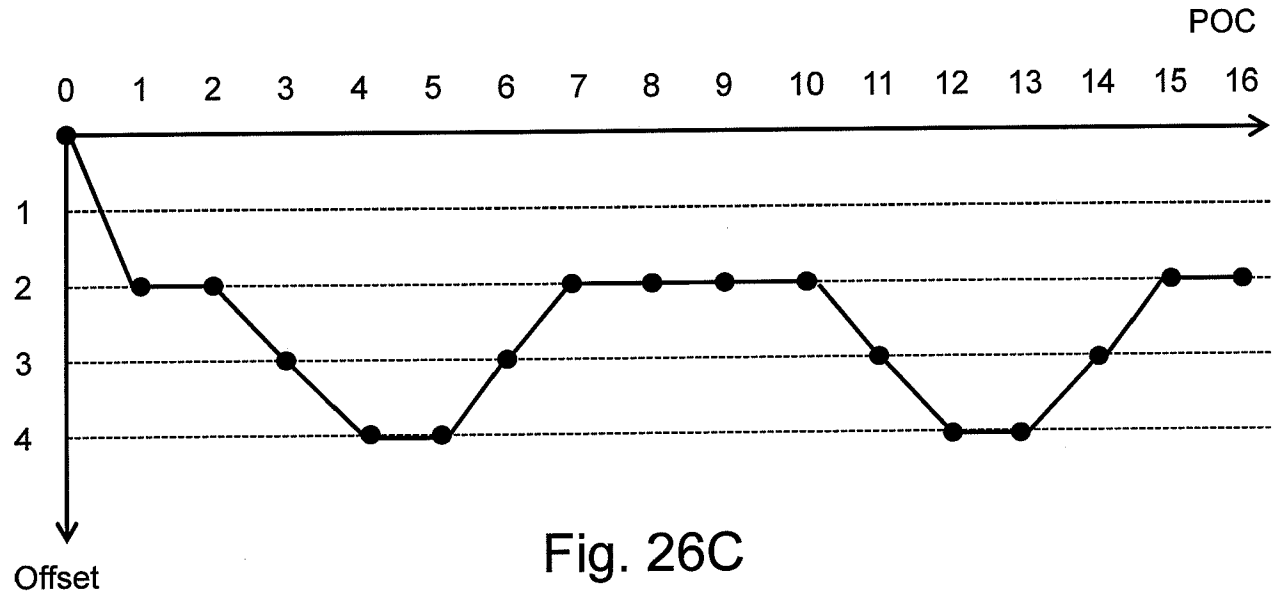


Fig. 26C

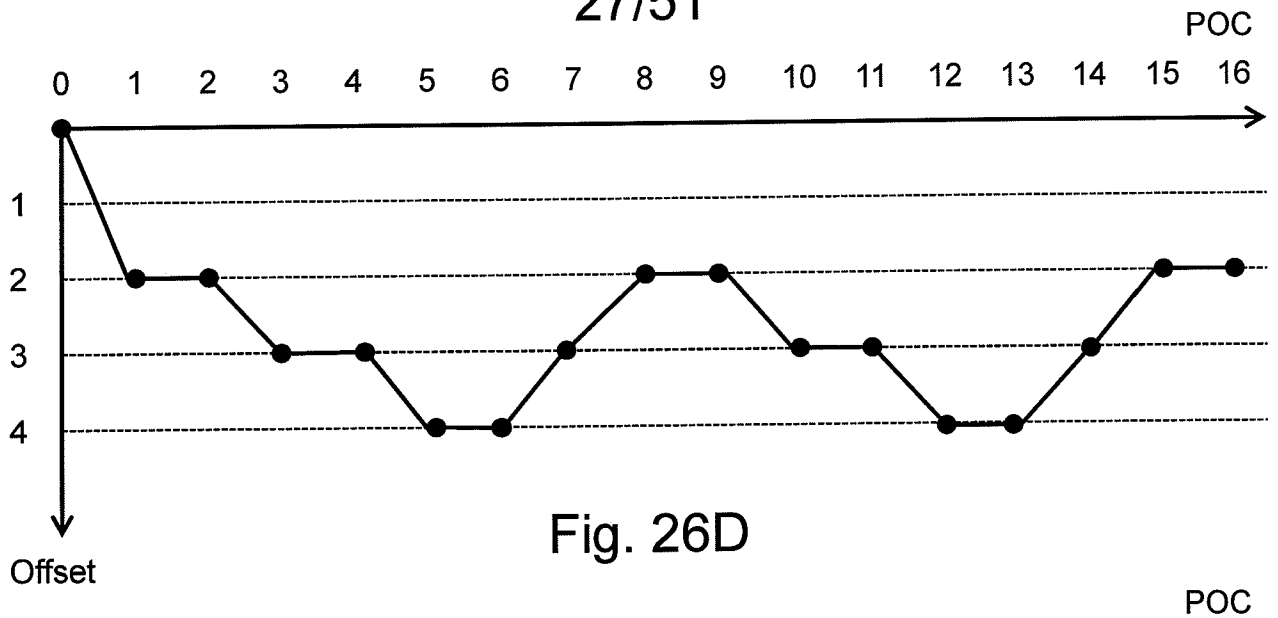


Fig. 26D

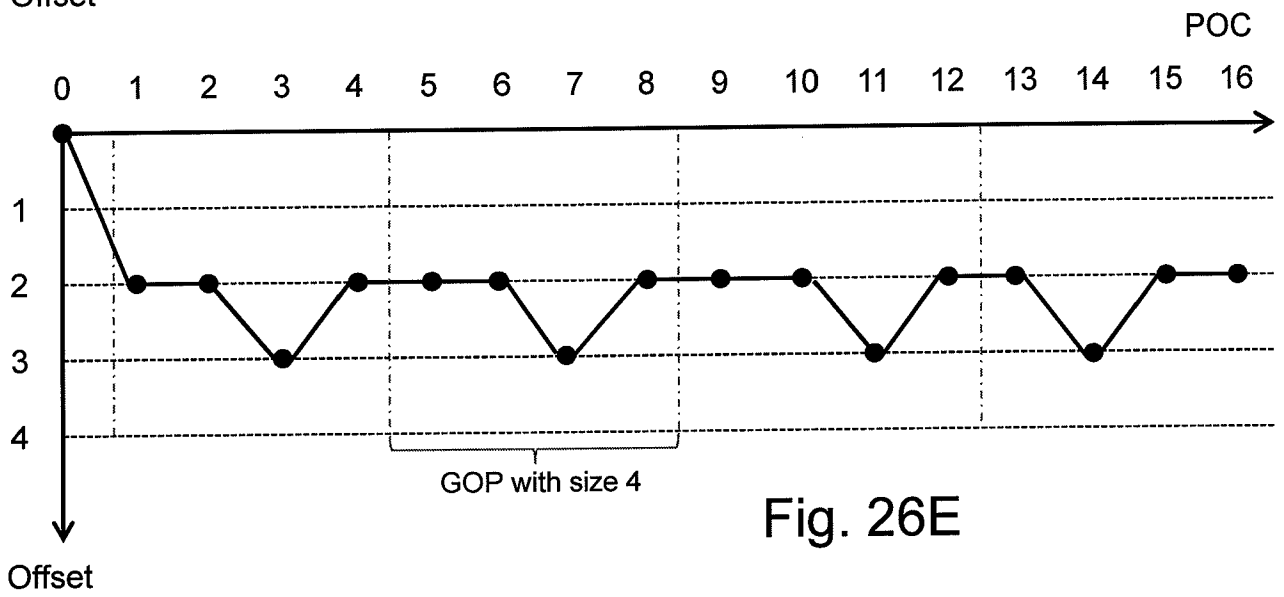


Fig. 26E

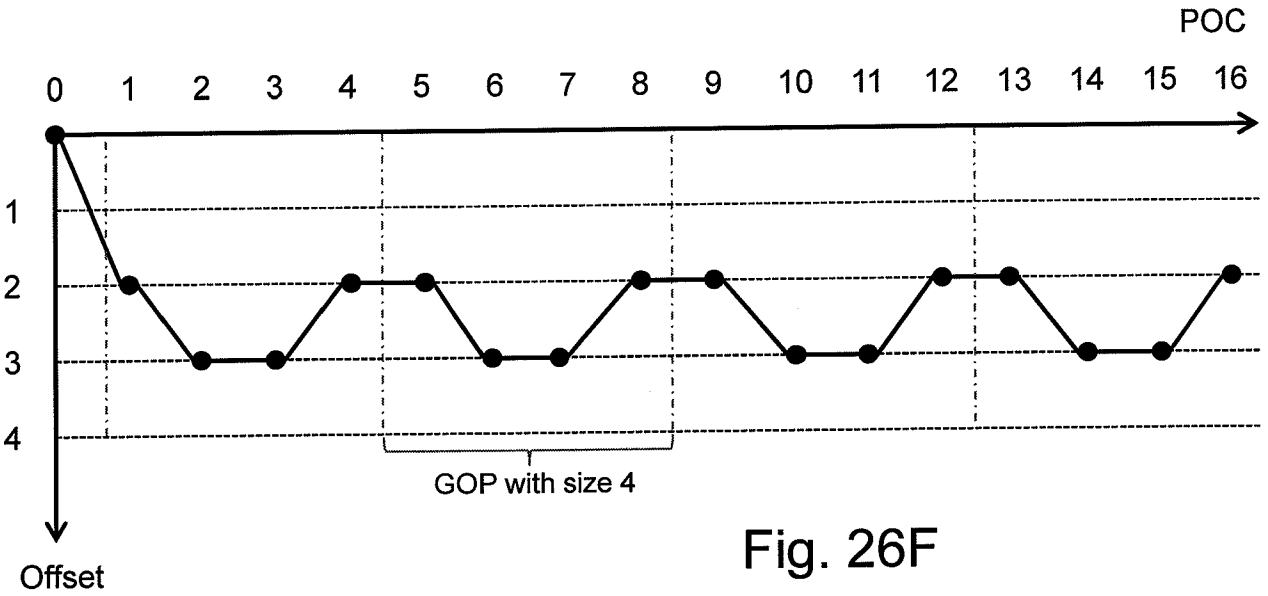
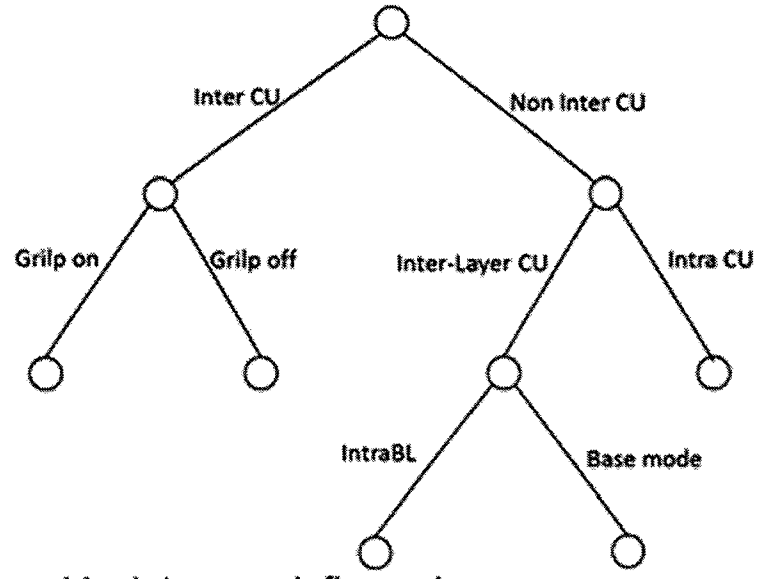


Fig. 26F



enable_intra_pred_flag = 1
enable_inter_layer_flag = 1
enable_grilp_flag = 1

Fig. 27

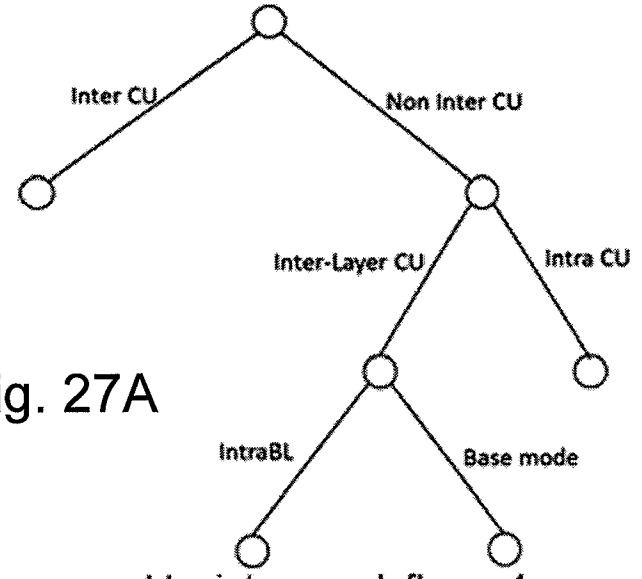
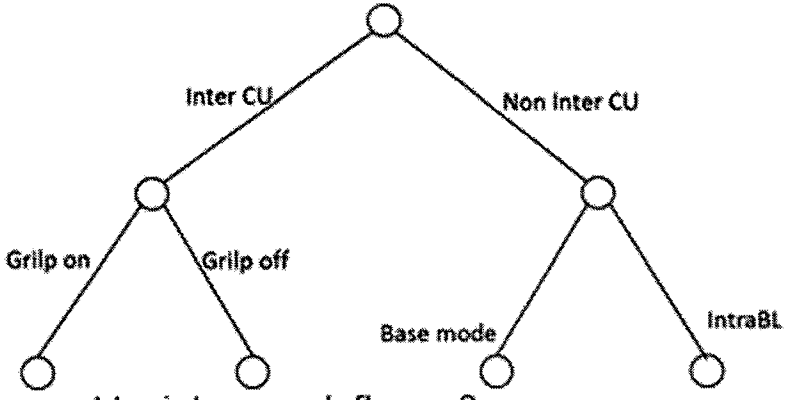


Fig. 27A

enable_intra_pred_flag = 1
enable_inter_layer_flag = 1
enable_grilp_flag = 0



enable_intra_pred_flag = 0
enable_inter_layer_flag = 1
enable_grilp_flag = 1

Fig. 27B

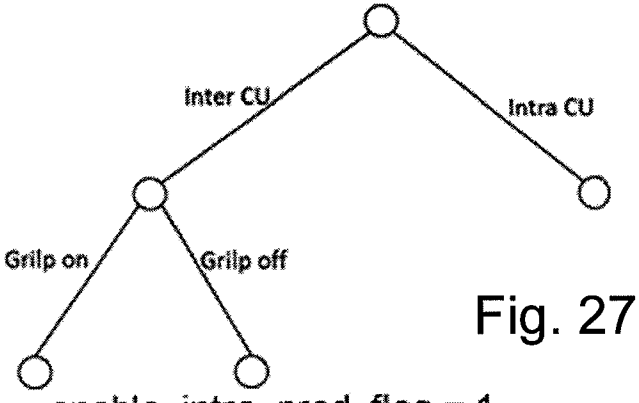


Fig. 27C

enable_intra_pred_flag = 1
enable_inter_layer_flag = 0
enable_grilp_flag = 1

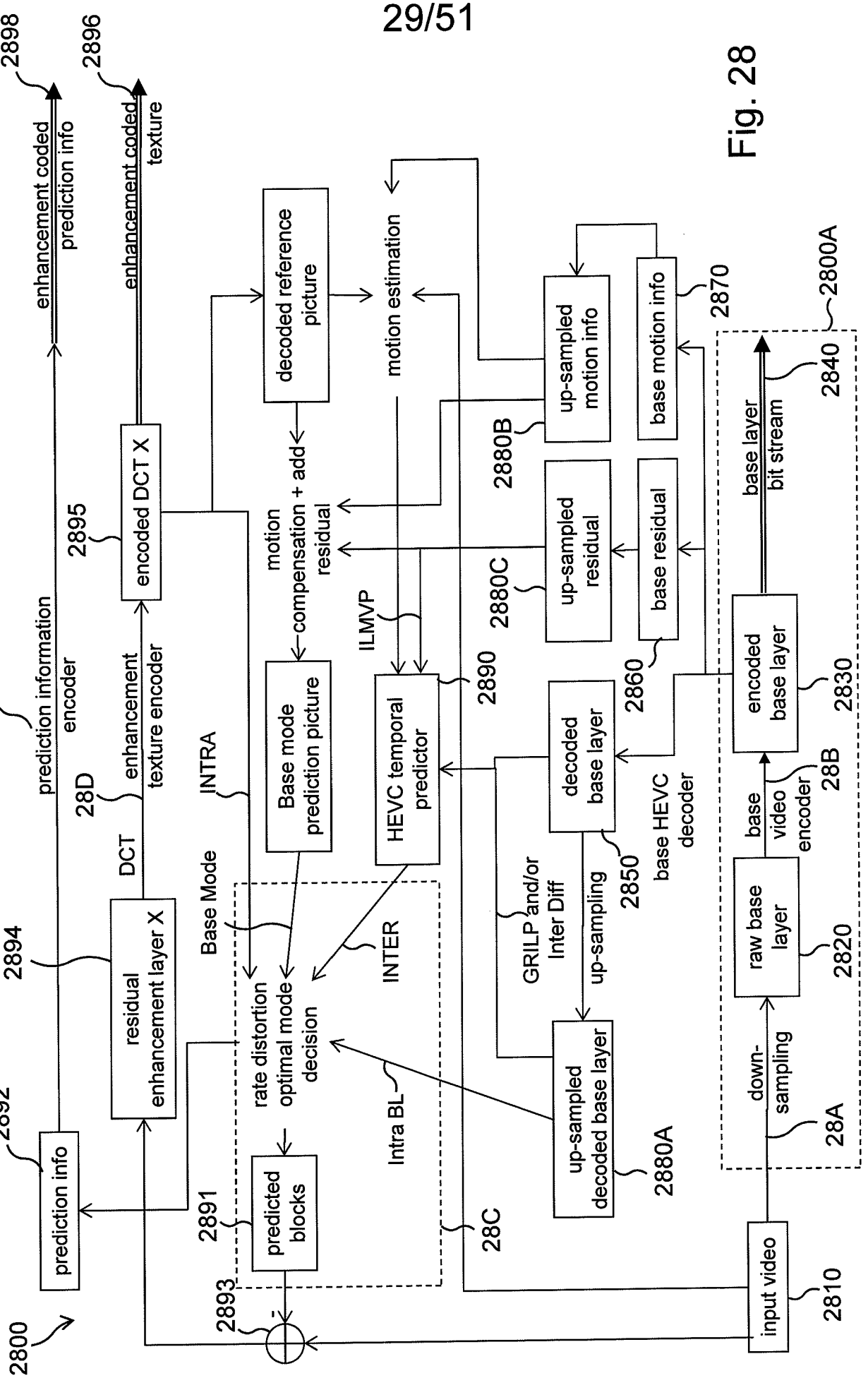

2898
2896
enhancement coded prediction info
enhancement coded texture
2897
prediction information encoder
2895
encoded DCT X
decoded reference picture
motion estimation
motion compensation + add residual
2880B
up-sampled motion info
base motion info
2870
2800A
2840
base layer bit stream
28D
DCT
enhancement texture encoder
INTRA
Base mode prediction picture
ILMVP
2890
HEVC temporal predictor
2880C
up-sampled residual
base residual
2860
2894
residual enhancement layer X
Base Mode
INTER
GRILP and/or Inter Diff
decoded base layer
2850
base HEVC decoder
encoded base layer
2830
2892
prediction info
rate distortion optimal mode decision
up-sampling
base video encoder
28B
2891
predicted blocks
Intra BL
up-sampled decoded base layer
2880A
raw base layer
2820
down-sampling
28A
2893
28C
input video
2810
2800

Fig. 28

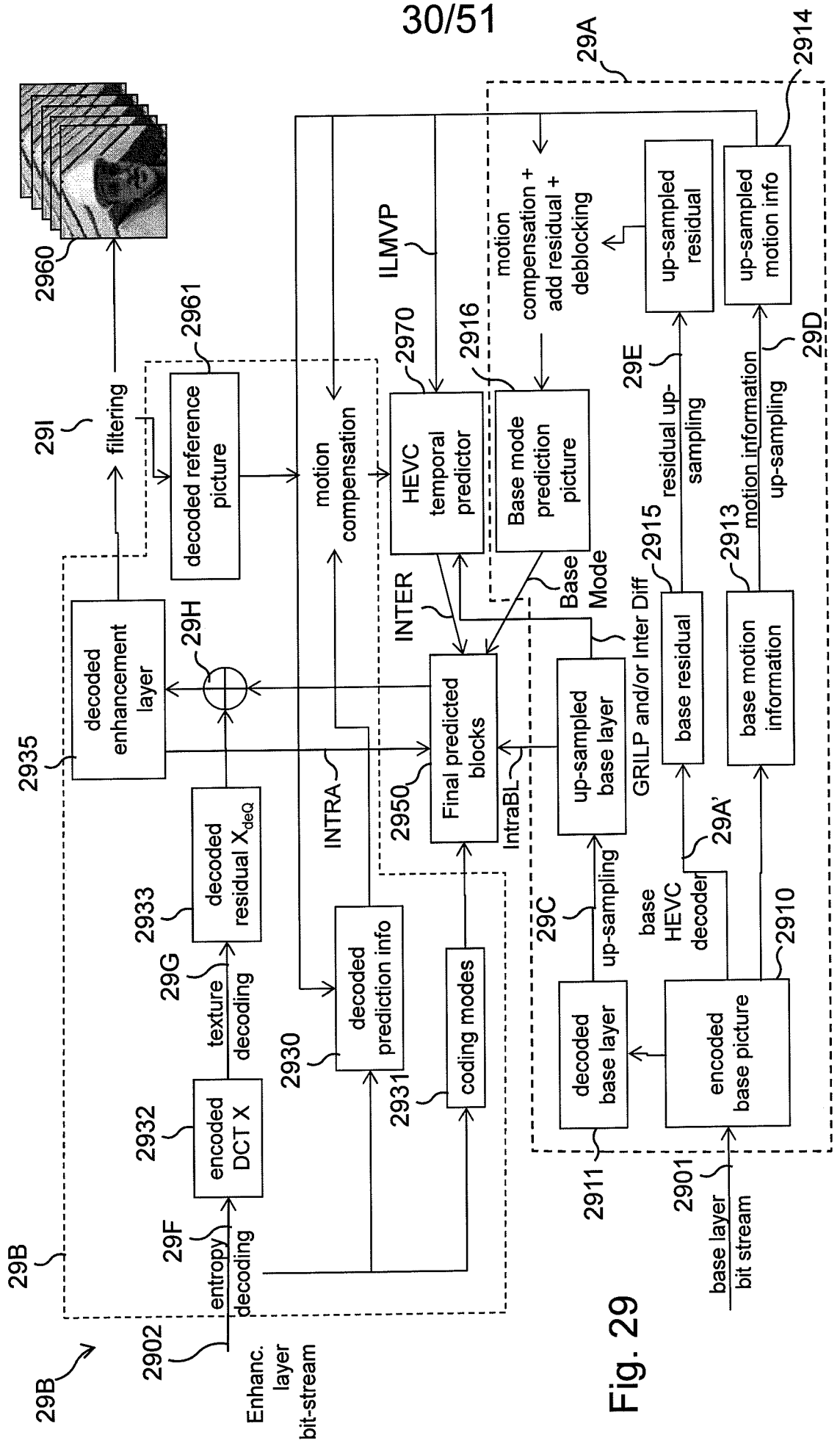

29B
2902
Enhanc. layer bit-stream
entropy decoding
29F
encoded DCT X
2932
texture decoding
29G
decoded residual X_{deQ}
2933
2935
decoded enhancement layer
29H
filtering
29I
2960
2961
decoded reference picture
motion compensation
INTRA
INTER
ILMVP
2970
HEVC temporal predictor
2916
Base mode prediction picture
Base Mode
2950
Final predicted blocks
IntraBL
2930
decoded prediction info
2931
coding modes
29A
2914
motion compensation + add residual + deblocking
29E
residual up-sampling
2915
GRILP and/or Inter Diff
base residual
29D
motion information up-sampling
2913
base motion information
up-sampled residual
up-sampled motion info
29C
up-sampling
up-sampled base layer
base HEVC decoder
29A'
2910
decoded base layer
encoded base picture
2911
2901
base layer bit stream

Fig. 29

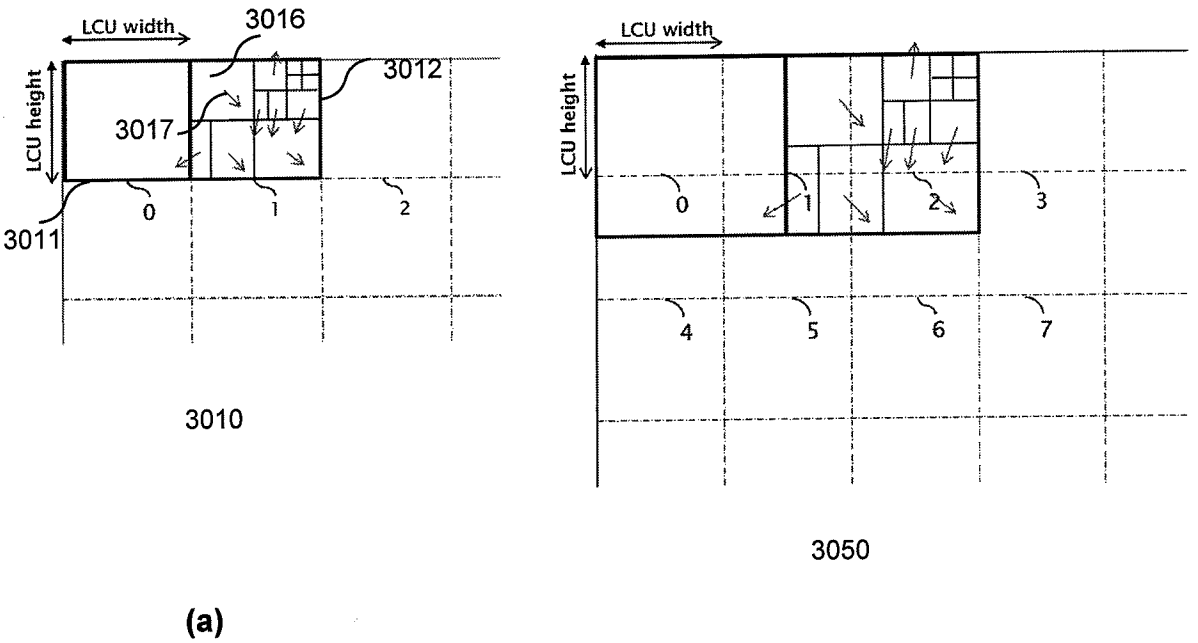


Fig. 30

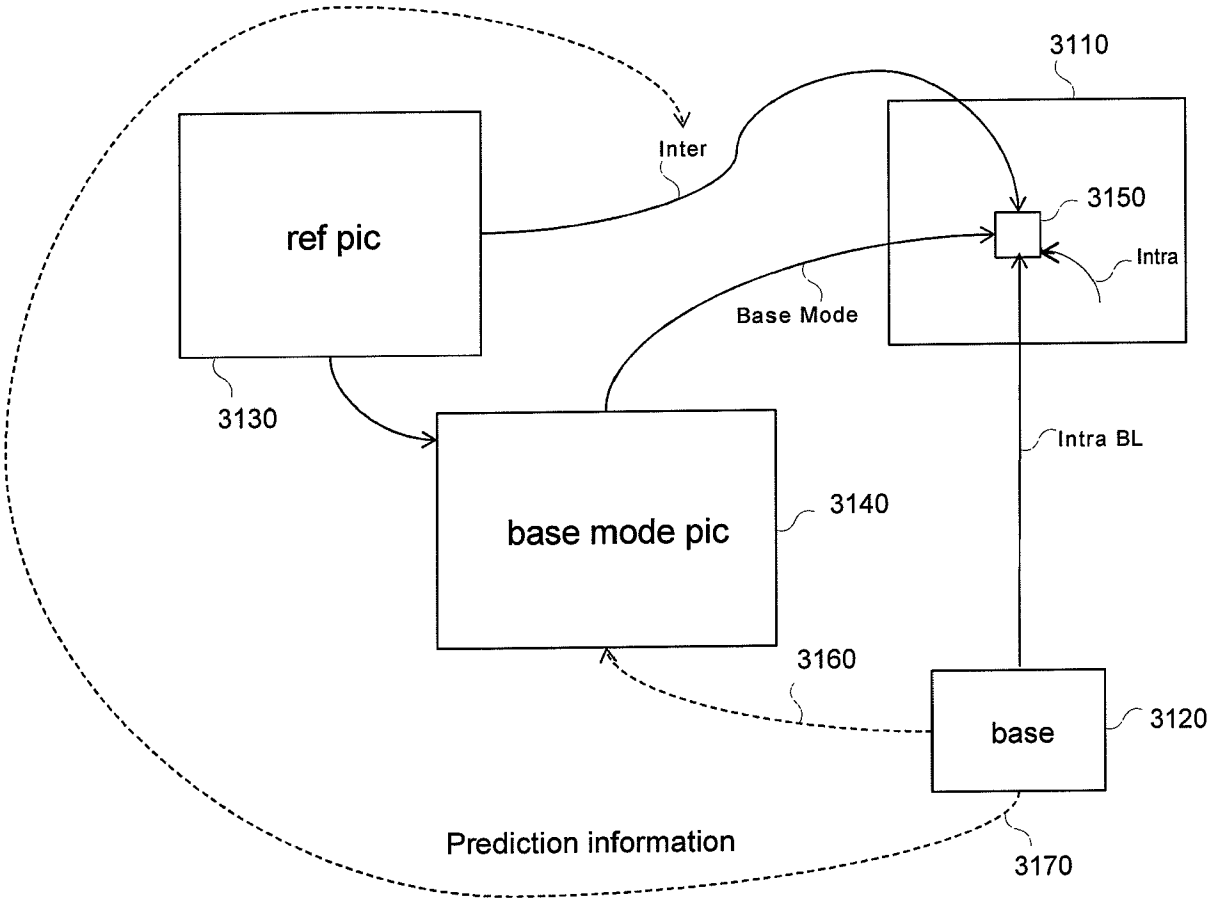


Fig. 31A

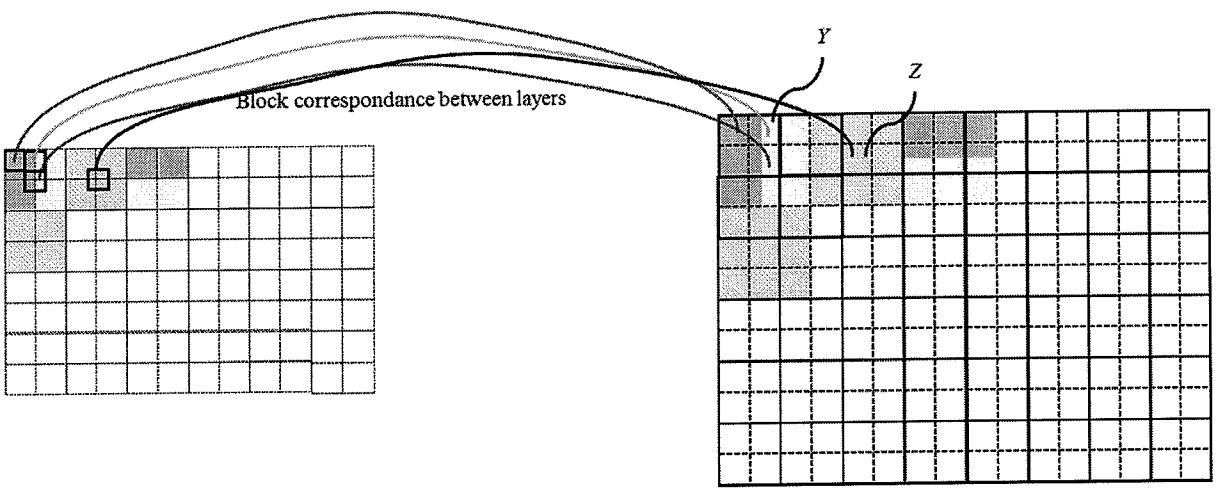


Fig. 31B

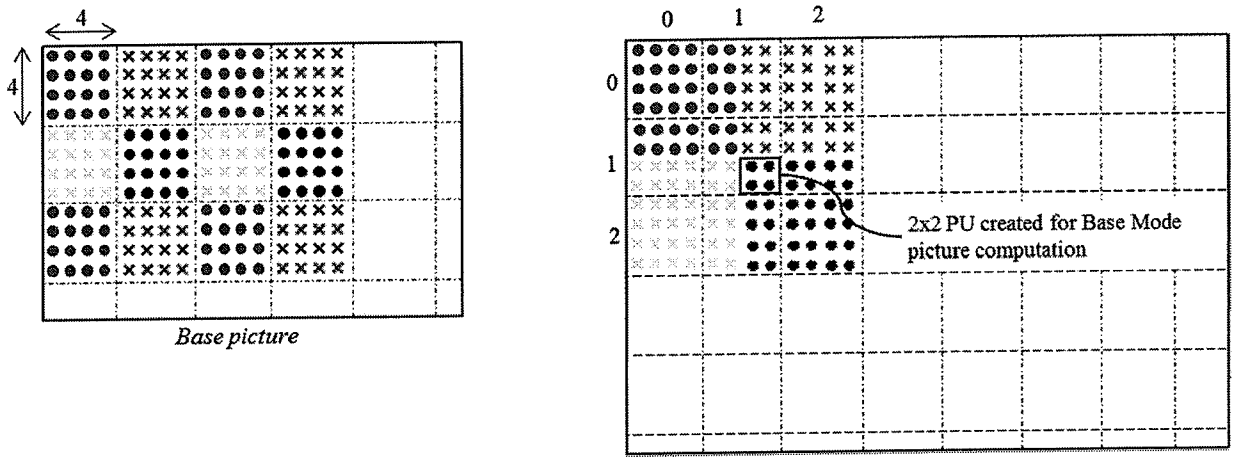


Fig. 32

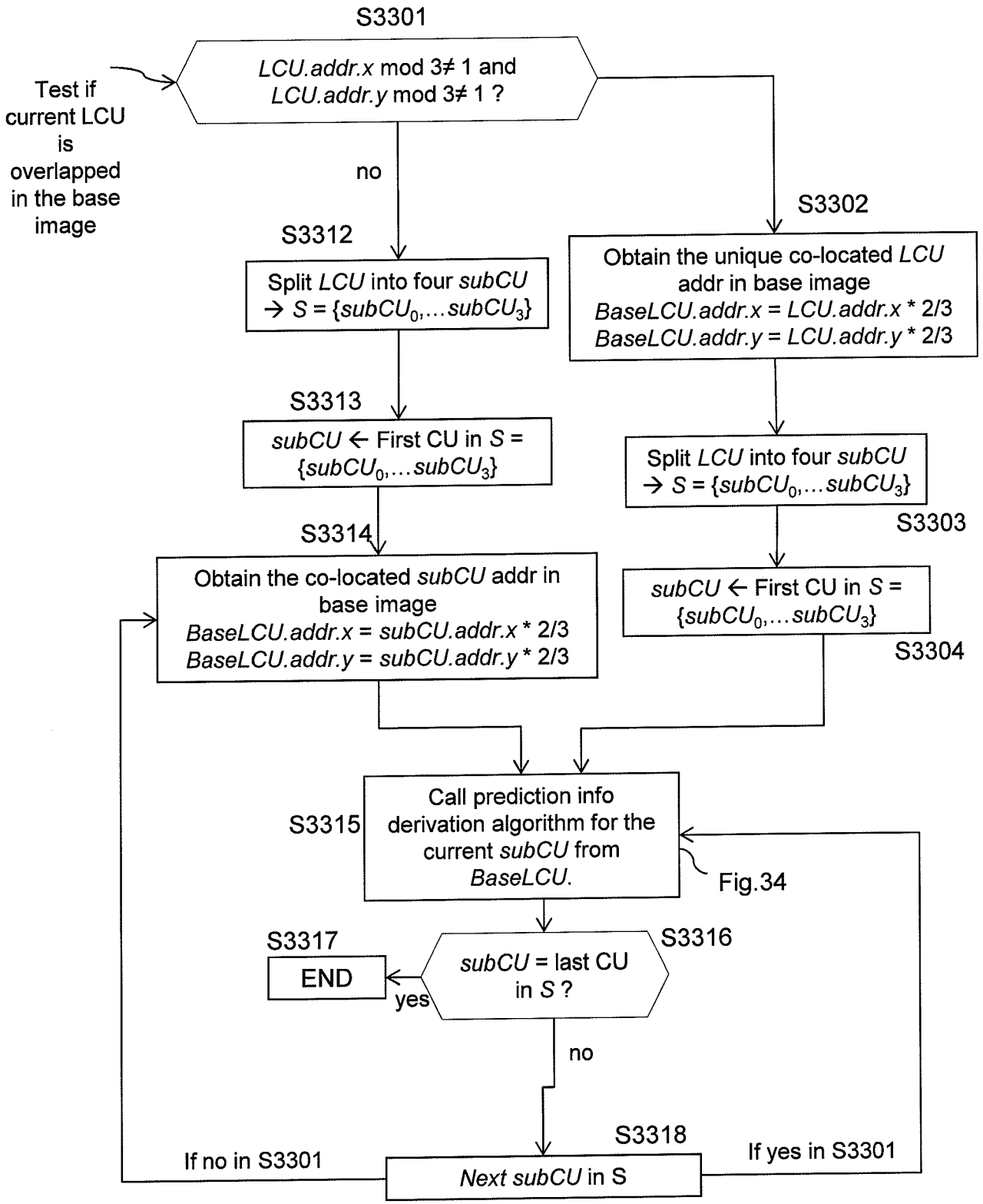

S3301
Test if current LCU is overlapped in the base image
LCU.addr.x mod 3≠ 1 and LCU.addr.y mod 3≠ 1 ?
no
S3302
Obtain the unique co-located LCU addr in base image
BaseLCU.addr.x = LCU.addr.x * 2/3
BaseLCU.addr.y = LCU.addr.y * 2/3
S3312
Split LCU into four subCU
→ S = {subCU0,...subCU3}
S3313
subCU ← First CU in S = {subCU0,...subCU3}
Split LCU into four subCU
→ S = {subCU0,...subCU3}
S3303
S3314
Obtain the co-located subCU addr in base image
BaseLCU.addr.x = subCU.addr.x * 2/3
BaseLCU.addr.y = subCU.addr.y * 2/3
subCU ← First CU in S = {subCU0,...subCU3}
S3304
S3315
Call prediction info derivation algorithm for the current subCU from BaseLCU.
Fig.34
S3317
END
yes
S3316
subCU = last CU in S ?
no
S3318
If no in S3301
Next subCU in S
If yes in S3301

Fig. 33

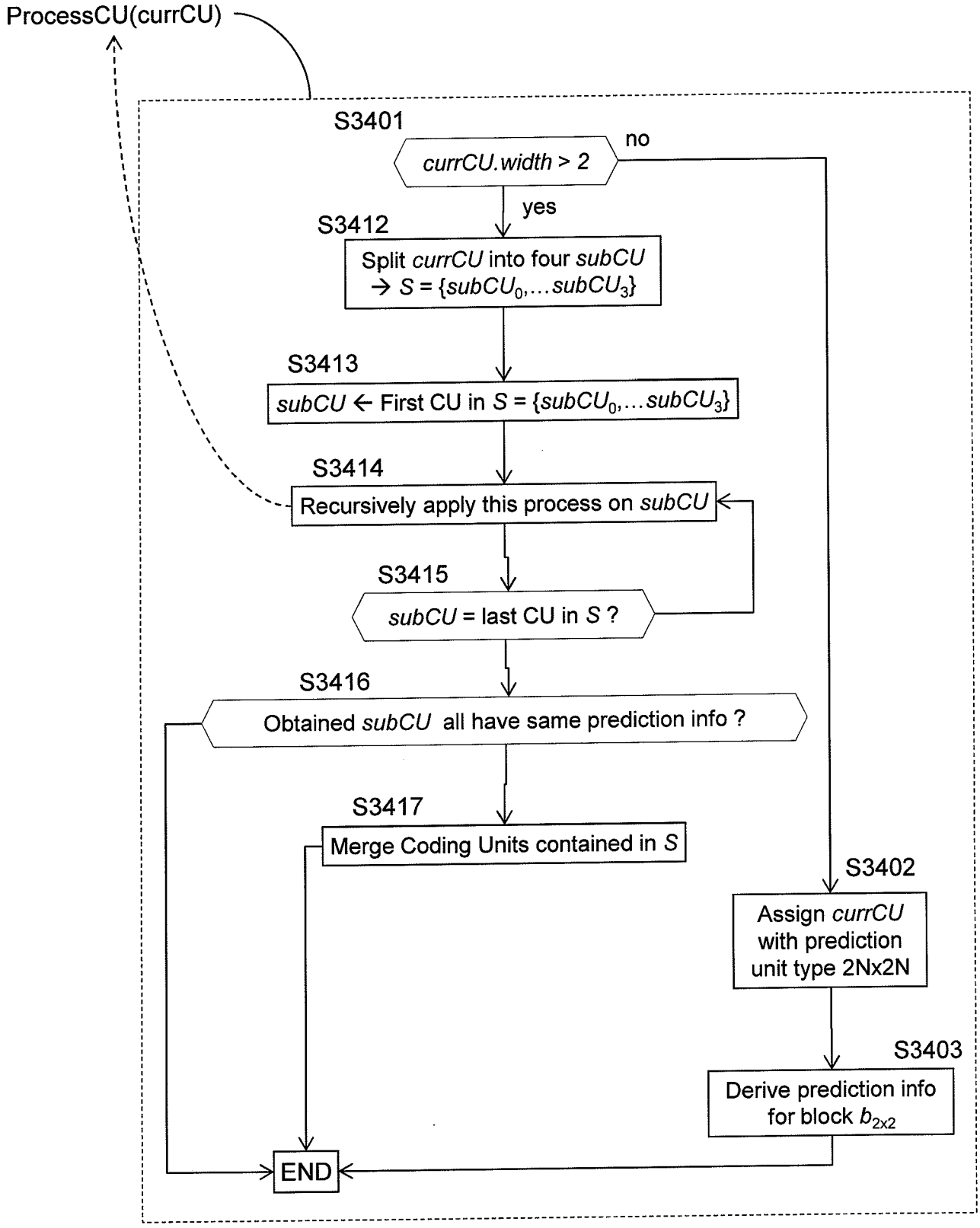

ProcessCU(currCU)
S3401
currCU.width > 2
no
yes
S3412
Split currCU into four subCU
→ S = {subCU0,...subCU3}
S3413
subCU ← First CU in S = {subCU0,...subCU3}
S3414
Recursively apply this process on subCU
S3415
subCU = last CU in S ?
S3416
Obtained subCU all have same prediction info ?
S3417
Merge Coding Units contained in S
S3402
Assign currCU with prediction unit type 2Nx2N
S3403
Derive prediction info for block b2x2
END

Fig. 34

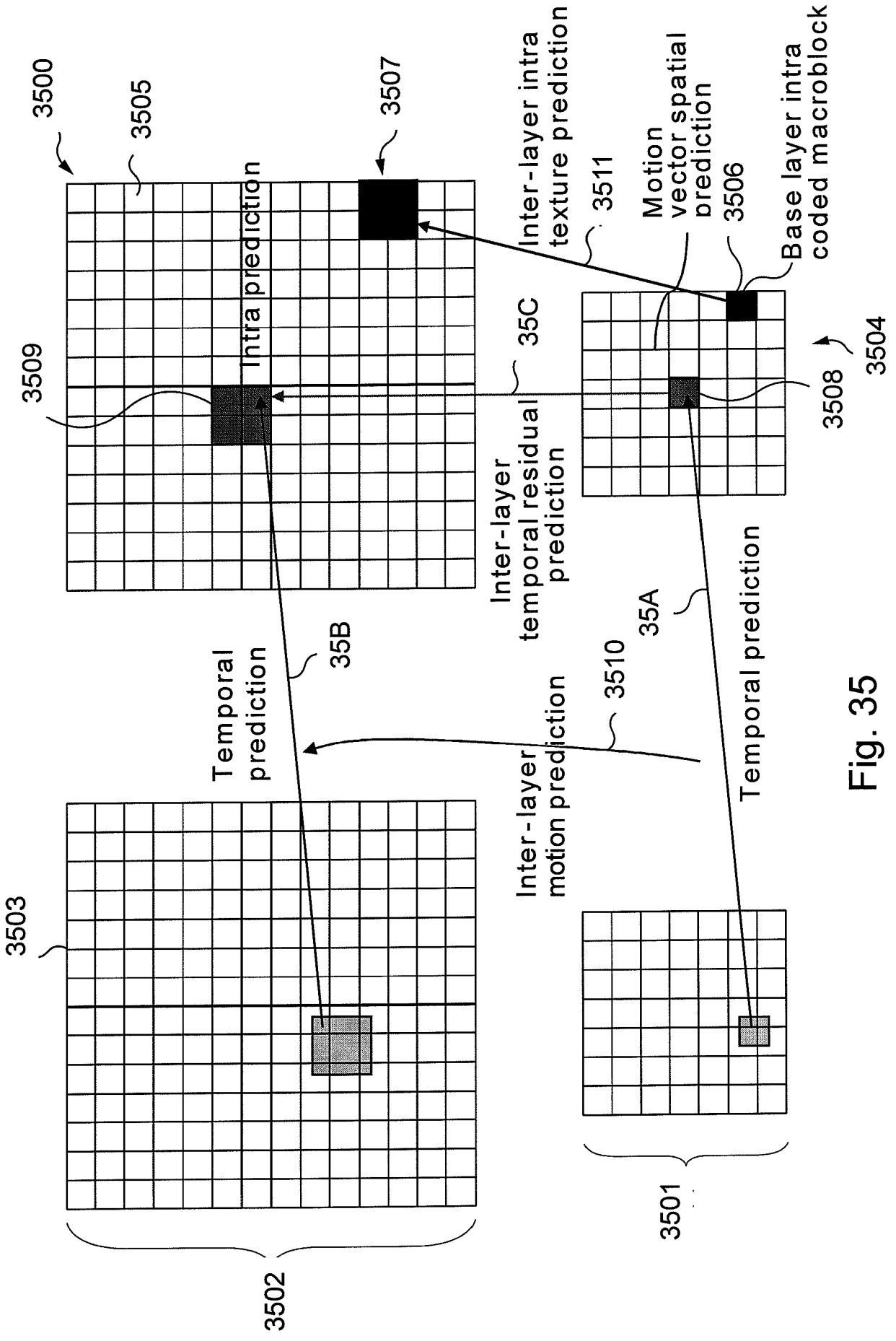

3500
3505
3507
Intra prediction
3509
Inter-layer intra texture prediction
3511
Motion vector spatial prediction
3506
Base layer intra coded macroblock
35C
3504
3508
Inter-layer temporal residual prediction
35A
Temporal prediction
35B
Temporal prediction
3510
Inter-layer motion prediction
3503
3501
3502

Fig. 35

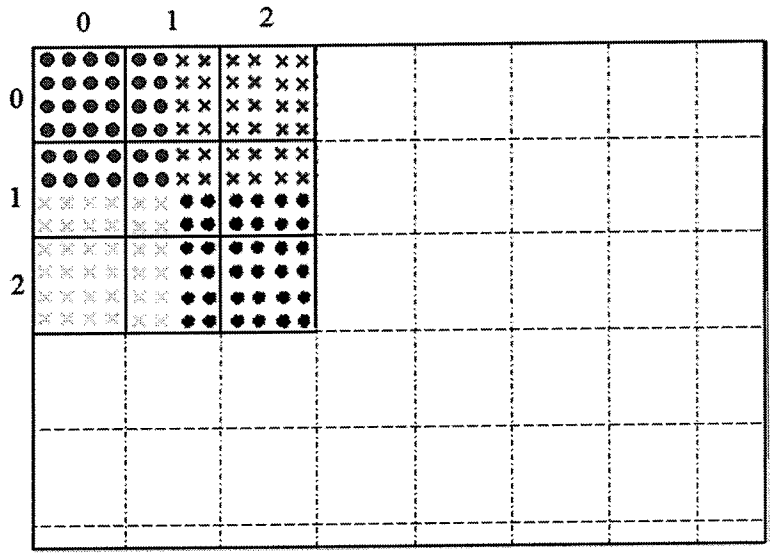


Fig. 36

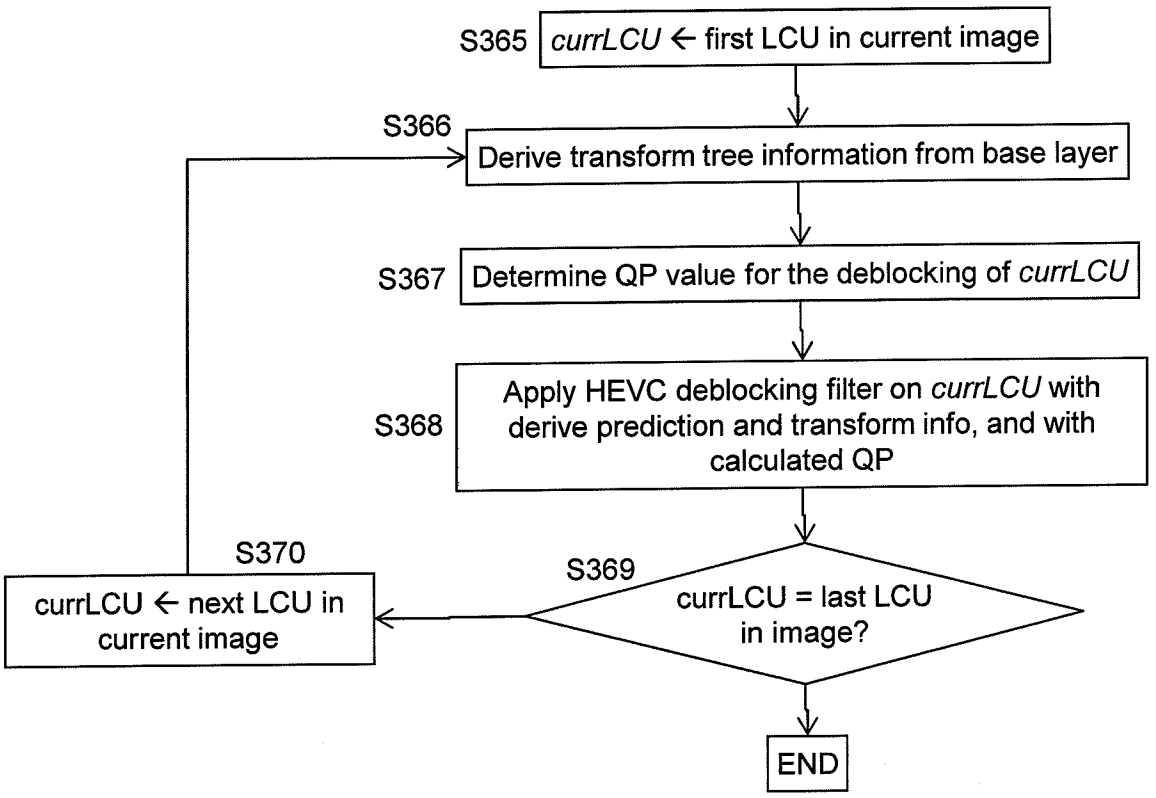


Fig. 36A

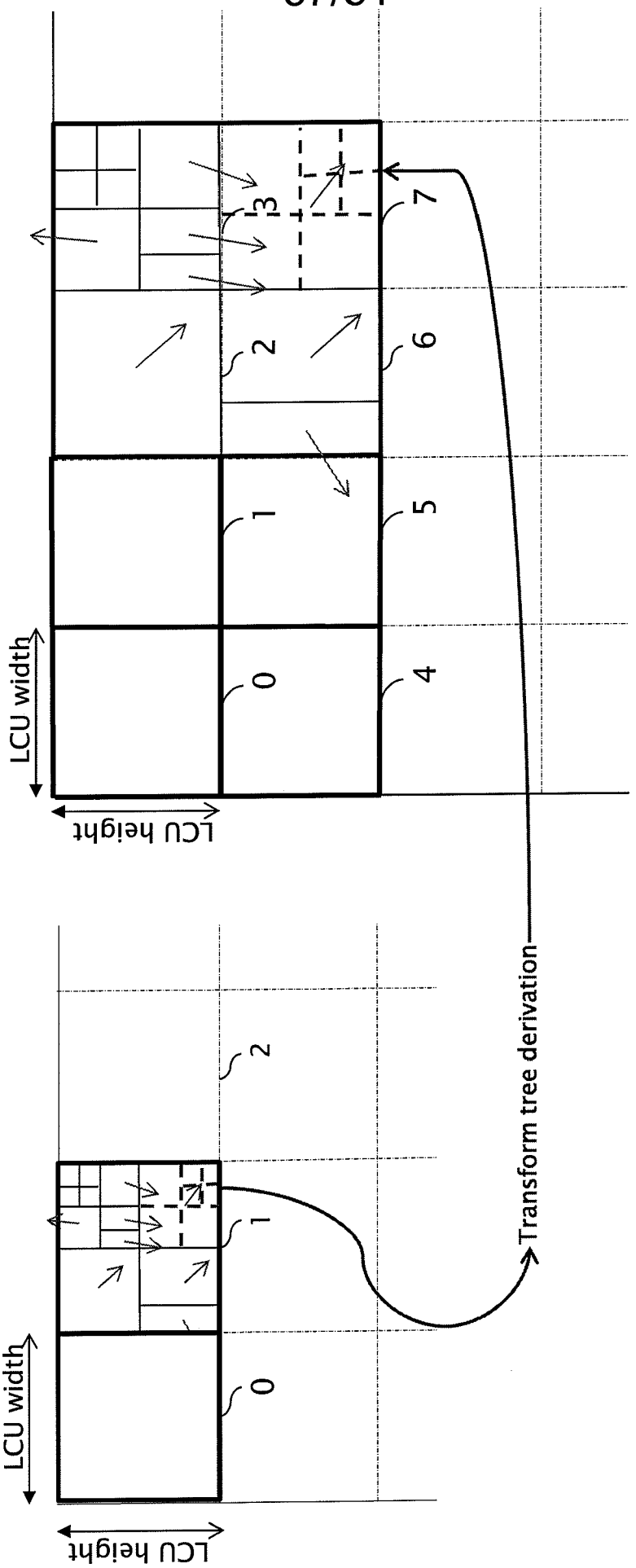

LCU width
LCU height
0
1
2
3
4
5
6
7
Transform tree derivation
LCU width
LCU height
0
1
2

Fig. 36B

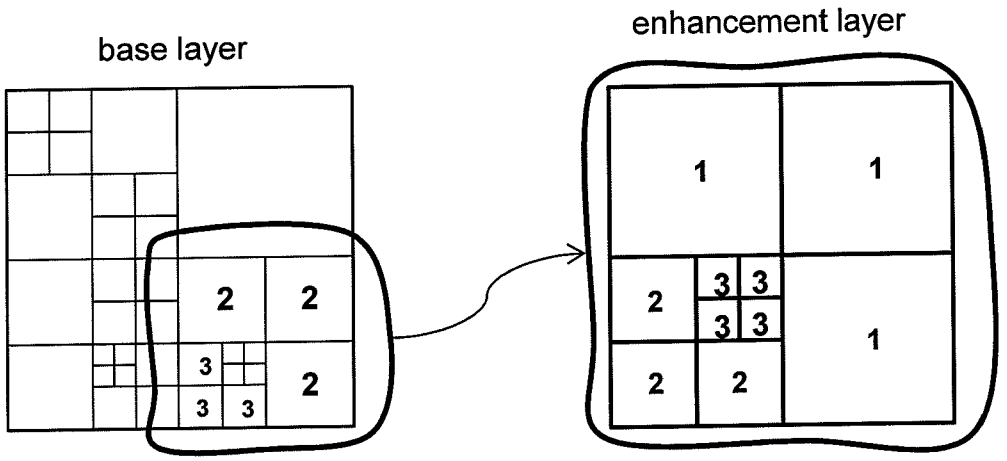


Fig. 36C

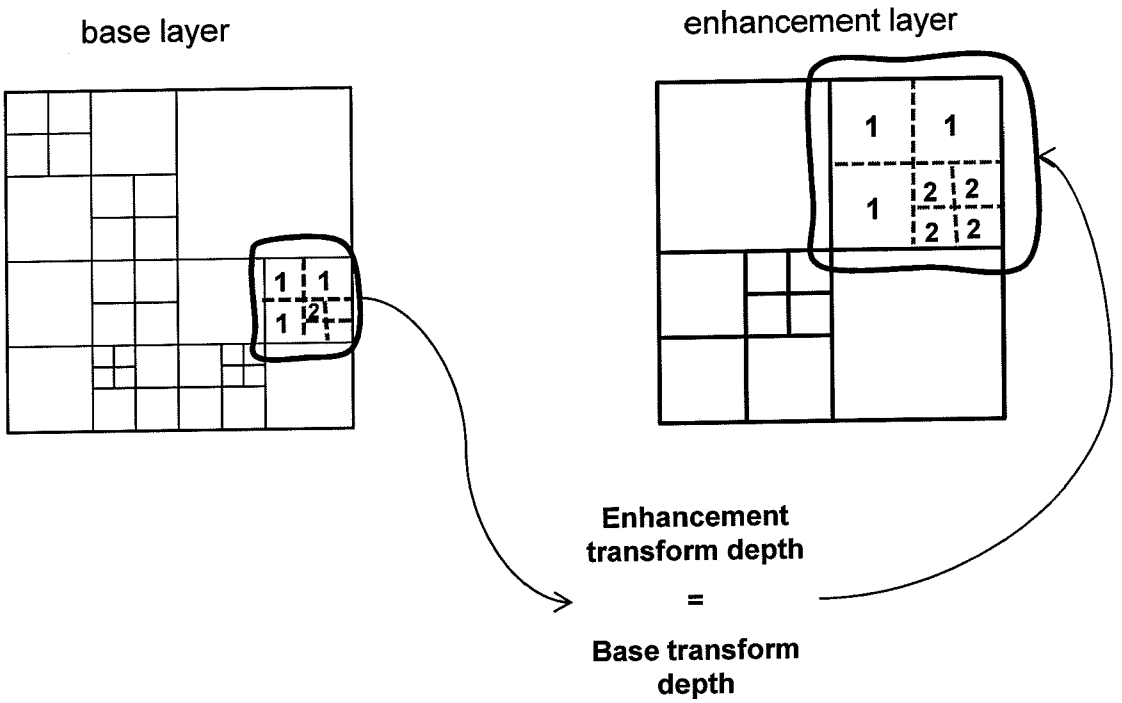


Fig. 36D

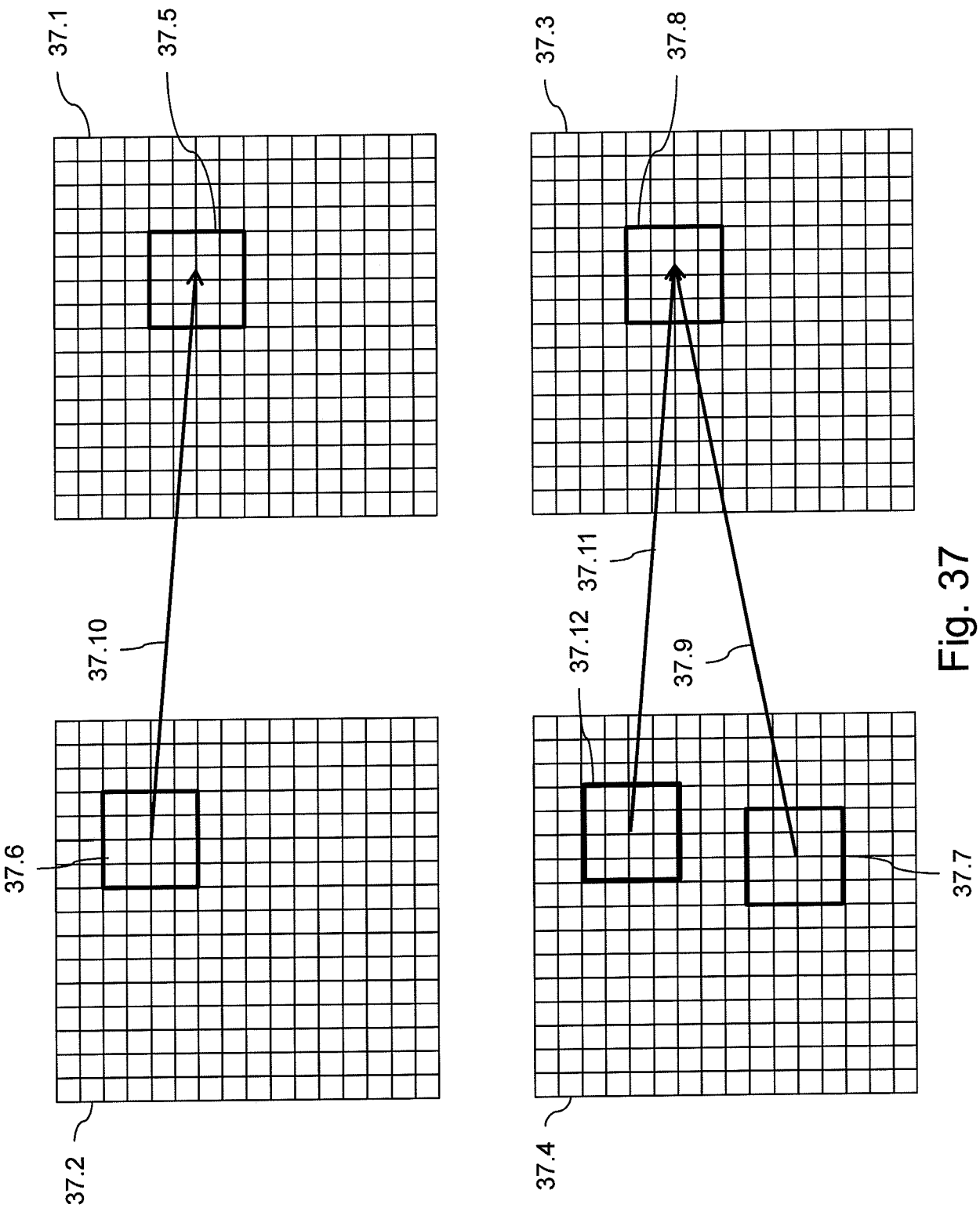


Fig. 37

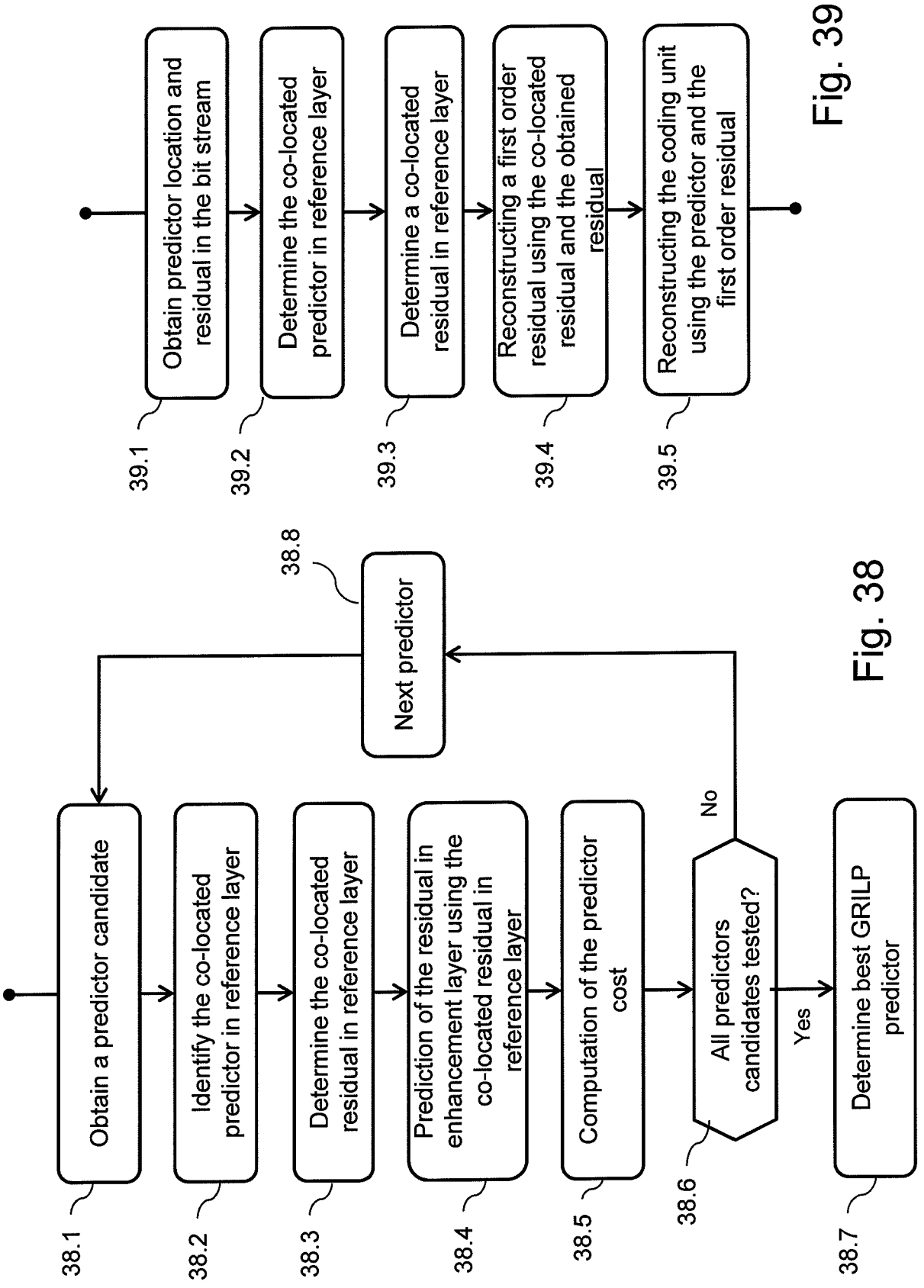

Obtain a predictor candidate
Identify the co-located predictor in reference layer
Determine the co-located residual in reference layer
Prediction of the residual in enhancement layer using the co-located residual in reference layer
Computation of the predictor cost
All predictors candidates tested?
No
Yes
Next predictor
Determine best GRILP predictor
38.1
38.2
38.3
38.4
38.5
38.6
38.7
38.8

Fig. 38

Obtain predictor location and residual in the bit stream
Determine the co-located predictor in reference layer
Determine a co-located residual in reference layer
Reconstructing a first order residual using the co-located residual and the obtained residual
Reconstructing the coding unit using the predictor and the first order residual
39.1
39.2
39.3
39.4
39.5

Fig. 39

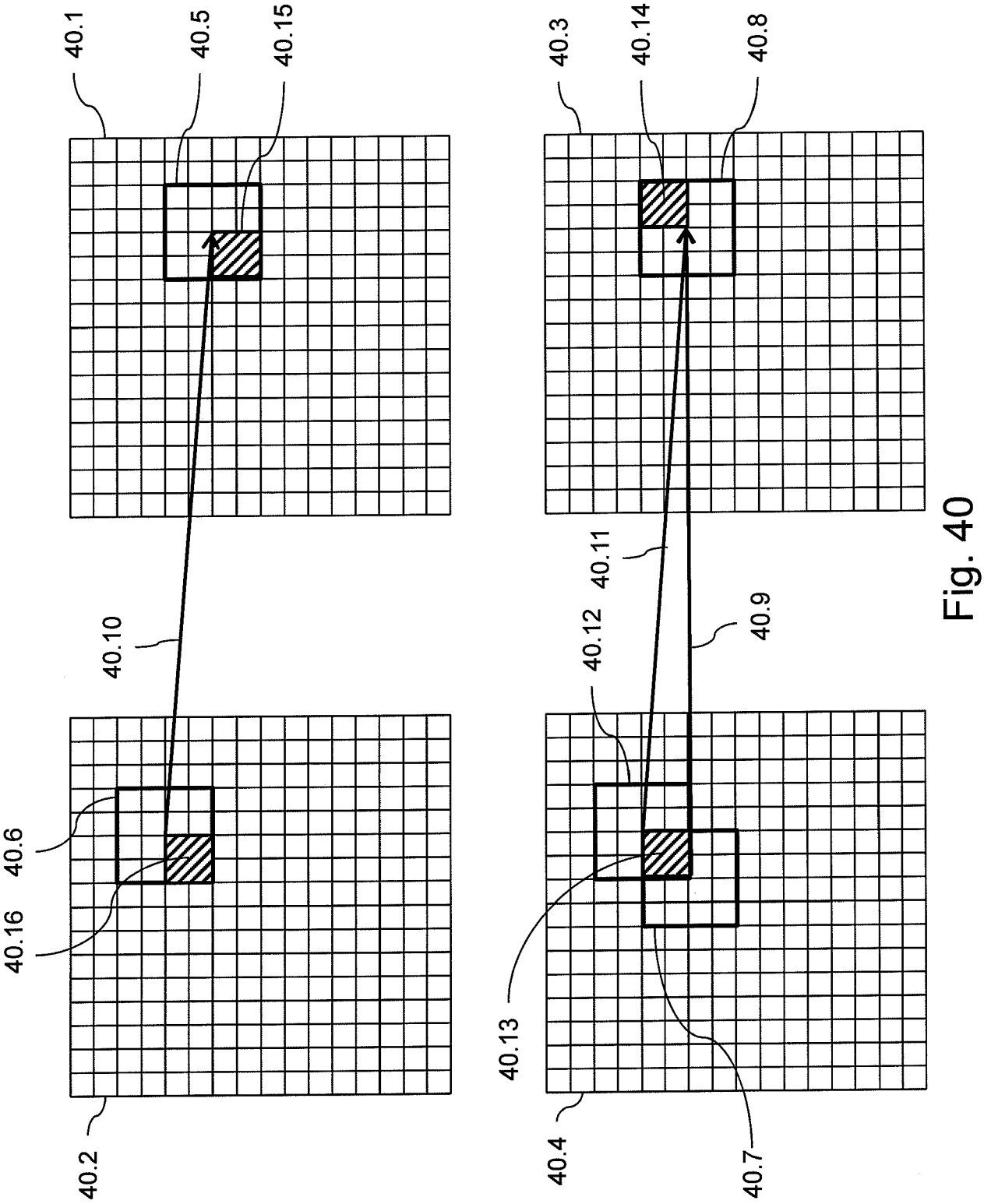

40.1
40.5
40.15
40.10
40.6
40.16
40.2
40.3
40.14
40.8
40.11
40.12
40.9
40.13
40.4
40.7

Fig. 40

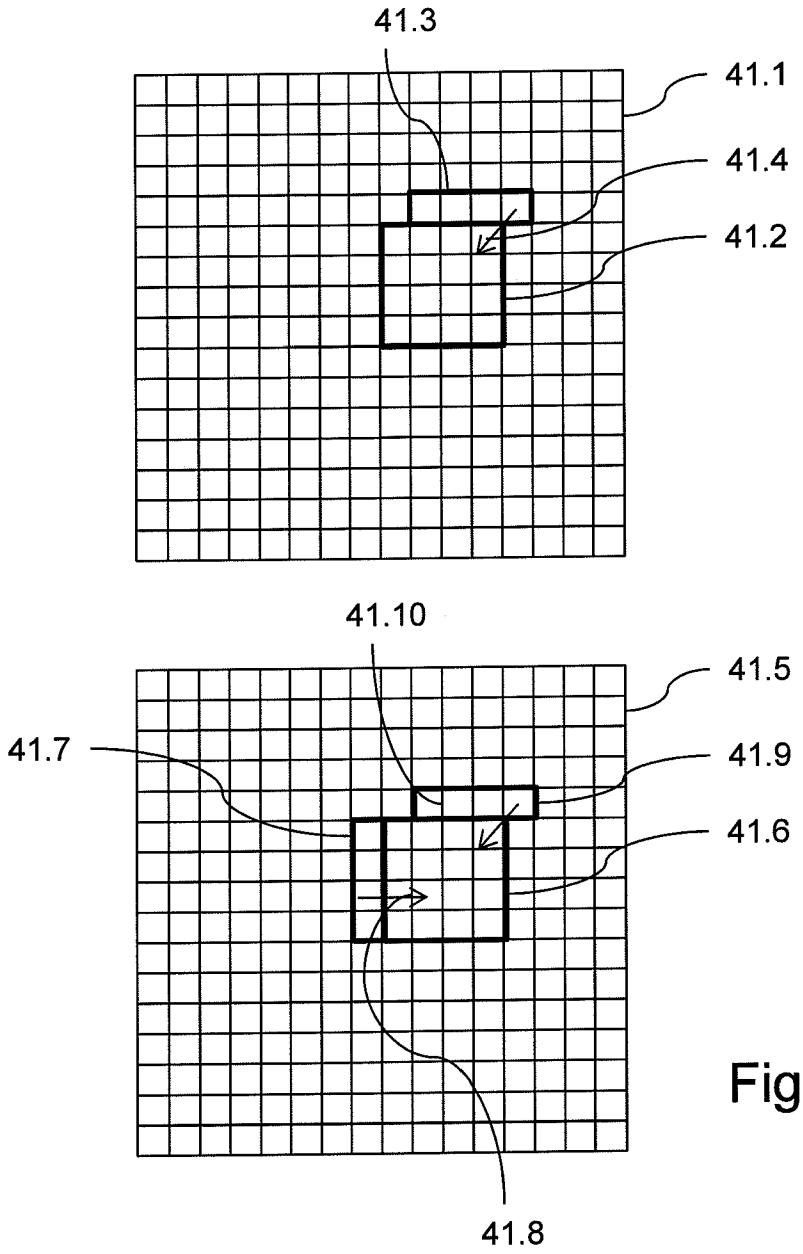


Fig. 41

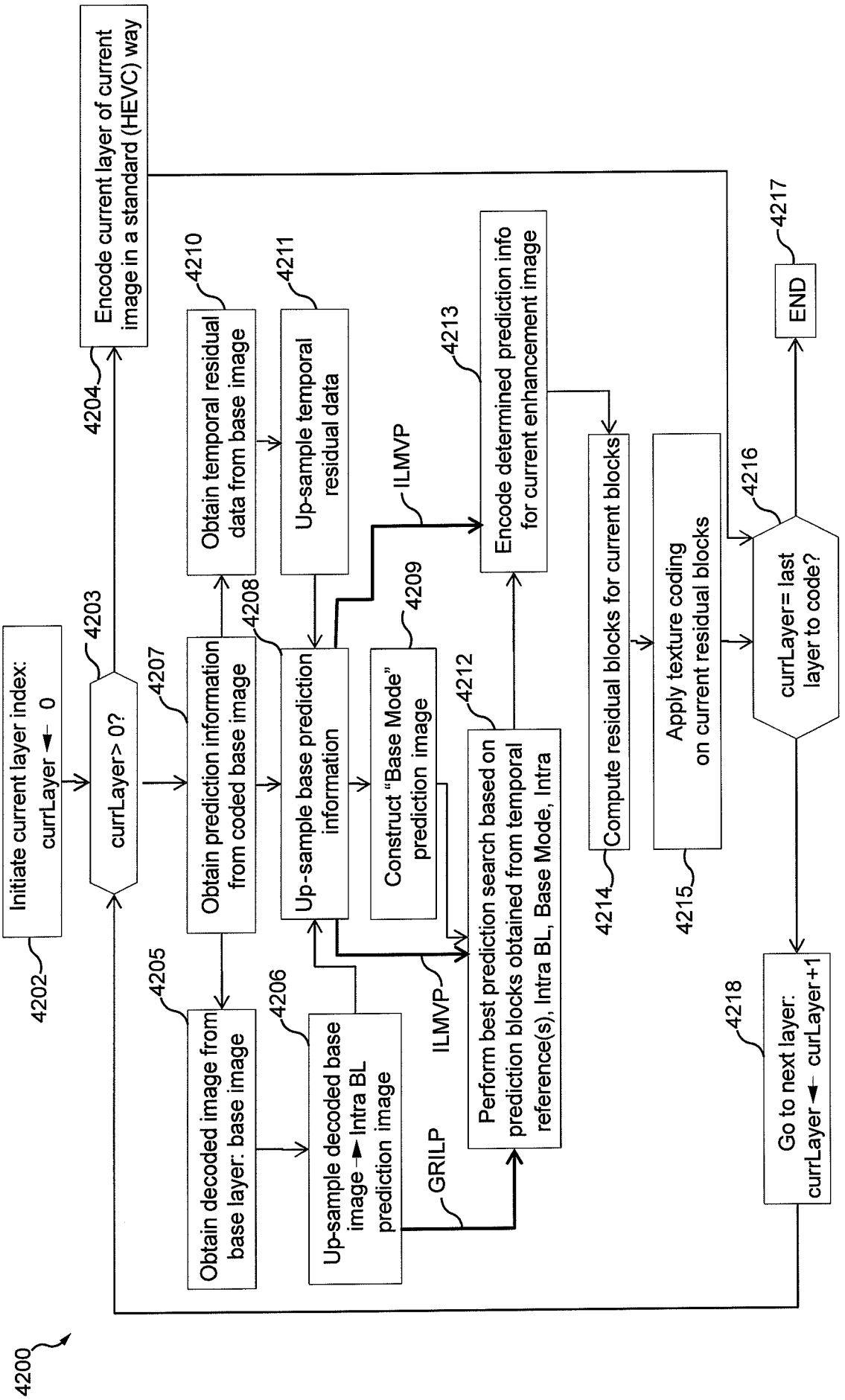

4200
Initiate current layer index: currLayer ← 0
4202
currLayer> 0?
4203
Encode current layer of current image in a standard (HEVC) way
4204
Obtain prediction information from coded base image
4207
Obtain decoded image from base layer: base image
4205
Up-sample decoded base image → Intra BL prediction image
4206
Obtain temporal residual data from base image
4208
Up-sample temporal residual data
4210
Up-sample base prediction information
4211
Construct "Base Mode" prediction image
4209
Perform best prediction search based on prediction blocks obtained from temporal reference(s), Intra BL, Base Mode, Intra
4212
Encode determined prediction info for current enhancement image
4213
Compute residual blocks for current blocks
4214
Apply texture coding on current residual blocks
4215
currLayer= last layer to code?
4216
END
4217
Go to next layer: currLayer ← curLayer+1
4218
ILMVP
ILMVP
GRILP

Fig. 42

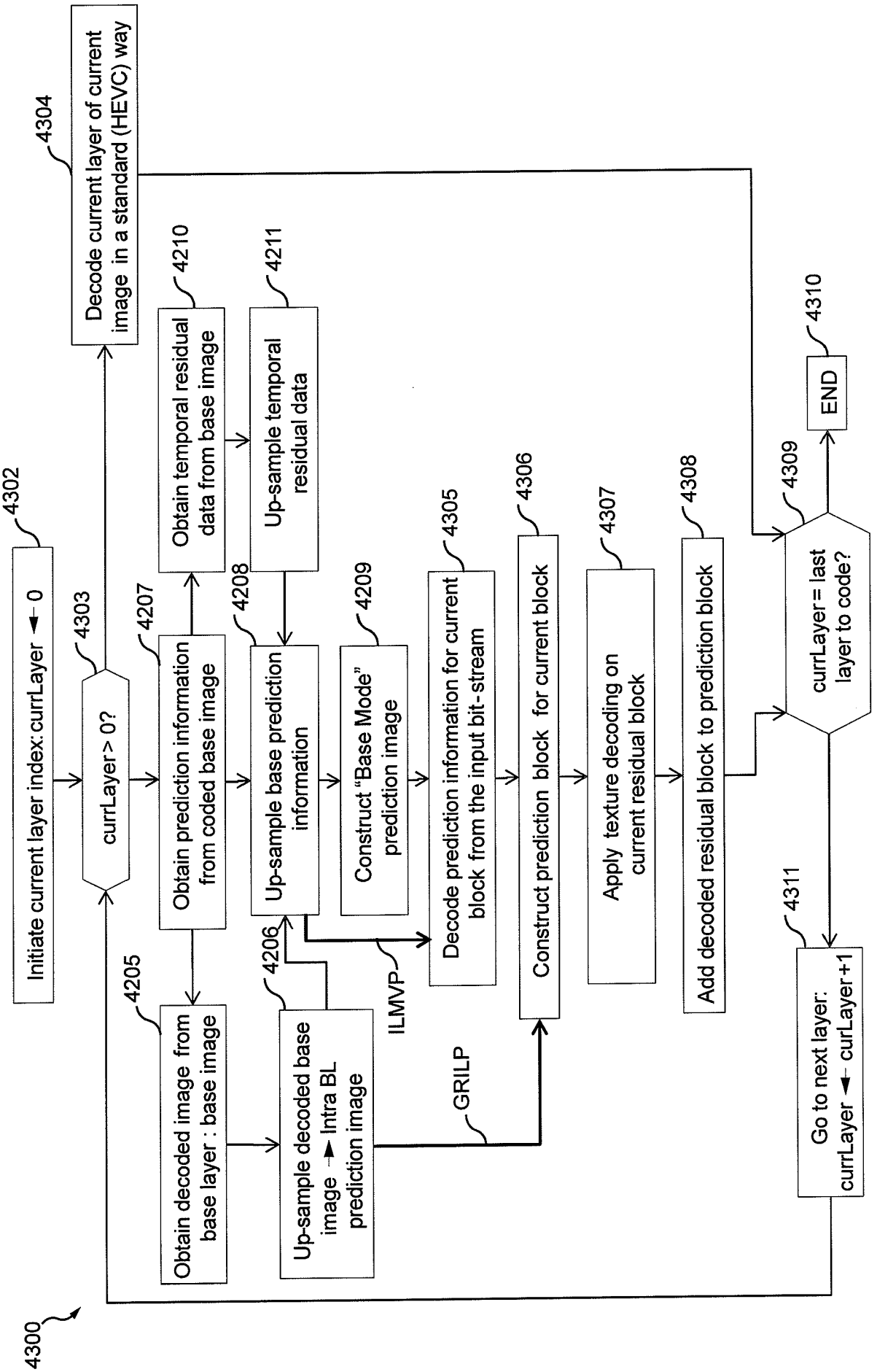

4300
Initiate current layer index: currLayer ← 0
4302
currLayer > 0?
4303
Decode current layer of current image in a standard (HEVC) way
4304
Obtain decoded image from base layer : base image
4205
Up-sample decoded base image → Intra BL prediction image
4206
Obtain prediction information from coded base image
4207
Up-sample base prediction information
4208
Construct "Base Mode" prediction image
4209
Obtain temporal residual data from base image
4210
Up-sample temporal residual data
4211
ILMVP
GRILP
Decode prediction information for current block from the input bit-stream
4305
Construct prediction block for current block
4306
Apply texture decoding on current residual block
4307
Add decoded residual block to prediction block
4308
currLayer = last layer to code?
4309
END
4310
Go to next layer: currLayer ← curLayer+1
4311

Fig. 43

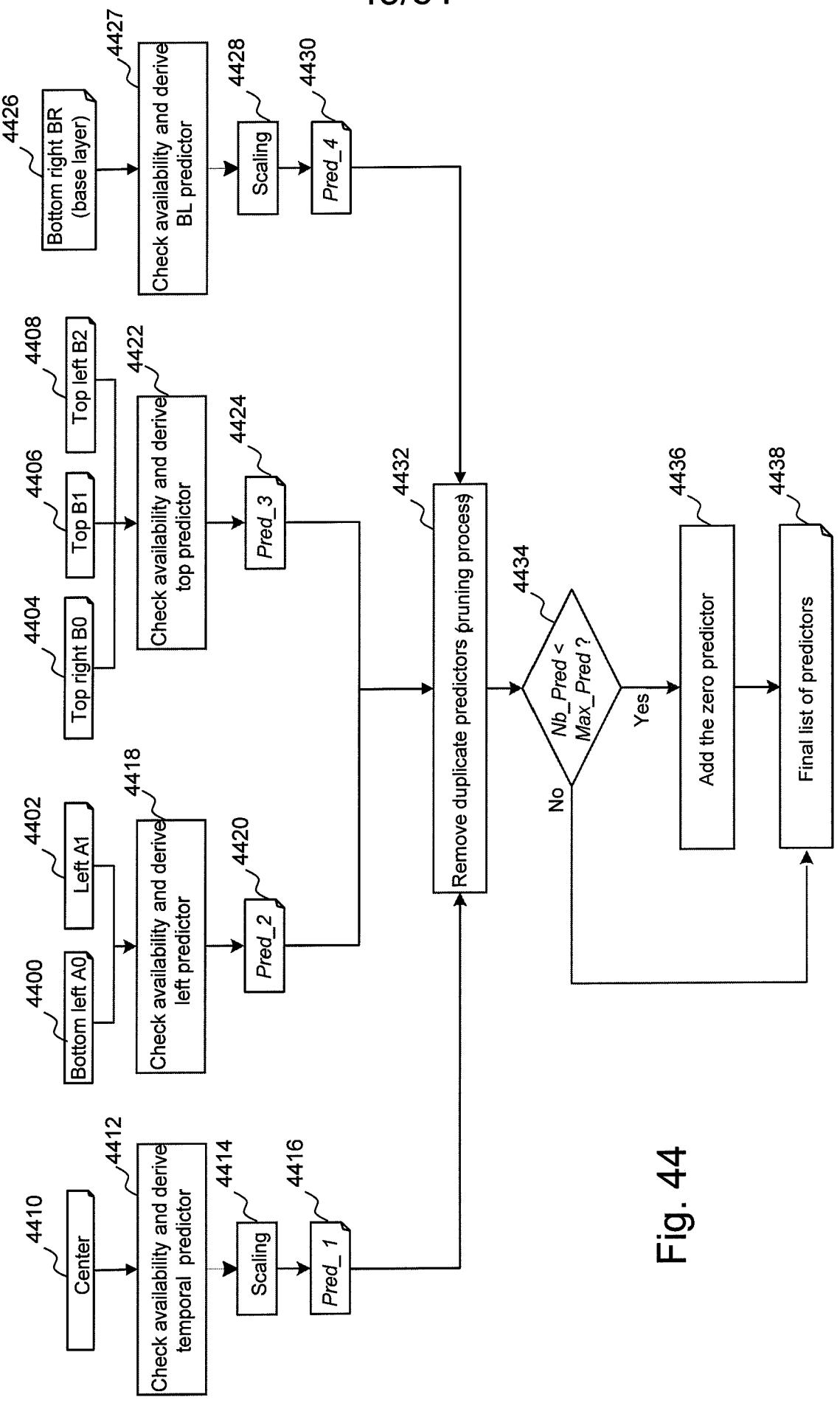

4410
Center
4412
Check availability and derive temporal predictor
4414
Scaling
4416
Pred_1
4400
Bottom left A0
4402
Left A1
4418
Check availability and derive left predictor
4420
Pred_2
4404
Top right B0
4406
Top B1
4408
Top left B2
4422
Check availability and derive top predictor
4424
Pred_3
4426
Bottom right BR (base layer)
4427
Check availability and derive BL predictor
4428
Scaling
4430
Pred_4
4432
Remove duplicate predictors (pruning process)
4434
Nb_Pred < Max_Pred ?
Yes
No
4436
Add the zero predictor
4438
Final list of predictors

Fig. 44

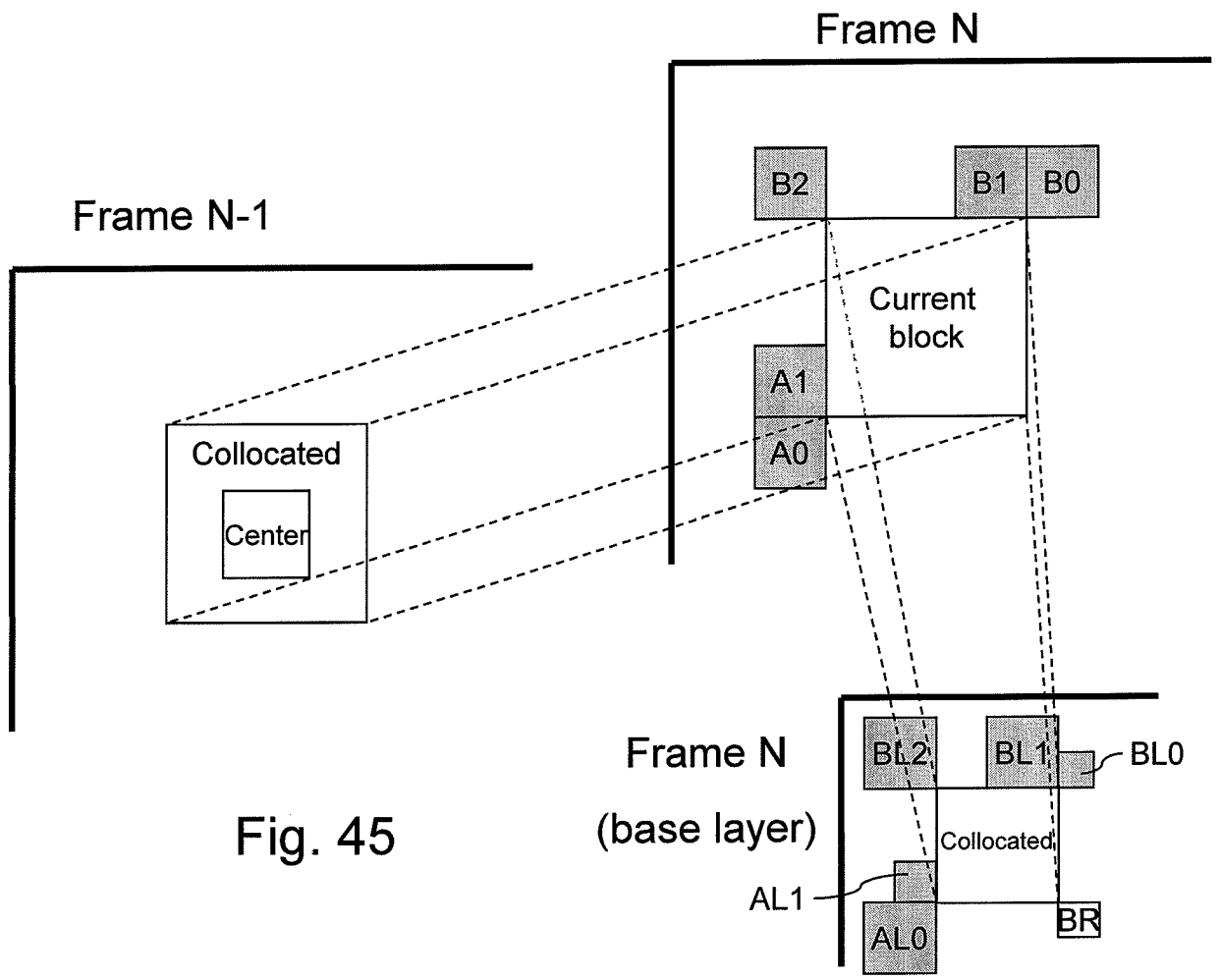

Frame N
Frame N-1
B2
B1
B0
Current block
A1
A0
Collocated
Center
Frame N
(base layer)
BL2
BL1
BL0
Collocated
AL1
AL0
BR

Fig. 45

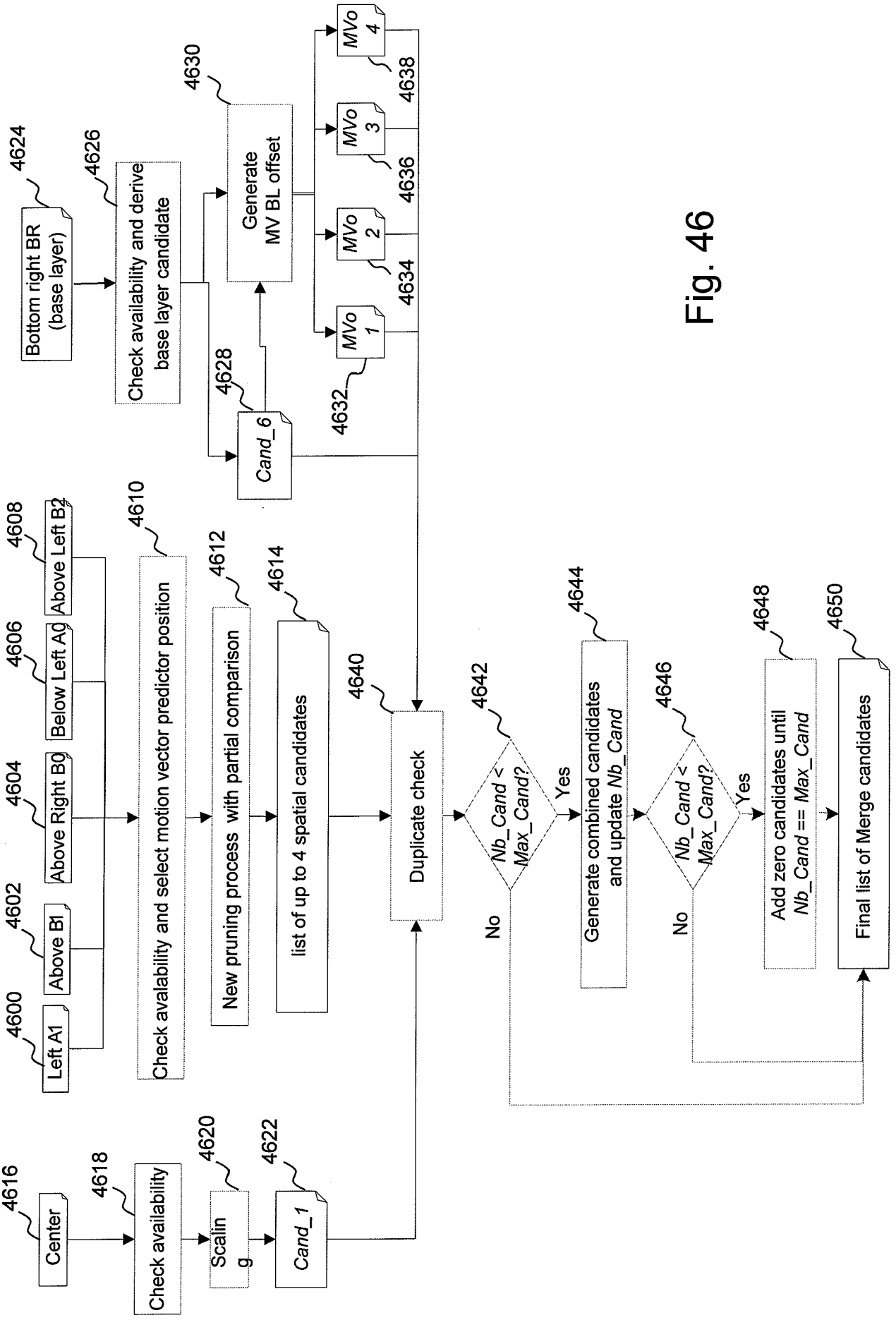

4624
Bottom right BR (base layer)
4626
Check availability and derive base layer candidate
4630
Generate MV BL offset
MVo 4
MVo 3
MVo 2
MVo 1
4638
4636
4634
4632
4628
Cand_6
4600
Left A1
4602
Above B1
4604
Above Right B0
4606
Below Left A0
4608
Above Left B2
4610
Check availability and select motion vector predictor position
4612
New pruning process with partial comparison
4614
list of up to 4 spatial candidates
4640
Duplicate check
4642
Nb_Cand < Max_Cand?
No
Yes
4644
Generate combined candidates and update Nb_Cand
4646
Nb_Cand < Max_Cand?
No
Yes
4648
Add zero candidates until Nb_Cand == Max_Cand
4650
Final list of Merge candidates
4616
Center
4618
Check availability
4620
Scaling
4622
Cand_1

Fig. 46

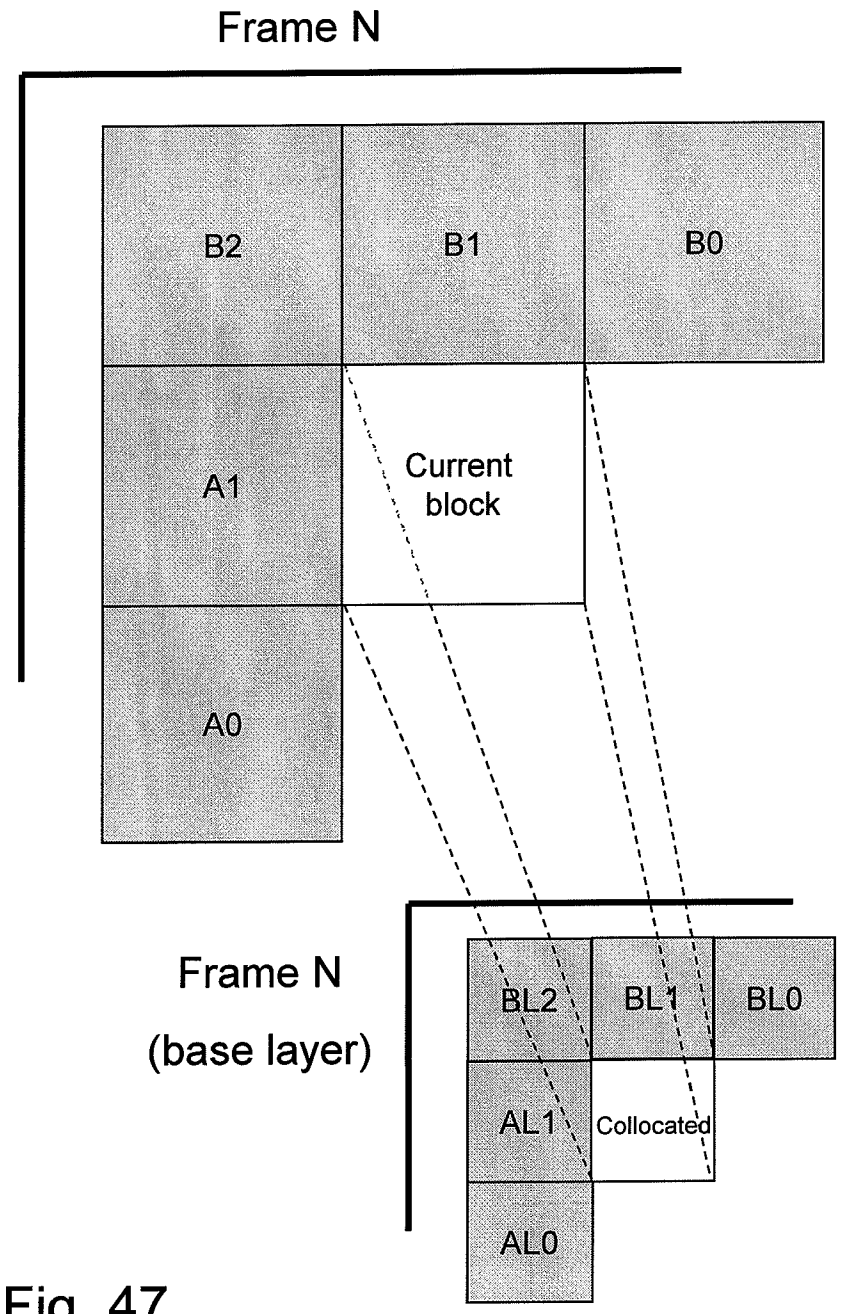

Frame N
B2
B1
B0
A1
Current block
A0
Frame N
(base layer)
BL2
BL1
BL0
AL1
Collocated
AL0

Fig. 47

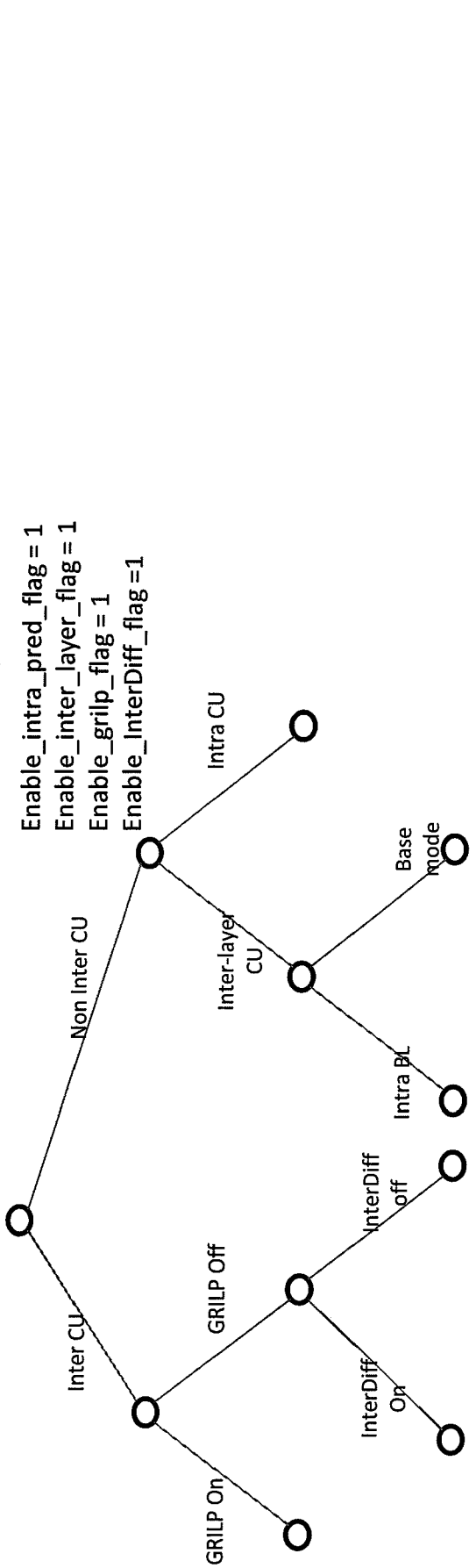


Fig. 48A

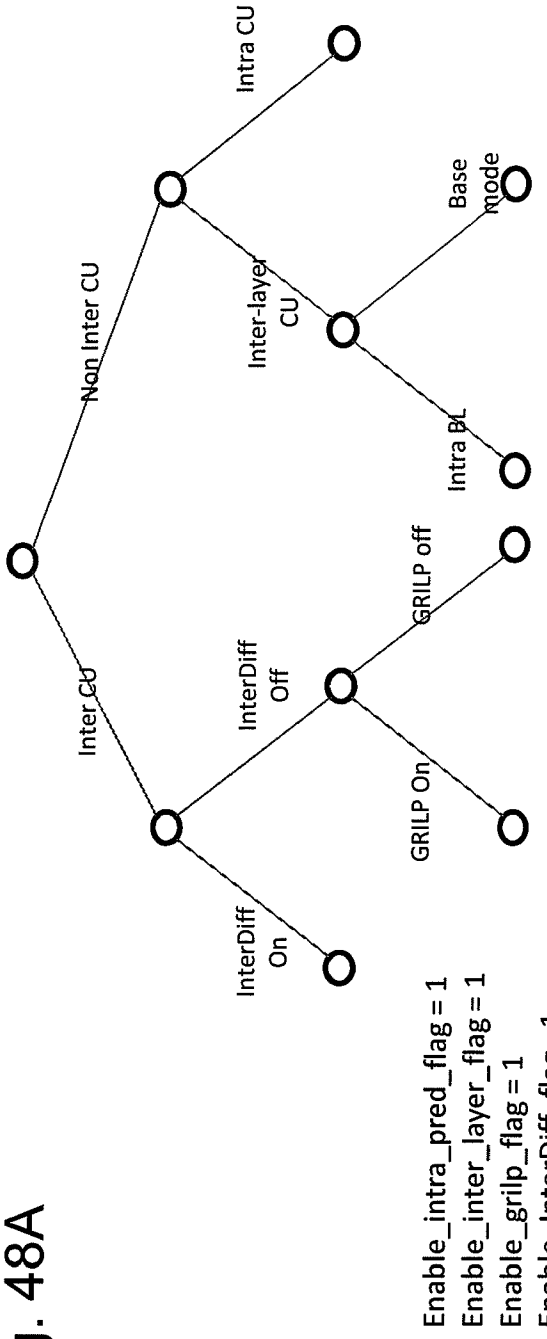


Fig. 48B

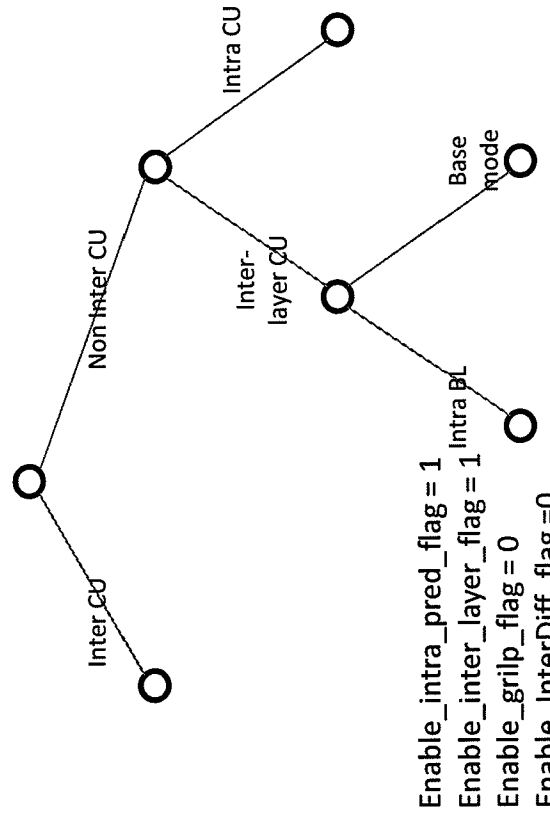


Fig. 48D

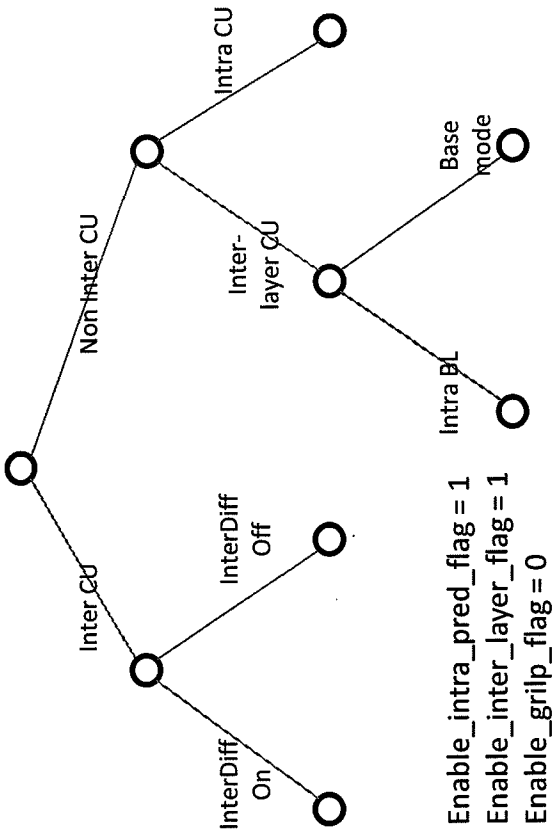


Fig. 48C

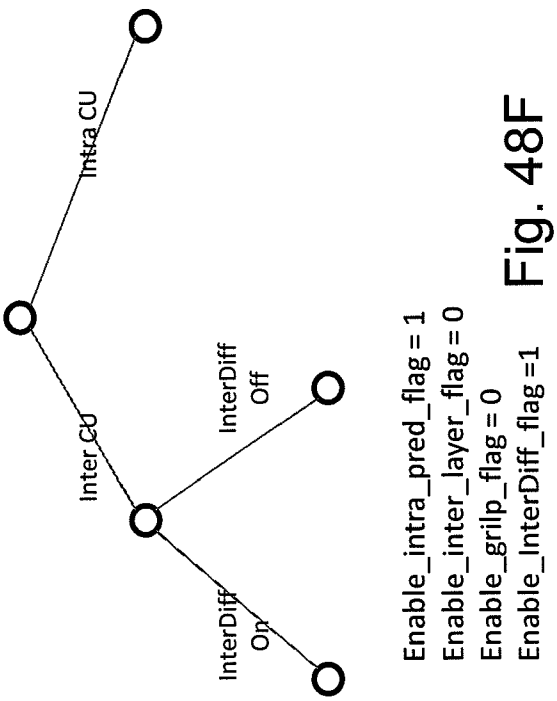


Fig. 48F

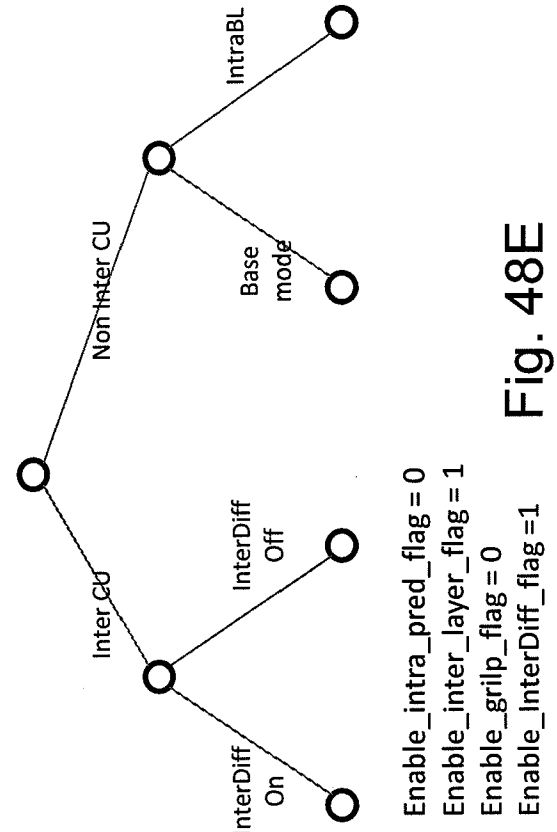


Fig. 48E

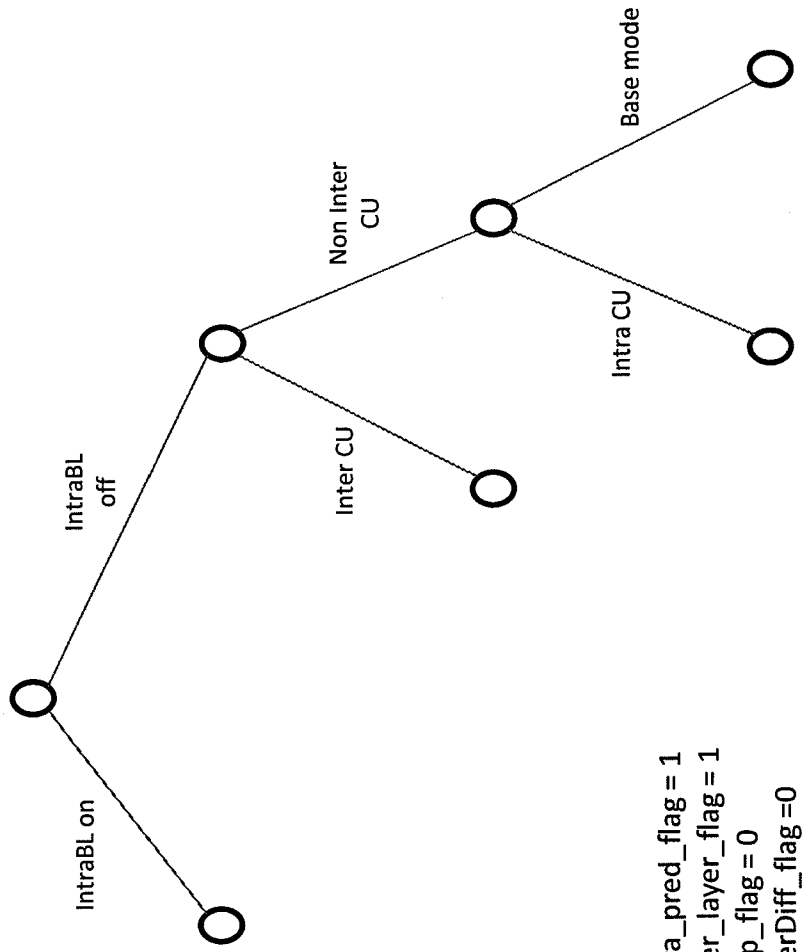

IntraBL on
IntraBL off
Inter CU
Non Inter CU
Intra CU
Base mode
Enable_intra_pred_flag = 1
Enable_inter_layer_flag = 1
Enable_grilp_flag = 0
Enable_InterDiff_flag =0

Fig. 48G