



(51) International Patent Classification:

H04N 21/854 (2011.01) *H04N 21/2343* (2011.01)
H04N 21/4728 (2011.01) *H04N 19/17* (2014.01)

(21) International Application Number:

PCT/EP2014/050704

(22) International Filing Date:

15 January 2014 (15.01.2014)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

1300950.1 18 January 2013 (18.01.2013) GB

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

(71) Applicant (*for AE only*): **CANON EUROPE LIMITED** [GB/GB]; 3 The Square, Stockley Park, Uxbridge, Middlesex UB11 1ET (GB).

(72) Inventors: **LE FLOCH, Hervé**; 10 Rue Monseigneur Duchesne, F-35000 Rennes (FR). **MAZE, Frédéric**; 6, rue des Tilleuls, F-35850 Langan (FR). **DENOUEAL, Franck**; 8 La Ville es Ray, F-35190 Saint Domineuc (FR).

(74) Agent: **SANTARELLI**; P.O. 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, IR, IS, JP, KE, KG, 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, SA, 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, KM, ML, MR, NE, SN, TD, TG).

Published:

- with international search report (Art. 21(3))
- before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))

(54) Title: METHOD OF DISPLAYING A REGION OF INTEREST IN A VIDEO STREAM

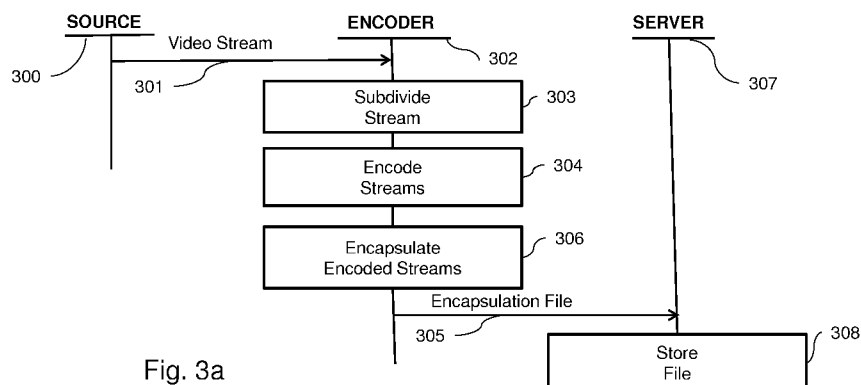


Fig. 3a

(57) Abstract: A method of processing video data comprising subdividing a video stream into image portions, encoding the image portions with high resolution in respective encoded video streams and reassembling a selection of encoded video streams in order to display a region of interest with high resolution. Embodiments of the invention provide high resolution display of specific region of interest without a priori knowledge of the position of the region of interest.

5 METHOD OF DISPLAYING A REGION OF INTEREST IN A VIDEO STREAM

The present invention relates to video data processing for video streaming.

10 In particular, the present invention relates to video data processing making it possible to display a portion of images of a video stream with a higher quality than the other portions of the images.

Video streaming is illustrated in **Figure 1**.

15 Audio data and video data are acquired during respective steps 100 and 101. The audio data are compressed during a step 102 using a compression algorithm such as MP3. In parallel, video data are compressed during a step 103 using algorithms such as MPEG4, AVC, HEVC, SVC. Once compression has been performed, audio and video elementary streams 104 and 105 are respectively obtained.

20 Next, during a step 106, the elementary streams are encapsulated into an encapsulation file as a global media presentation. For example, the ISO BMFF standard (acronym for ISO Base Media File format), or its extension to AVC, SVC, HEVC may be used for describing as the content of the encoded audio and video elementary streams a global media presentation. The
25 encapsulation file may contain:

- media data (the audio and video elementary streams), and
- metadata describing the media data.

The encapsulated file can be used directly for a local playback during step 107. It can also be streamed over a network during step 108. For example,
30 DASH (acronym for Dynamic Adaptive Streaming over HTTP) can be used as a streaming protocol.

A need exists for easily displaying with better quality spatial sub-parts of a video. This functionality is illustrated in **Figure 2**. This figure shows frames 200 of a video stream. For example, the video stream is the output of a camera sensor. The video stream is encoded during a step 201 and the encoded video stream is encapsulated during a step 202 into a container (encapsulation file).
5 The container is a file containing the elementary stream (media data) and a description of the elementary stream (metadata).

One solution to display a high quality spatial sub-part of a video on a end user display is to transmit the full video with quality to the end user receiver device. However, in case of UHD (Ultra High Definition) video, transmitting the
10 full video stream requires a high bitrate that may not be supported by the receiver displaying the video on the display.

The end user may though select a region of interest of the video stream (ROI) 203 and make request for receiving and displaying the ROI with a quality higher than the other parts of the video (the other parts being received and displayed with a basic quality).
15

For example, the ROI may be encoded with the same spatial resolution than the remainder of the video but with a higher quality. Also, the whole video may be spatially up-sampled with complementary details for improving the visual rendering of the ROI. However, due to the initial encoding
20 in step 201, implementation of this functionality is difficult.

Encoding the whole initial video stream with high quality and transcoding the video when the ROI is known necessitates performing highly complex computations.

25 Streaming the pixels of the ROI with a high quality, when the ROI is known in advance, requires data extraction from the encapsulation file, transcoding and a new encapsulation. These operations are complex and necessitate a large amount of processing resources. Also, having a ROI set once and for all in advance cannot give the user the opportunity to select
30 dynamically the ROI.

Thus, there is need for enhancing video streaming so as to make it possible to display a ROI of a video stream with a better quality.

A **first aspect of the invention** relates to a method of processing a video stream for encapsulation into an encapsulation file, the method comprising the following steps:

- obtaining a subdivided video stream, wherein each image of the video stream is subdivided into a plurality of image portions, and wherein the images of the video stream are subdivided according to a same subdivision grid,
- encoding the video stream as a plurality of encoded video streams, each encoded video stream comprising encoded images with at least one same respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid, and
- encapsulating the plurality of encoded video streams into an encapsulation file, each image portion of the encapsulated encoded video streams being associated with a resolution level data indicating a level of resolution with which said each image portion is encoded.

A method according to the first aspect make it possible to stream a ROI either predefined (i.e. already know during encoding) arbitrary defined by a end user.

Thus, any region of interest in the video stream can be reassembled in order to be displayed with high resolution. The region of interest can be unknown during encoding.

The encapsulation file provided makes it possible to simplify the processing needed for generating video data for enhanced display of a region of interest.

The resolution level data may comprise a superimposition data indicating a position of the data portion in a superimposition of image layers.

Therefore, reassembling of the region of interest may comprise superimposing frames of the encoded video streams of the plurality. According to the superimposition data, image portions with high resolution are displayed or not.

High resolution may be understood as high spatial resolution wherein the number of pixels of the high resolution image portions is higher than the

number of pixels in the low resolution image portions (for example VGA, HD, UHD or other kind of spatial resolution). High resolution may also be understood as high image quality without any difference in terms of number of pixels between high resolution image portions and low resolution image portions.

5 For example, each image portion of the encapsulated encoded video stream is associated with a location information indicating a position of the image portion in the subdivision grid.

 Therefore, the encoded video streams needed for displaying the region of interest with high resolution may be easily identified.

10 For example, said location information comprises:

- coordinates of a starting point of the at least one same respective image portion, and

- dimension information representing a dimension of said at least one same respective image portion.

15 Thus, encapsulation file format currently existing may be adapted for implementation of methods according to the first aspect.

 The method may further comprise encapsulating, into said encapsulation file, said subdivided video stream encoded with a low resolution.

 Thus, non-scalable encoding may be used.

20 For example, each one of the plurality of encoded video streams is encoded with a base layer with a low resolution and an enhancement layer with said images with at least one same respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid.

25 Thus, scalable encoding may be used.

 The base layer may be the same for the plurality of encoded video streams.

 A **second aspect of the invention** relates to a method of processing an encapsulation file for streaming video data, the method comprising the
30 following steps:

- obtaining a request for enhanced displaying of a region of interest in images of a video stream with a higher resolution than the remainder of said images,

- accessing an encapsulation file comprising encapsulated encoded video streams, encoded from a common subdivided video stream, wherein each image of each encoded video stream is subdivided into a plurality of image portions according to a same subdivision grid, wherein each encoded video stream comprises encoded images with at least one same respective image portion of said subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid, and wherein each image portion of the at least one encoded video stream is associated with a resolution level data indicating a level of resolution with which the image portion is encoded,

- determining at least one image portion corresponding to said region of interest, and

- transmitting at least one encapsulated encoded video stream corresponding to the at least one image portion determined.

The method according to the second aspect makes it possible to stream video data while outputting high resolution data on demand for requested regions of interest.

For example, said resolution level data comprises a superimposition data indicating a position of the data portion in a superimposition of image layers.

For example, each image portion of the encapsulated encoded video stream is associated with a location information indicating a position of the image portion in the subdivision grid.

For example, said location information comprises:

- coordinates of a starting point of the at least one same respective image portion, and

- dimension information representing a dimension of said at least one same respective image portion.

For example, there is encapsulated, into said encapsulation file, said common subdivided video stream encoded with a low resolution, the method

further comprising streaming said common subdivided video stream encoded with a low resolution.

For example, each one of the plurality of encoded video streams is encoded with a base layer with a low resolution and an enhancement layer with said images with at least one same respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid.

For example, said base layer is the same for the plurality of encoded video streams.

10 A **third aspect of the invention** relates to a method of processing video data for displaying a video stream, the method comprising the following steps:

 - transmitting to a streaming server, a request for enhanced displaying of a region of interest in images of a video stream with a higher resolution than the remainder of said images,

15 - receiving at least one encapsulation file comprising at least one encapsulated encoded video stream, wherein each image of each at least one encoded video stream is subdivided into a plurality of image portions according to a same subdivision grid, wherein each at least one encoded video stream comprises encoded images with at least one same respective image portion of said subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid, and wherein said at least one same respective image portion of said subdivision grid corresponds to said region of interest, each image portion of the at least one encoded video stream being associated with a resolution level data indicating a level of resolution with which said each image portion is encoded,

25 - decoding said at least one encapsulated encoded video stream received, and

 - generating, based on at least one video stream obtained by said decoding and the resolution level data associated with the image portions therein, a video signal, wherein resolution in said region of interest in the images is higher than in the remainder of the images.

Thus, regions of interest with high resolution may be requested on demand without necessitating complex data processing.

For example, said resolution level data comprises a superimposition data indicating a position of the data portion in a superimposition of image
5 layers.

For example, each image portion of the encapsulated encoded video stream is associated with a location information indicating a position of the image portion in the subdivision grid.

For example, a plurality of encoded video streams are received, the
10 method further comprising combining the plurality of video streams obtained by said decoding, said combining comprising assembling the at least one same respective image portion of said subdivision grid of each video stream thereby obtaining said region of interest.

For example, said at least one encoded video stream is encoded
15 from a common subdivided video stream, the method further comprising:

- receiving said common subdivided video stream encoded with a low resolution,
- decoding said encoded common subdivided video stream, and
- generating said video signal based at least on said decoded
20 common subdivided video stream.

For example, each one of the plurality of encoded video streams is encoded with a base layer with a low resolution and an enhancement layer with said images with at least one same respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the
25 subdivision grid, and wherein said decoding is performed based at least on said base layer and enhancement layer.

For example, said base layer is the same for the plurality of encoded video streams.

A **fourth aspect of the invention** relates to a device for
30 implementing a method according to the first aspect, such device may comprise means for implementing the steps of the method, such as a processing unit configured for executing said steps.

A **fifth aspect of the invention** relates to a device for implementing a method according to the second aspect. Such device may comprise means for implementing the steps of the method, such as a processing unit configured for executing said steps.

5 A **sixth aspect of the invention** relates to a device for implementing a method according to the third aspect. Such device may comprise means for implementing the steps of the method, such as a processing unit configured for executing said steps.

10 A **seventh aspect of the invention** relates to systems comprising at least one device according to the fourth, fifth and sixth aspects of the invention.

 An **eighth aspect of the invention** relates to computer programs and computer program products comprising instructions for implementing methods according to the first, second and/or third aspect(s) of the invention, when loaded and executed on computer means of a programmable apparatus
15 such as an encoding device, a server device and/or a client device.

 According to an embodiment, information storage means readable by a computer or a microprocessor store instructions of a computer program, that it makes it possible to implement a method according the first, second and/or third aspect of the invention.

20 The objects according to the second, third, fourth, fifth, sixth, seventh, and eighth aspects of the invention provide at least the same advantages as those provided by the method according the first aspect of the invention.

 Other features and advantages of the invention will become apparent
25 from the following description of non-limiting exemplary embodiments, with reference to the appended annexes and drawings, in which, in addition to Figures 1 and 2:

- **Figures 3a** and **3b** are a schematic illustration of a general context of implementation of embodiments,
- 30 - **Figures 4a** and **4b** are schematic illustrations of encoding and displaying according to embodiment,

- **Figures 5a** and **5b** are schematic illustrations of encapsulation with a non-scalable codec according to embodiments,
- **Figures 6a** and **6b** are schematic illustrations of encapsulation with a scalable codec according to embodiments,
- 5 - **Figure 6c** is a schematic illustration of segment files,
- **Figures 7** and **8** are illustrations of the use of a file format according to embodiments
- **Figure 9** is a schematic illustration of a device according to embodiments, and
- 10 - **Annexes A, B** and **C** illustrate file formats according to embodiments.

In what follows, a general context of implementation of embodiments of the invention is first presented. Next, more specific details of implementation are described.

15 **Figure 3a** is an illustration of the generation of an encapsulation file according to embodiments.

A source device 300 generates a video stream 301. The source device may be a video camera, a playback device or another kind of video source device. The video stream is received by an encoding device 302.

20 The encoding device subdivides the video stream received, according to a subdivision grid, during a step 303. Each image (or frame) of the video stream received is subdivided according to said same subdivision grid into image portions. Alternatively, the encoding device may receive a video stream already subdivided.

25 Next, during a step 304, a plurality of video streams are encoded, based on the subdivided video stream. In each encoded video stream, at least one image portion of the grid is encoded with a higher quality than the other image portions. In each frame of the same encoded video stream, the same image portion is encoded with higher quality.

30 Once the video stream received is encoded as a plurality of encoded video streams, the encoded video streams are encapsulated into an encapsulation file 305 during a step 306.

The encapsulation file is subsequently transmitted to a server device 307, in order to be stored during a step 308.

One or several devices presented with reference to Figure 3a may belong to a same device or system.

5 **Figure 3b** is an illustration of the use of the encapsulation file according to embodiments.

A client device 309, such as a display device, sends a video request 310 to the server device. The video request relates to the video stream 301 encoded by the encoding device 302.

10 The server device identifies the video stream and accesses the corresponding encapsulation file 305 during a step 311. The server device then starts streaming of the video by transmitting to the client device segment files 312.

15 Based on the segment files received from the server device, the client device decodes the video stream during step 313. A video signal is generated, during step 314, based on the decoding step, in order to display the video stream on a screen.

20 During display of the video stream, appears the need for having a region of interest in the video stream displayed with higher quality. For example, a user identifies such region of interest and requests display with high quality.

A request 315 is transmitted from the client device to the server device. The request comprises an identification of the region of interest.

25 Upon receipt of the request, the server device identifies the region of interest and, during a step 316, it accesses the encapsulation file in order to determine, during a step 317, image portions of the grid that correspond to the region of interest.

The encoded video streams corresponding to the image portions determined are then transmitted to the client device through segment files 318.

30 Upon receipt of the segment files, the client device extracts the encoded video streams from segment files and decodes them during a step 319 and combines the video streams during a step 320 in order to generate a video signal (step 321), wherein the region of interest is defined with higher quality

than the other parts of the images. The client device uses the video signal generated for displaying the video stream according to the request, i.e. with the region of interest displayed with high quality,

Encoding according to embodiments is described with reference to
5 **Figure 4a**. An initial video stream 400 is encoded and encapsulated in order to make it possible for the user to select a region of interest (ROI) in the video stream and have the ROI displayed with a higher quality than the remainder of the video stream.

Each image (or “frame” hereinafter) of the video stream 400 is
10 subdivided into image portions (or “tiles” hereinafter) 401. For example, in Figure 4a, each image is subdivided according to a rectangular grid of 2 by 4 squares. The grid has four upper tiles T1, T2, T3, T4 and four lower tiles T5, T6, T7, T8. The grid is common to the frames of the video stream. However, the embodiments of the invention are not limited to the grid presented in Figure 4a.
15 Other designs of the grid may be envisaged.

Once the initial video stream is subdivided according to the grid, a plurality of encoded video streams 402, 403 are generated. Each encoded video stream generated has in each frame an image portion encoded with a higher quality than the other image portions. For example, in encoded video
20 stream 402, the frames have tile T1 encoded with a higher quality than the other tiles T2-T8. In encoded video stream 403, the frames have tile T8 encoded with a higher quality than the other tiles T1-T7. A plurality of encoded video streams is obtained wherein each tile T1-T8 is encoded with high quality in at least one encoded video stream. Although this is not represented in Figure 4a, one, two
25 or more tiles may be encoded with high quality in a same encoded video stream.

The encoded video streams are thereafter encapsulated into an encapsulation file. The encapsulation file may be a media presentation having as many video tracks as encoded video streams.

30 Display and streaming according to embodiments is described with reference to **Figure 4b**. In the example illustrated in Figure 4b, the initial video

stream 400 has been subdivided into 16 tiles (numbered 1 to 16), according to a rectangular grid 404 of 4 by 4 rectangles.

An ROI 406 is defined, for example by a user, in order to have it displayed with a higher quality than the remainder of the video streams. In the example of Figure 4b, the ROI extends over four tiles (1, 2, 5 and 6). Thus, four
5 encoded video streams are selected wherein the tiles (1, 2, 5 and 6) are (respectively) encoded with high quality.

The selected encoded video streams are then decoded and combined to display frames wherein the ROI has a higher resolution than the
10 remainder of the frame.

In what follows, encapsulation is described with more details with reference to **Figure 5a** and **Figure 5b**. An initial non-scalable video stream 500 is considered. The initial video stream is subdivided into four tiles “a”, “b”, “c” and “d”, according to a 2 by 2 rectangles grid 501.

15 During encoding 502, five encoded video streams (“elementary streams” hereinafter) 503, 504, 505, 506 and 507 are generated from the initial video stream. Each tile of the grid has been encoded with a higher quality in a respective encoded video stream. Since the grid has four tiles and five elementary streams are generated, one of the elementary streams (503) is
20 wholly encoded with low quality.

Elementary streams are illustrated in more details in Figure 5b. All tiles “a”, “b”, “c” and “d” in elementary stream 503 are encoded with basic or low quality (LQ). In elementary stream 504, tile “a” is encoded with high quality (HD), whereas the other tiles have been encoded with low quality. In elementary
25 stream 505 (not represented), tile “b” is encoded with high quality (HD), whereas the other tiles have been encoded with low quality. In elementary stream 506 (not represented), tile “c” is encoded with high quality (HD), whereas the other tiles have been encoded with low quality. In elementary stream 507, tile “d” is encoded with high quality (HD), whereas the other tiles
30 have been encoded with low quality.

Once all the elementary streams are generated, they are encapsulated into an encapsulation file 508 (“encapsulation format” hereinafter).

ISO Base Media File Format and its extensions may be used. However, other formats may be used.

The encapsulation file comprises several video tracks respectively corresponding to the generated elementary streams. The file is a media presentation. Video tracks 509 ("Track 0"), 510 ("Track 1"), 511 ("Track 2"), 512 ("Track 3"), 512 ("Track 4") respectively correspond to encapsulated elementary streams 503, 504, 505, 506, 507.

The ISO BMFF and the extension for DASH make it possible to put each track in different segment files. In such case, an initialization segment is generated. DASH is defined in documents "*ISO/IEC 23009-1, Dynamic adaptive streaming over HTTP (DASH), Part1: Media presentation description and segment formats*", "*ISO/IEC 14496-12, Information technology – Coding of audio-visual objects – Part 12: ISO base media file format*" and "*ISO/IEC 14496-12:2008/FPDAM 3 & ISO/IEC 14496-12:2008/DAM 3 – Coding of audio-visual objects – Part 12: ISO base media file format, AMENDMENT 3: DASH support and RTP*".

The initialization segment contains data defining and initializing the tracks. The initialization segment is associated with segment files. Each video track may be put in a respective segment file. Therefore, each track may be streamed independently. Based on these segment files, only the video tracks (and thus the segment files) useful for the end user can be sent.

Encapsulation is described with reference to **Figure 6a** and **Figure 6b**, with an initial video stream 600. The encapsulation is similar to the encapsulation described with reference to Figures 5a and 5b, but the video codec used is a scalable video codec. For example, the SVC video codec may be used.

The initial video stream is subdivided into four tiles "a", "b", "c" and "d", according to a 2 by 2 rectangles grid 601.

During encoding 602, four encoded video streams ("elementary streams" hereinafter) 603, 604, 605, and 606 are generated from the initial video stream. Each tile of the grid has been encoded with a higher quality in a respective encoded video stream. Since a scalable video codec is used, each

elementary stream contains NAL units (acronym for Network Abstraction Layer) corresponding to the base layer and NAL units corresponding to the enhancement layer.

Elementary streams are illustrated in more details in Figure 6b. The enhancement layers of the elementary streams differ from one another because each one contains a different respective tile encoded with high quality tile. In elementary stream 603, tile "a" is encoded with high quality (HD), whereas the other tiles have been encoded with low quality. In elementary stream 604, tile "b" is encoded with high quality (HD), whereas the other tiles have been encoded with low quality. In elementary stream 605 (not represented), tile "c" is encoded with high quality (HD), whereas the other tiles have been encoded with low quality. In elementary stream 606, tile "d" is encoded with high quality (HD), whereas the other tiles have been encoded with low quality.

The elementary streams 603, 604, 606 comprise respective base layers 607, 608, 609. Elementary stream 605 (not represented) also comprises a base layer. The base layers are the same for all the elementary streams.

Once the elementary streams have been generated, they are encapsulated into an encapsulation file 610 ("file format" hereinafter). The ISO Base Media File Format and its extensions may be used. However, other formats may be used.

The encapsulation file comprises several video tracks respectively corresponding to the generated elementary streams. The encapsulation file also comprises a video track corresponding to the base layer. Since all the elementary streams share the same base layer, it is possible to create a video track containing the NAL units of the base layer. The file is a media presentation. Video tracks 611 ("Track 0") correspond to the base layer and video tracks 612 ("Track 1"), 613 ("Track 2"), 614 ("Track 3"), 615 ("Track 4") respectively correspond to encapsulated elementary streams -603, 604, 605, 606. In order to reduce redundancy, extractors (NALU) may be used. Extractors are specific NAL units that make it possible to point to other NALU in other external tracks. Thus, the video tracks (612, 613, 614 and 615) may be video

tracks containing extractors pointing to track 611. The extractors are replaced during de-encapsulation by the NAL units of the base layer.

The ISO BMFF and the extension for DASH make it possible to put each track in different segment files. In such case, an initialization segment is generated. The initialization segment contains data defining and initializing the tracks. The initialization segment is associated with segment files. Each video track may be put in a respective segment file. Therefore, each track may be streamed independently. Based on these segment files, only the video tracks (and thus the segment files) useful for the end user can be sent.

The segment files (or media segments) are described with reference to Figure 6c. The ISO BMFF and the extensions for DASH make it possible to split a media presentation into autonomous fragments. Each fragment corresponds to a respective period of time. A fragment comprises at least a "movie fragment box" (moof) and a "media data box" (not represented in the figure). The media data box contains the elementary stream corresponding to the period of time of the fragment. The movie fragment box contains the metadata data corresponding to the elementary stream. Fragments corresponding to a same track can be grouped together in a same media segment (or segment file). This is illustrated in Figure 6c. Two tracks are defined. The first track is a video track with the "track_ID" data equal to 0x01 (with two representations), the second track is an audio track with "track_ID" equal to 0x02.

The two tracks are initially defined in an initialization segment 650. For example, the initialization segment contains a definition of each track (track box, track header box etc.) and the composition information of the different tracks (still in the track boxes). A set of segment files 651, 652, 653 and 654 can be defined. Media segment 651 contains fragments corresponding to the first track. Media segment 652 contains fragments related to the same first track. These fragments correspond to a different period of time. Media segment 653 contains fragments related to the second track. Media segment 654 contains fragments related to the same second track. Fragment 653 corresponds to a period of time different from the one associated with fragment

654. These media segments can be streamed separately and concatenated together with an initialization segment. The resulting media presentation is compatible with the ISO BMFF file format standard.

When using the ISO BMFF reference file format, modifications of the
5 file format may be envisaged. Modifications are presented with reference to Annex A, Annex B and Annex C.

Annex A is the code for the track header box ('tkhd') in the current version of the file format defined in document "ISO/IEC 14496-12, Information technology – Coding of audio-visual objects – Part 12: ISO base media file
10 format". The file format is an encapsulation format that describes the elementary streams of the tracks comprised in a media presentation. The file format has tools for composing the tracks. The track box contains several other boxes. One of the boxes is the track header box ('tkhd'). The "layer" attribute specifies the front-to-back ordering of the video tracks. Tracks with lower
15 numbers are displayed in front. "0" is the default value, and "-1" correspond to images in front, and so on.

The "width" and "height" attributes specify the track's visual presentation size as fixed-point 16.16 values. These values need not to be the same as the pixel dimensions of the images, which is documented in the
20 sample description(s). All images in the sequence are scaled to this size, before any overall transformation of the track represented by the matrix. The pixel dimensions of the images are the default values.

Annex B shows the attributes that may be added into the track header box. Embodiments of the invention are not limited to names indicated in
25 Annex B. Any other names could be used. The attributes make it possible to define spatial sub-parts of frames of video tracks as layers, once the decoding of the frame is performed. Thus, it may be possible to combine together spatial sub-parts of each frame of decoded video tracks. Since the attributes added are located in the track header box, they will be applied for each decoded frame of
30 the corresponding video track.

The attribute "nb_tiles" specifies the number of spatial sub-parts that are defined in the track. For each sub-parts, the following attributes may be applied:

5 - "(horizontal_origin, vertical_origin)" specifies the track's visual presentation origin of the current spatial sub-part (left and up corner of the rectangular track). The units of these two parameters are the same as the one used for the (height, width) attributes of the track header.

10 - "width" and "height" specify the rectangular spatial sub-part visual presentation size as fixed-point 16.16 values. The units of these two parameters are the same as the one used for the (height, width) attributes of the track header.

 - "tile_layer" specifies the front-to-back ordering of the current spatial sub-part. Video sub-parts with lower numbers are displayed in front. "0" is the default value, and "-1" correspond to images in front, and so on.

15 If the proposed new attributes are not used, the default 'layer' value indicated in the track header is used for each frame of the video.

Annex C illustrates embodiments wherein the tile and layering information are not located in one given box of the file format, as in Annex B wherein the information is put in the track header box, but rather spread over
20 several boxes.

 For example, in code C1, the movie header box ('mvhd') and the track header box ('tkhd') are modified compared to the 'mvhd' and 'tkhd' boxes in document "ISO/IEC 14496-12, Information technology – Coding of audio-visual objects – Part 12: ISO base media file format".

25 In the movie header box of code C2, the tile information is introduced from two parameters: "tile_width" and "tile_height". These two parameters define the horizontal and vertical sizes of a tile. These parameters are used for all the video tracks existing in the movie presentation. For example, with reference to the Figure 4a, the parameters (tile_width, tile_height) are used for
30 indicating that the video is partitioned into non-overlapping tiles and that the size of each tile is given by (tile_width, tile_height). Once the tiles are defined, they are indexed from the left up to right bottom (1 to 8 in Figure 4a). If the

parameters (tile_width, tile_height) are set to "0", it is interpreted as no tile is defined.

In the track header box of Code C1, new attributes for the track header box are presented:

- 5 - "nb_tiles": specifies the number of spatial sub-parts defined in the track. The value of this parameter is the same as the number of tiles defined by the (tile_width, tile_height) attributes of the movie header box.

Each sub-part has the following attributes:

- 10 - "Tile_reference": is an index pointing to one of the tiles defined by the (tile_width, tile_height) attributes of the movie header box.
- "tile_layer": specifies the front-to-back ordering of the spatial sub-part (spatial sub-parts with lower numbers are on top of the superimposed layers: "0" is the default value, "-1" is the value of the attribute of sub-parts in front of tracks having the attribute set to "0", and so on.
- 15

With reference to **Figure 7**, definition of the values of the attributes according to embodiments is described. Figure 7 more particularly focuses on the server side.

- 20 An exemplary initial video is subdivided into 4 tiles ("a", "b", "c" and "d"). Four elementary streams (700, 701, 702 and 703) are generated. For example, a scalable codec is used. The enhancement layers of the elementary streams differ from one another because each one contains a different respective tile encoded with high quality tile. In elementary stream 700, tile "a" is
- 25 encoded with high quality (HD), whereas the other tiles have been encoded with low quality. In elementary stream 701, tile "b" is encoded with high quality (HD), whereas the other tiles have been encoded with low quality. In elementary stream 702 (not represented), tile "c" is encoded with high quality (HD), whereas the other tiles have been encoded with low quality. In elementary
- 30 stream 703, tile "d" is encoded with high quality (HD), whereas the other tiles have been encoded with low quality.

The elementary streams 700, 701, 702, 703 each comprise respective base layers. The base layers are the same for all the elementary streams.

Once the elementary streams have been generated, they are
5 encapsulated into an encapsulation file 704 (or "file format"). The ISO Base Media File Format and its extensions may be used. However, other formats may be used.

The encapsulation file comprises several video tracks respectively corresponding to the generated elementary streams. Video track 705 ("Track 0")
10 corresponds to the base layer and video tracks 706 ("Track 1"), 707 ("Track 2"), 708 ("Track 3"), 709 ("Track 4") respectively correspond to encapsulated elementary streams 700, 701, 702, 703.

During encapsulation, the boxes presented hereinabove, with reference to Annexes A, B and C are filled in. Since Track 0 corresponds to the
15 base layer, there is no high quality spatial sub-part defined. The high quality sub-parts are however defined for the other tracks. For the sake of conciseness, the values given to the attributes are shown on Figure 7 only for Track 1 and Track 2.

Concerning Track 1:

- 20 - the default layer value for the whole video (e.g. for each frame of the video) is "0",
- "Nb_tiles" is set to "1" because the video track has one tile defined with high quality (tile a)
 - the location of this high quality tile is defined by:
25
 - its origin (0,0)
 - its size (100,100)
 - "Tile_Layer" is set to "-1", which means that the high quality layer should be placed on top of the superimposed video tracks during combination for display.

Concerning Track 2:

- 30 - the default layer value for the whole video (e.g. for each frame of the video) is "0",

- "Nb_tiles" is set to "1" because the video track has one tile defined with high quality (tile b),
- the location of the high quality tile is defined by:

- its origin (100,0)
- 5 ○ its size (100,100)

- "Tile_Layer" is set to "-1", which means that the high quality layer should be placed on top of the superimposed video tracks during combination for display.

The video tracks are the video tracks of a same media presentation. They can be included in a same global file or they can be split into independent
10 segment files. In the case of independent segment files, an initialization segment is generated. This initialization segment contains the track header boxes.

With reference to **Figure 8**, the use of the values of the attributes according to embodiments is described. Figure 8 more particularly focuses on
15 the client side.

In the example shown in Figure 8, it is supposed that an ROI, (which may be selected by a user) is included in tile 'a' and tile 'c' of the initial video stream of Figure 7. In order to display the ROI with high quality, three video tracks are sent from the server to the client:

- 20 - Track 0 (705) which contains the base layer,
- Track 1 (706) which contains high quality tile a,
- Track 3 (708) which contains high quality tile c.

Tracks 1 and 3 both use Track 0 through extractors, which make it possible to save bandwidth and memory space.

25 Independent streaming of each video track is made possible by definition of segment files (one initialization segment introduces the different video tracks and segment files containing the encapsulated elementary streams).

Once the three tracks are received, elementary streams contained in
30 Tracks 1 and 3 are extracted (step 803), using extractors pointing to Track 0. 'stbl' boxes within the track boxes (or 'trun' boxes) store composition or decoding times used for extracting Tracks 1 and 3.

After elementary stream extraction (frame by frame), the elementary stream is decoded (step 804). Decoded frames 807 and 808, respectively belonging to Track 1 and Track 3 are shown in Figure 8.

Once the frames are decoded, they are combined together based on the layering information ("Tile_Layer" attribute). The track header layering is read during step 805. For example, reading of the layering can be done once and for all as soon as the video tracks are received (for example when the initialization file is received).

The layering 810 of Track 1 has the layer value set to "-1" for tile "a", and the layer value set to "0" for the other tiles. The layering 809 of Track 3 has the layer value set to "-1" for tile "c", and the layer value set to "0" for the other tiles. Based on these layering maps, the pixels of frames 807, 808 are combined into a frame 811.

Since tile «a» has the "Tile_Layer" attribute set to "-1" in frame 810 and set to "0" in frame 809, tile «a» from frame 810 is shown in front. Since tile «c» has the "Tile_Layer" attribute set to "-1" in frame 809 and set to "0" in frame 810, tile «c» from frame 809 is shown in front. Therefore, Frame 811 has pixels in tiles "a" and "c" coming from a video stream encoded with high quality.

The ROI being included in tiles "a" and "c", it can be displayed with high quality.

Figure 9 is a schematic block diagram of a computing device 900 for implementation of one or more embodiments of the invention. The computing device 900 may be a device such as a micro-computer, a workstation or a portable device. The computing device 900 comprises a communication bus connected to:

- a central processing unit 901 (CPU), such as a microprocessor;
- a random access memory 902 (RAM), for storing the executable code of methods according to embodiments of the invention and/or register for variables and parameters used for implementation of the methods;
- a read only memory 903 (ROM), for storing computer programs for implementing embodiments of the invention;

- a network interface 904 connected to a communication network over which digital data to be processed are transmitted or received. The network interface 904 can be a single network interface, or composed of a set of different network interfaces (for instance wired and/or wireless interfaces.

5 Data transmission may be controlled by a software application executed by the CPU;

- a user interface 905 for receiving inputs from a user or to display information to a user;

- a hard disk 906 (HD);

10 - an I/O module 907 for receiving/sending data from/to external devices such as a video source or display.

The executable code may be stored either in read only memory 903, on the hard disk 906 or on a removable digital medium such as disk. The executable code of the programs may also be received by means of a communication network, via the network interface 904, in order to be stored in one of the storage means of the communication device 900, such as the hard disk 906, before being executed.

The central processing unit 901 is configured for controlling execution of the instructions or portions of software code of the program or programs according to embodiments of the invention, which instructions are stored in one of the aforementioned storage means. The CPU 901 may execute instructions from the RAM memory after the instructions have been loaded from the ROM memory or the hard-disc (HD) for example. Such software application, when executed by the CPU 901, causes the steps of methods according to embodiments.

25 A computer program according to embodiments may be designed based on the flowcharts of Figures 3a, 3b, Annexes A, B, C and the present description.

Such computer program may be stored in a ROM memory of a system or device as described with reference to Figure 9. It may be loaded into and executed by a processor of such device for implementing steps of a method according to the invention.

Embodiments of the inventions may also be implemented in hardware (for example, in the form of an Application Specific Integrated Circuit or ASIC).

While the invention has been illustrated and described in detail in the
5 drawings and foregoing description, such illustration and description are to be considered illustrative or exemplary and not restrictive, the invention being not restricted to the disclosed embodiment. Other variations to the disclosed
embodiment can be understood and effected by those skilled in the art in practicing the claimed invention, from a study of the drawings, the disclosure
10 and the appended claims.

In the claims, the word “comprising” does not exclude other elements or steps, and the indefinite article “a” or “an” does not exclude a plurality. A single processor or other unit may fulfill the functions of several items recited in the claims. The mere fact that different features are recited in mutually different
15 dependent claims does not indicate that a combination of these features cannot be advantageously used. Any reference signs in the claims should not be construed as limiting the scope of the invention.

ANNEX A

```
aligned(8) class TrackHeaderBox extends FullBox('tkhd', version, flags){
    if (version==1) {
5      unsigned int(64) creation_time;
        unsigned int(64) modification_time;
        unsigned int(32) track_ID;
        const unsigned int(32) reserved = 0;
        unsigned int(64) duration;
10    } else { // version==0
        unsigned int(32) creation_time;
        unsigned int(32) modification_time;
        unsigned int(32) track_ID;
        const unsigned int(32) reserved = 0;
15      unsigned int(32) duration;
    }
    const unsigned int(32)[2] reserved = 0;
    template int(16) layer = 0;
    template int(16) alternate_group = 0;
20    template int(16) volume = {if track_is_audio 0x0100 else 0};
    const unsigned int(16) reserved = 0;
    template int(32)[9] matrix= {
0x00010000,0,0,0,0x00010000,0,0,0,0x40000000 }; // unity matrix
    unsigned int(32) width;
25    unsigned int(32) height;
}
```

ANNEX B

```

aligned(8) class TrackHeaderBox extends FullBox('tkhd', version, flags){
  if (version==1) {
    5      unsigned int(64) creation_time;
        unsigned int(64) modification_time;
        unsigned int(32) track_ID;
        const unsigned int(32) reserved = 0;
        unsigned int(64) duration;
  } else { // version==0
    10      unsigned int(32) creation_time;
        unsigned int(32) modification_time;
        unsigned int(32) track_ID;
        const unsigned int(32) reserved = 0;
        unsigned int(32) duration;
    15  }
    const unsigned int(32)[2] reserved = 0;
    template int(16) layer = 0;
    template int(16) alternate_group = 0;
    template int(16) volume = {if track_is_audio 0x0100 else 0};
    20  const unsigned int(16) reserved = 0;
    template int(32)[9] matrix=
        { 0x00010000,0,0,0,0x00010000,0,0,0,0x40000000 };
    // unity matrix
    unsigned int(32) width;
    25  unsigned int(32) height;

    unsigned int nb_tiles;
    for (unsigned int i= 1 to nb_tiles;i++){
    30      unsigned int(32) horizontal_origin;
        unsigned int(32) vertical_origin;
        unsigned int(32) width;
        unsigned int(32) height;
        template int(16) tile_layer = 0;
    35  }
}

```

ANNEX C**Code C1**

```

aligned(8) class TrackHeaderBox
extends FullBox('tkhd', version, flags){
5  if (version==1) {
        unsigned int(64) creation_time;
        unsigned int(64) modification_time;
        unsigned int(32) track_ID;
        const unsigned int(32) reserved = 0;
10     unsigned int(64) duration;
    } else { // version==0
        unsigned int(32) creation_time;
        unsigned int(32) modification_time;
        unsigned int(32) track_ID;
15     const unsigned int(32) reserved = 0;
        unsigned int(32) duration;
    }
    const unsigned int(32)[2] reserved = 0;
    template int(16) layer = 0;
20    template int(16) alternate_group = 0;
    template int(16) volume = {if track_is_audio 0x0100 else 0};
    const unsigned int(16) reserved = 0;
    template int(32)[9] matrix=
    { 0x00010000,0,0,0,0x00010000,0,0,0,0x40000000 };
25    // unity matrix
    unsigned int(32) width;
    unsigned int(32) height;

    unsigned int nb_tiles;
30    for (unsigned int i=1 to nb_tiles;i++){
        unsigned int tile_reference;
        template int(16) tile_layer;
    }
    }
35

```

Code C2

```
aligned(8) class MovieHeaderBox extends FullBox('mvhd', version, 0) {
  if (version==1) {
    unsigned int(64) creation_time;
5    unsigned int(64) modification_time;
    unsigned int(32) timescale;
    unsigned int(64) duration;
  } else { // version==0
    unsigned int(32) creation_time;
10   unsigned int(32) modification_time;
    unsigned int(32) timescale;
    unsigned int(32) duration;
  }
  template int(32) rate = 0x00010000; // typically 1.0
15   template int(16) volume = 0x0100; // typically, full volume
  const bit(16) reserved = 0;
  const unsigned int(32)[2] reserved = 0;
  template int(32)[9] matrix =
  { 0x00010000,0,0,0,0x00010000,0,0,0,0x40000000 };
20   // Unity matrix
  bit(32)[6] pre_defined = 0;
  unsigned int(32) next_track_ID;

  unsigned int tile_width;
25   unsigned int tile_height;
}
```

CLAIMS

1. A method of processing a video stream for encapsulation into an encapsulation file, the method comprising the following steps:

- 5 - obtaining a subdivided video stream, wherein each image of the video stream is subdivided into a plurality of image portions according to a subdivision grid,
- encoding the video stream as a plurality of encoded video streams, each encoded video stream comprising encoded images with at least one
- 10 corresponding respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid, and
- encapsulating the plurality of encoded video streams into an encapsulation file.

- 15 2. A method according to claim 1, wherein each image portion of the encapsulated encoded video streams being associated with a resolution level data indicating a level of resolution with which said each image portion is encoded and said resolution level data comprises a superimposition data indicating a position of the data portion in a superimposition of image layers.

20

3. A method according to any one of claims 1 and 2, wherein each image portion of the encapsulated encoded video stream is associated with a location information indicating a position of the image portion in the subdivision grid.

25

4. A method according to claim 3, wherein said location information comprises:

- coordinates of a starting point of the at least one same respective image portion, and
- 30 - dimension information representing a dimension of said at least one same respective image portion.

5. A method according to any one of the preceding claims, further comprising encapsulating, into said encapsulation file, said subdivided video stream encoded with a low resolution.

5 6. A method according to any one of claims 1 to 4, wherein each one of the plurality of encoded video streams is encoded with a base layer with a low resolution and an enhancement layer with said images with at least one same respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid.

10

7. A method according to claim 6, wherein said base layer is the same for the plurality of encoded video streams.

8. A method of processing an encapsulation file for streaming video data, the method comprising the following steps:

15

- obtaining a request for enhanced displaying of a region of interest in images of a video stream with a higher resolution than the remainder of said images,

20

- accessing an encapsulation file comprising encapsulated encoded video streams, encoded from a common subdivided video stream, wherein each image of each encoded video stream is subdivided into a plurality of image portions according to a subdivision grid, wherein each encoded video stream comprises encoded images with at least one corresponding respective image portion of said subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid,

25

- determining at least one image portion corresponding to said region of interest, and

- transmitting at least one encapsulated encoded video stream corresponding to the at least one image portion determined.

30

9. A method according to claim 8, wherein wherein each image portion of the at least one encoded video stream is associated with a resolution

level data indicating a level of resolution with which the image portion is encoded, and wherein said resolution level data comprises a superimposition data indicating a position of the data portion in a superimposition of image layers.

5

10. A method according to any one of claims 8 and 9, wherein each image portion of the encapsulated encoded video stream is associated with a location information indicating a position of the image portion in the subdivision grid.

10

11. A method according to claim 10, wherein said location information comprises:

- coordinates of a starting point of the at least one same respective image portion, and

15

- dimension information representing a dimension of said at least one same respective image portion.

12. A method according to any one of claims 8 to 11, wherein there is encapsulated, into said encapsulation file, said common subdivided video stream encoded with a low resolution, the method further comprising streaming said common subdivided video stream encoded with a low resolution.

20

13. A method according to any one of claims 8 to 11, wherein each one of the plurality of encoded video streams is encoded with a base layer with a low resolution and an enhancement layer with said images with at least one same respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid.

25

14. A method according to claim 13, wherein said base layer is the same for the plurality of encoded video streams.

30

15. A method of processing video data for displaying a video stream, the method comprising the following steps:

- transmitting to a streaming server, a request for enhanced displaying of a region of interest in images of a video stream with a higher resolution than the remainder of said images, 5
- receiving at least one encapsulation file comprising at least one encapsulated encoded video stream, wherein each image of each at least one encoded video stream is subdivided into a plurality of image portions according to a subdivision grid, wherein each at least one encoded video stream 10 comprises encoded images with at least one corresponding respective image portion of said subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid, and wherein said at least one corresponding respective image portion of said subdivision grid corresponds to said region of interest,
- decoding said at least one encapsulated encoded video stream received, and 15
- generating, based on at least one video stream obtained by said decoding and the resolution level data associated with the image portions therein, a video signal, wherein resolution in said region of interest in the 20 images is higher than in the remainder of the images.

16. A method according to claim 15, each image portion of the at least one encoded video stream being associated with a resolution level data indicating a level of resolution with which said each image portion is encoded, 25 and wherein said resolution level data comprises a superimposition data indicating a position of the data portion in a superimposition of image layers.

17. A method according to any one of claims 15 and 16, wherein each image portion of the encapsulated encoded video stream is associated 30 with a location information indicating a position of the image portion in the subdivision grid.

18. A method according to any one of claims 15 to 17, wherein a plurality of encoded video streams are received, the method further comprising combining the plurality of video streams obtained by said decoding, said combining comprising assembling the at least one same respective image
5 portion of said subdivision grid of each video stream thereby obtaining said region of interest.

19. A method according to any one of claims 15 to 18, wherein said at least one encoded video stream is encoded from a common subdivided video
10 stream, the method further comprising:

- receiving said common subdivided video stream encoded with a low resolution,
- decoding said encoded common subdivided video stream, and
- generating said video signal based at least on said decoded
15 common subdivided video stream.

20. A method according to any one of claims 15 to 18, wherein each one of the plurality of encoded video streams is encoded with a base layer with a low resolution and an enhancement layer with said images with at least one
20 same respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid, and wherein said decoding is performed based at least on said base layer and enhancement layer.

21. A method according to claim 20, wherein said base layer is the same for the plurality of encoded video streams.
25

22. An image processing device for processing a video stream for encapsulation into an encapsulation file, the device comprising a processing
30 unit configured for:

- obtaining a subdivided video stream, wherein each image of the video stream is subdivided into a plurality of image portions, and wherein the images of the video stream are subdivided according to a subdivision grid,

- encoding the video stream as a plurality of encoded video streams,
5 each encoded video stream comprising encoded images with at least one corresponding respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid, and

- encapsulating the plurality of encoded video streams into an encapsulation file.

10

23. A device according to claim 22, wherein each image portion of the encapsulated encoded video streams being associated with a resolution level data indicating a level of resolution with which said each image portion is encoded and wherein said resolution level data comprises a superimposition
15 data indicating a position of the data portion in a superimposition of image layers.

24. A device according to any one of claims 22 and 23, wherein each image portion of the encapsulated encoded video stream is associated with a
20 location information indicating a position of the image portion in the subdivision grid.

25. A device according to claim 24, wherein said location information comprises:
25 - coordinates of a starting point of the at least one same respective image portion, and
- dimension information representing a dimension of said at least one same respective image portion.

30 26. A device according to any one of claims 22 to 25, the processing unit being further configured for encapsulating, into said encapsulation file, said subdivided video stream encoded with a low resolution.

27. A device according to any one of claims 22 to 25, wherein each one of the plurality of encoded video streams is encoded with a base layer with a low resolution and an enhancement layer with said images with at least one
5 same respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid.

28. A device according to claim 27, wherein said base layer is the same for the plurality of encoded video streams.
10

29. An image processing device for processing an encapsulation file for streaming video data, the device comprising a processing unit configured for:

- obtaining a request for enhanced displaying of a region of interest in
15 images of a video stream with a higher resolution than the remainder of said images,

- accessing an encapsulation file comprising encapsulated encoded video streams, encoded from a common subdivided video stream, wherein each image of each encoded video stream is subdivided into a plurality of image
20 portions according to a subdivision grid, wherein each encoded video stream comprises encoded images with at least one corresponding respective image portion of said subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid,

- determining at least one image portion corresponding to said region
25 of interest, and

- transmitting at least one encapsulated encoded video stream corresponding to the at least one image portion determined.

30. A device according to claim 29, wherein each image portion of
30 the at least one encoded video stream is associated with a resolution level data indicating a level of resolution with which the image portion is encoded and

wherein said resolution level data comprises a superimposition data indicating a position of the data portion in a superimposition of image layers.

31. A device according to any one of claims 29 and 30, wherein each
5 image portion of the encapsulated encoded video stream is associated with a location information indicating a position of the image portion in the subdivision grid.

32. A device according to claim 31, wherein said location information
10 comprises:
- coordinates of a starting point of the at least one same respective image portion, and
- dimension information representing a dimension of said at least one same respective image portion.

15 33. A device according to any one of claims 29 to 32, wherein there is encapsulated, into said encapsulation file, said common subdivided video stream encoded with a low resolution, the device further comprising streaming said common subdivided video stream encoded with a low resolution.

20 34. A device according to any one of claims 29 to 32, wherein each one of the plurality of encoded video streams is encoded with a base layer with a low resolution and an enhancement layer with said images with at least one same respective image portion of the subdivision grid encoded with a higher
25 resolution than the other image portions of the subdivision grid.

35. A device according to claim 34, wherein said base layer is the same for the plurality of encoded video streams.

30 36. An image processing device for processing video data for displaying a video stream, the device comprising a processing unit configured for:

- transmitting to a streaming server, a request for enhanced displaying of a region of interest in images of a video stream with a higher resolution than the remainder of said images,

- receiving at least one encapsulation file comprising at least one
5 encapsulated encoded video stream, wherein each image of each at least one encoded video stream is subdivided into a plurality of image portions according to a subdivision grid, wherein each at least one encoded video stream comprises encoded images with at least one corresponding respective image
10 image portions of the subdivision grid, and wherein said at least one corresponding respective image portion of said subdivision grid corresponds to said region of interest,

- decoding said at least one encapsulated encoded video stream received, and

15 - generating, based on at least one video stream obtained by said decoding and the resolution level data associated with the image portions therein, a video signal, wherein resolution in said region of interest in the images is higher than in the remainder of the images.

20 37. A device according to claim 36, wherein each image portion of the at least one encoded video stream being associated with a resolution level data indicating a level of resolution with which said each image portion is encoded and wherein said resolution level data comprises a superimposition data indicating a position of the data portion in a superimposition of image
25 layers.

38. A device according to any one of claims 36 and 37, wherein each image portion of the encapsulated encoded video stream is associated with a location information indicating a position of the image portion in the subdivision
30 grid.

39. A device according to any one of claims 36 to 38, wherein a plurality of encoded video streams are received, the processing unit being further configured for combining the plurality of video streams obtained by said decoding, said combining comprising assembling the at least one same
5 respective image portion of said subdivision grid of each video stream thereby obtaining said region of interest.

40. A device according to any one of claims 36 to 39, wherein said at least one encoded video stream is encoded from a common subdivided video
10 stream, the processing unit being further configured for:

- receiving said common subdivided video stream encoded with a low resolution,
- decoding said encoded common subdivided video stream, and
- generating said video signal based at least on said decoded
15 common subdivided video stream.

41. A device according to any one of claims 36 to 40, wherein each one of the plurality of encoded video streams is encoded with a base layer with a low resolution and an enhancement layer with said images with at least one
20 same respective image portion of the subdivision grid encoded with a higher resolution than the other image portions of the subdivision grid, and wherein said decoding is performed based at least on said base layer and enhancement layer.

25 42. A device according to claim 41, wherein said base layer is the same for the plurality of encoded video streams.

43. An image processing system comprising at least one of a device according to any one of claims 22 to 28, a device according to anyone of claims
30 29-35 and a device according to any one of claims 36 to 42.

44. A computer program product comprising instructions for implementing a method according to any one of claims 1 to 21 when the program is loaded and executed by a programmable apparatus.

5 45. A non-transitory information storage means readable by a computer or a microprocessor storing instructions of a computer program, characterized in that it makes it possible to implement a method according to any one of claims 1 to 21.

10 46. A device substantially as hereinbefore described with reference to, and as shown in, Figure 9 of the accompanying drawings.

47. A system substantially as hereinbefore described with reference to, and as shown in, Figures 3a and 3b of the accompanying drawings.

15

48. A method of processing image data substantially as hereinbefore described with reference to, and as shown in, Figure 3a and 3b of the accompanying drawings.

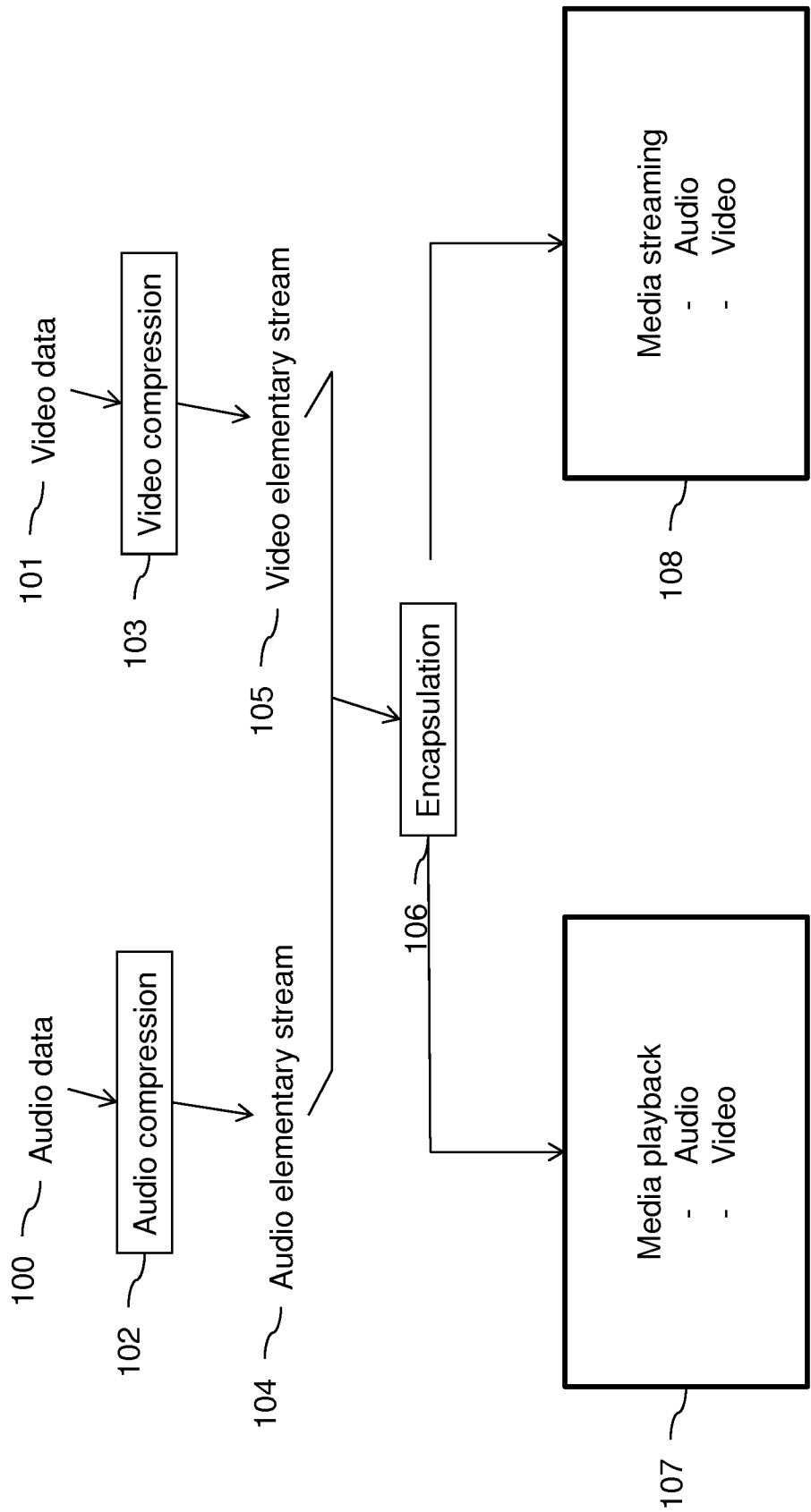


Fig. 1

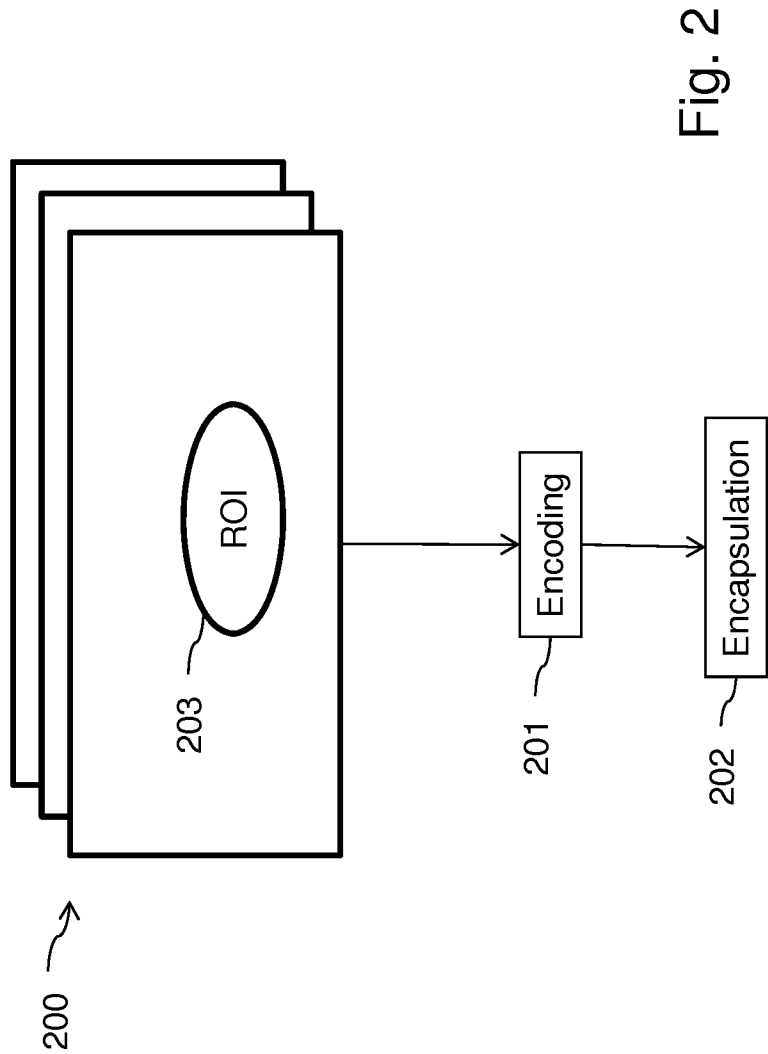


Fig. 2

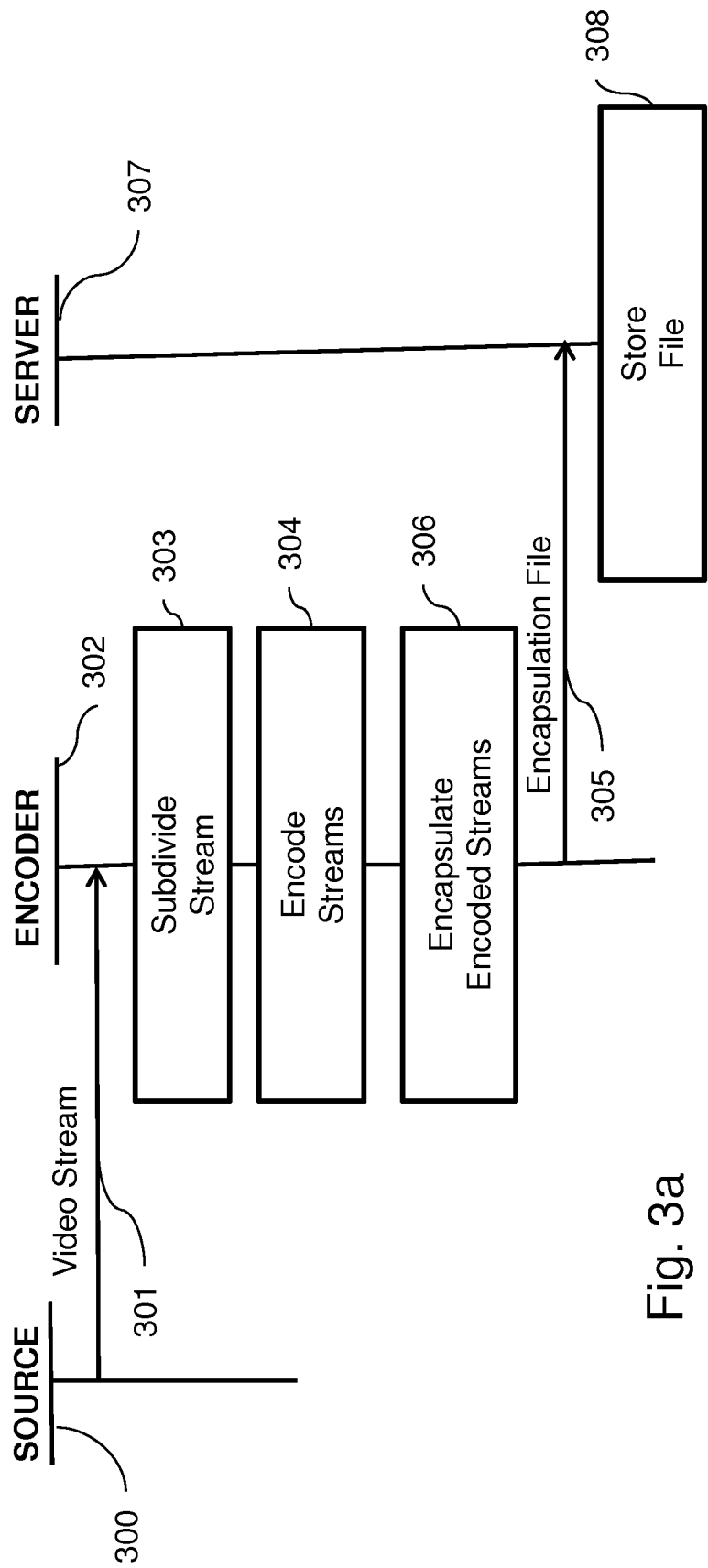


Fig. 3a

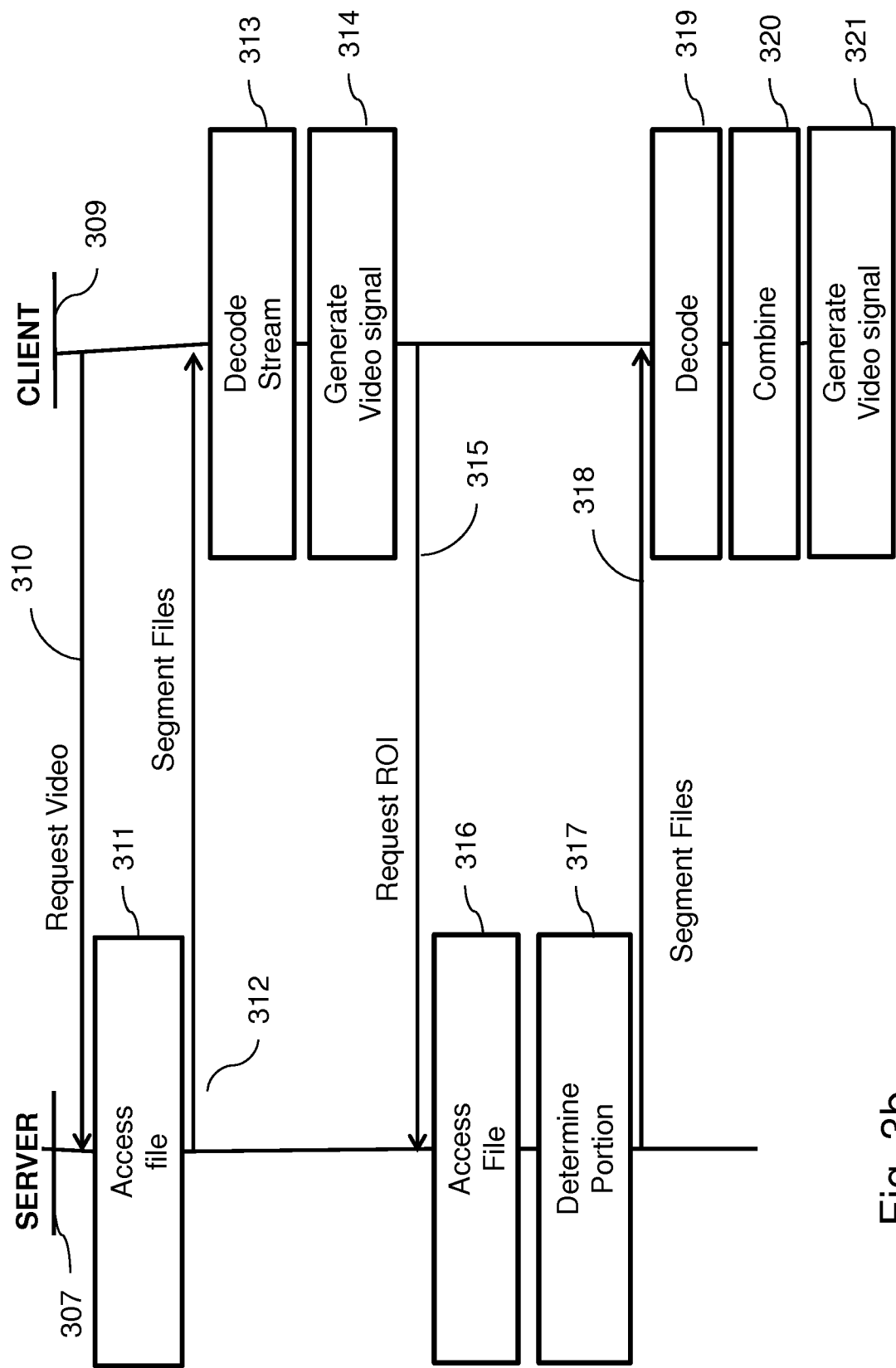
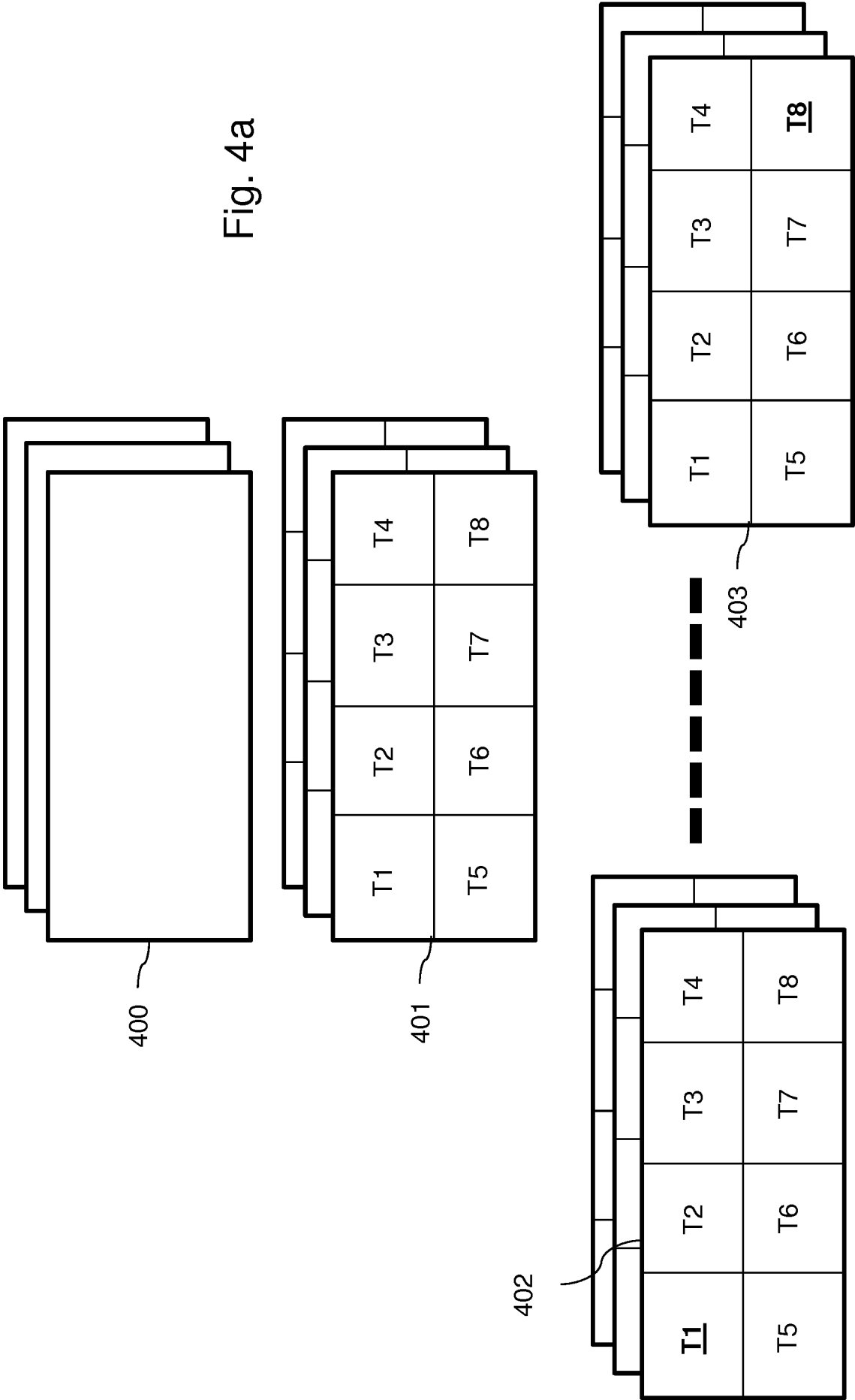


Fig. 3b

Fig. 4a



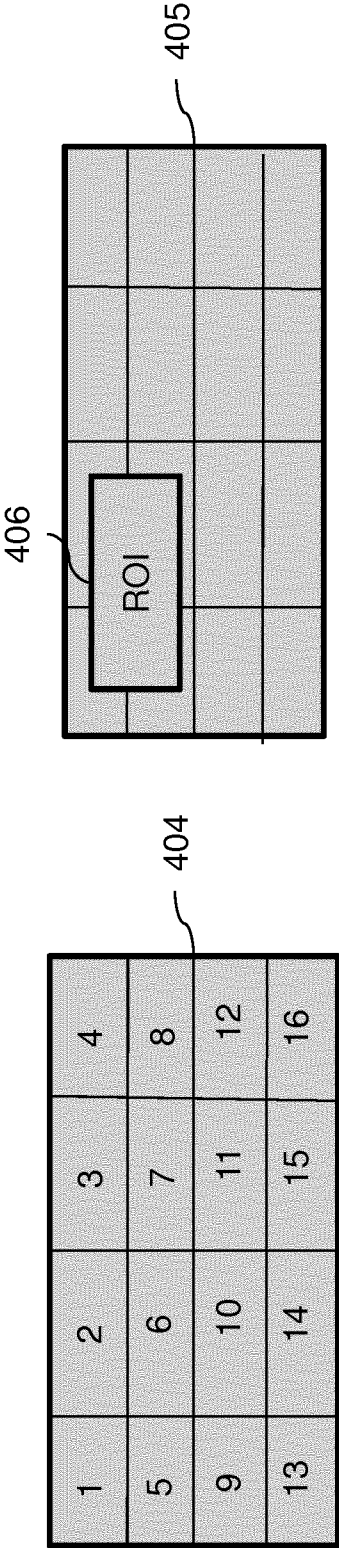


Fig. 4b

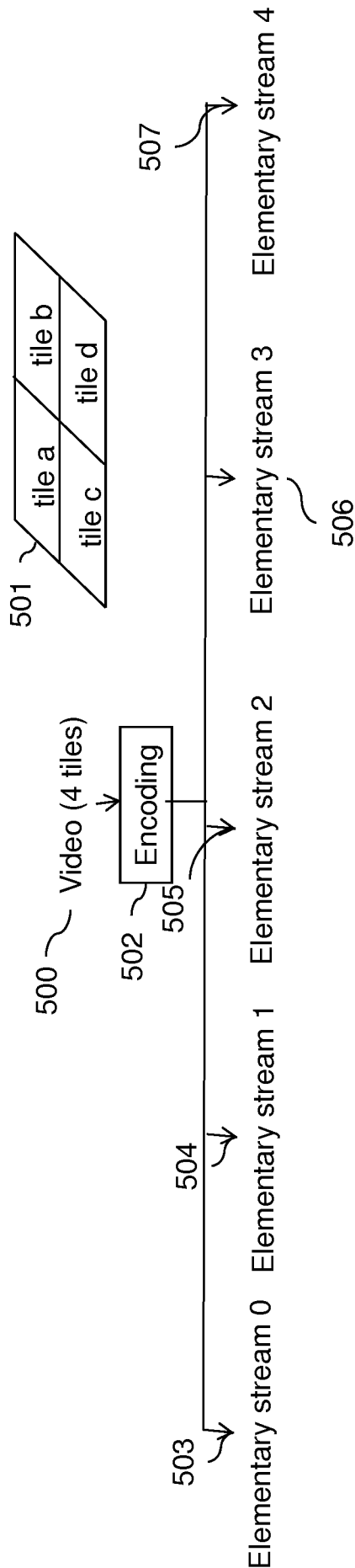


Fig. 5a

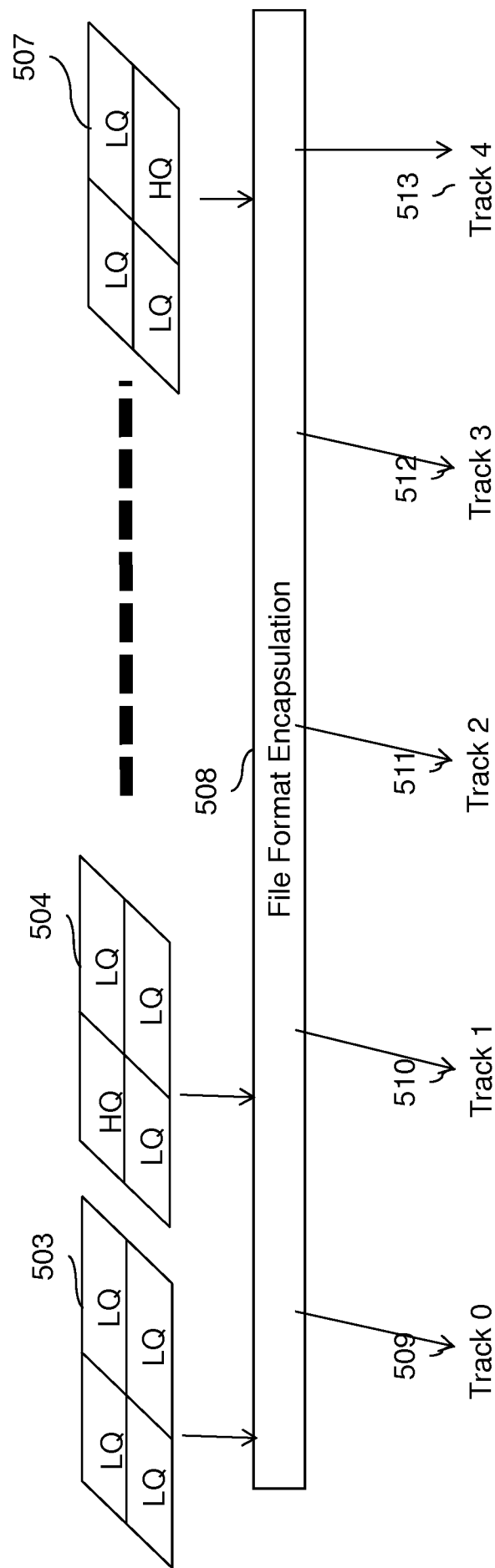


Fig. 5b

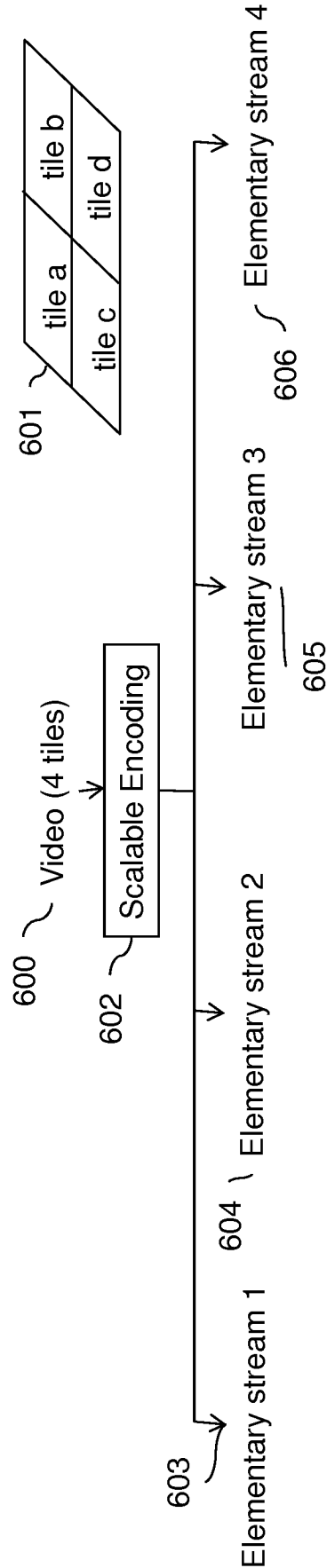


Fig. 6a

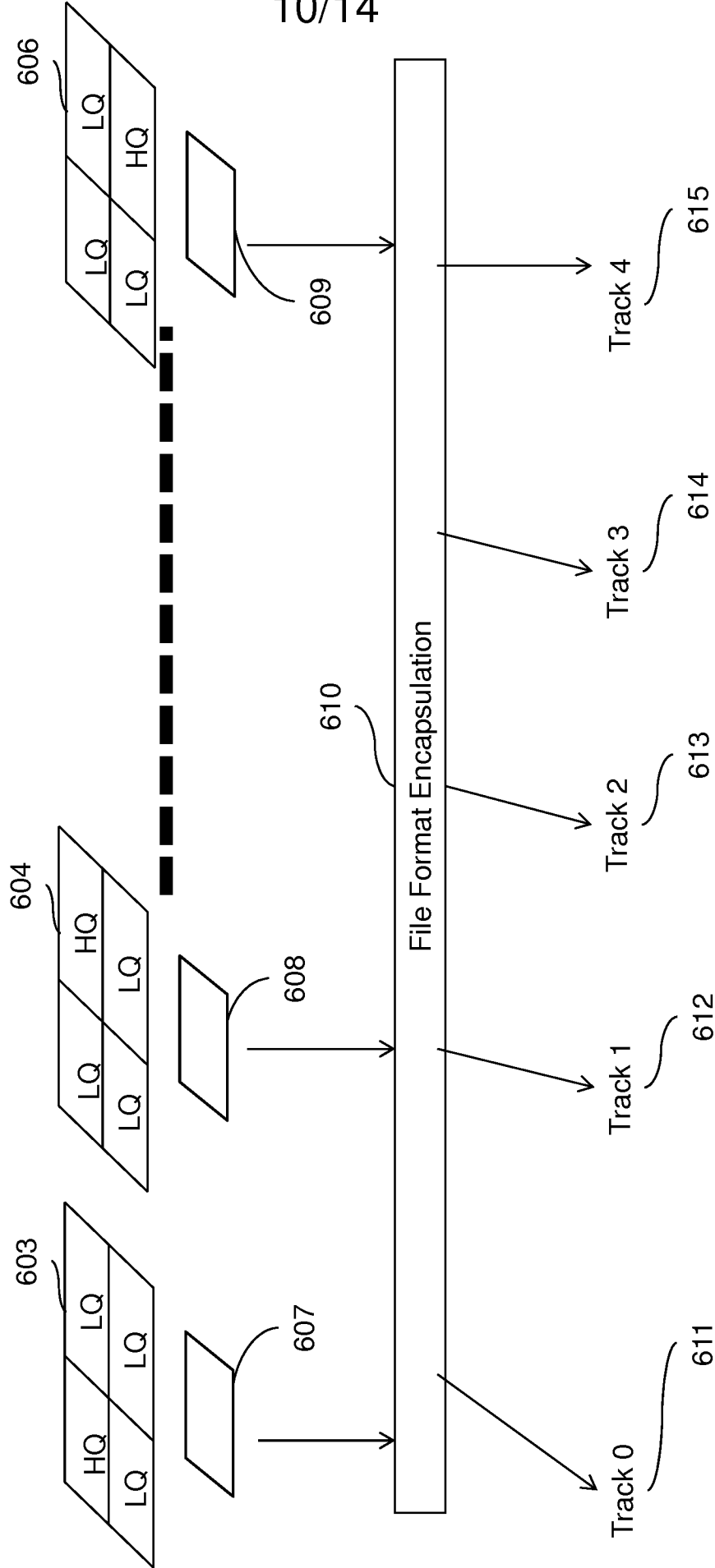


Fig. 6b

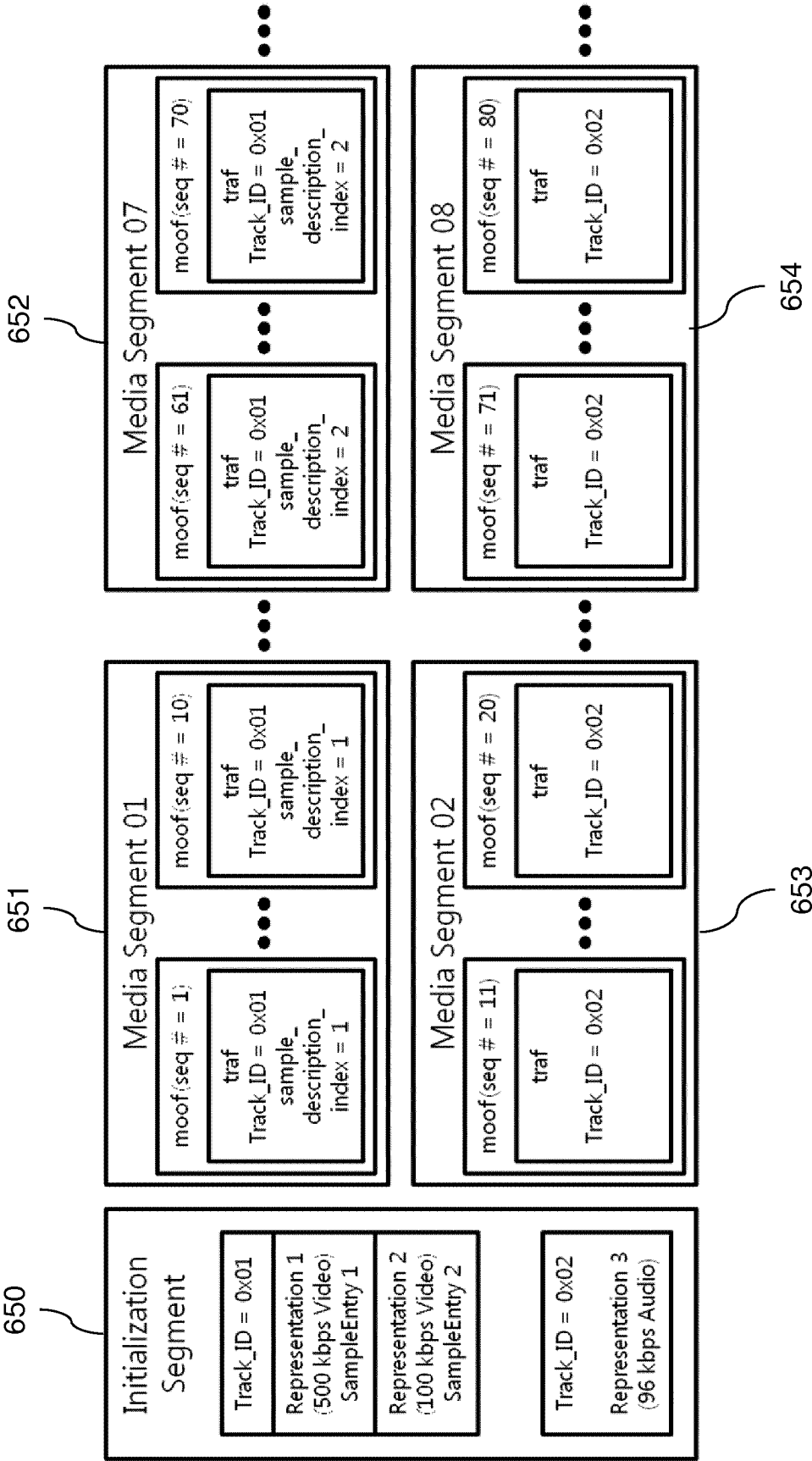


Fig. 6c

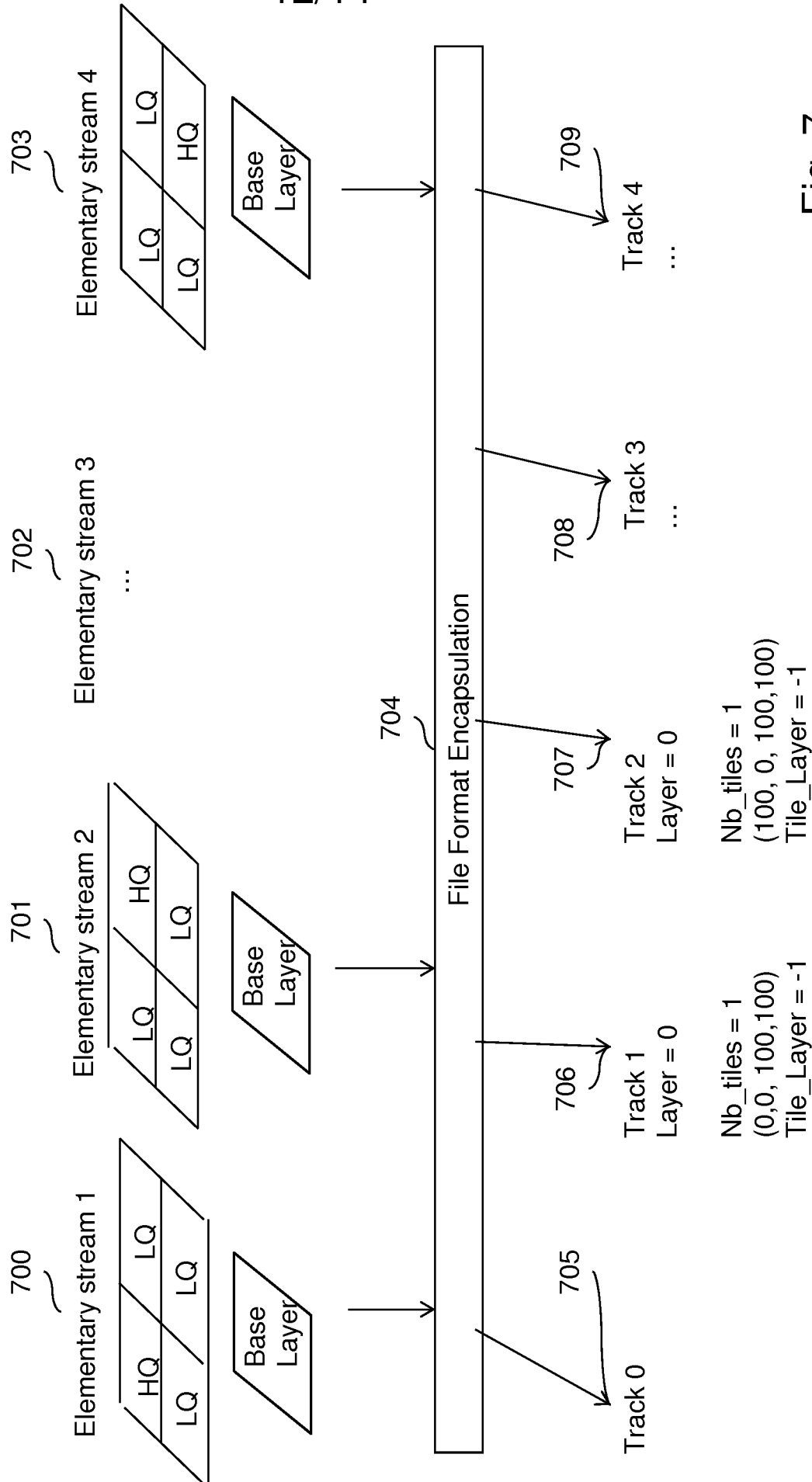


Fig. 7

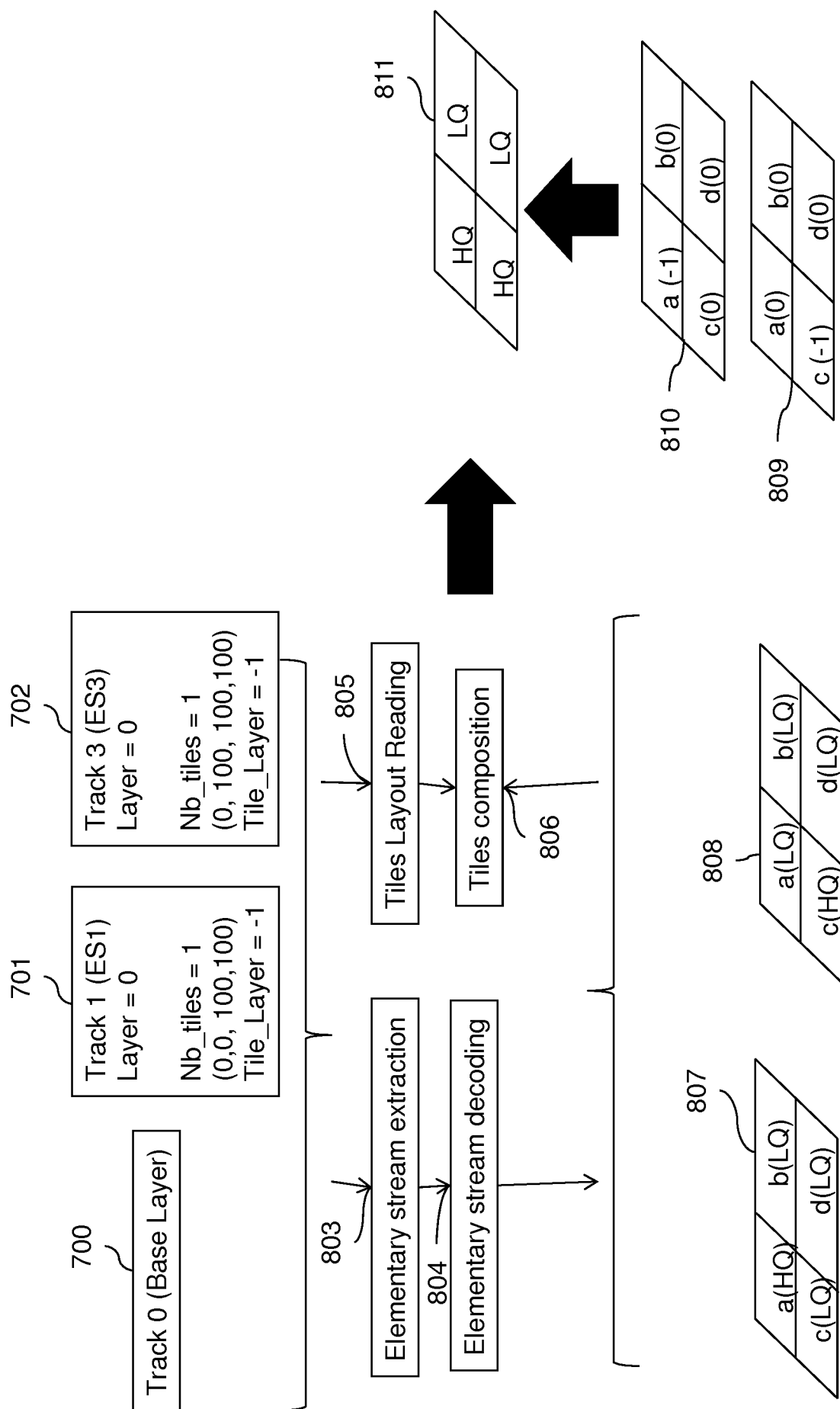


Fig. 8

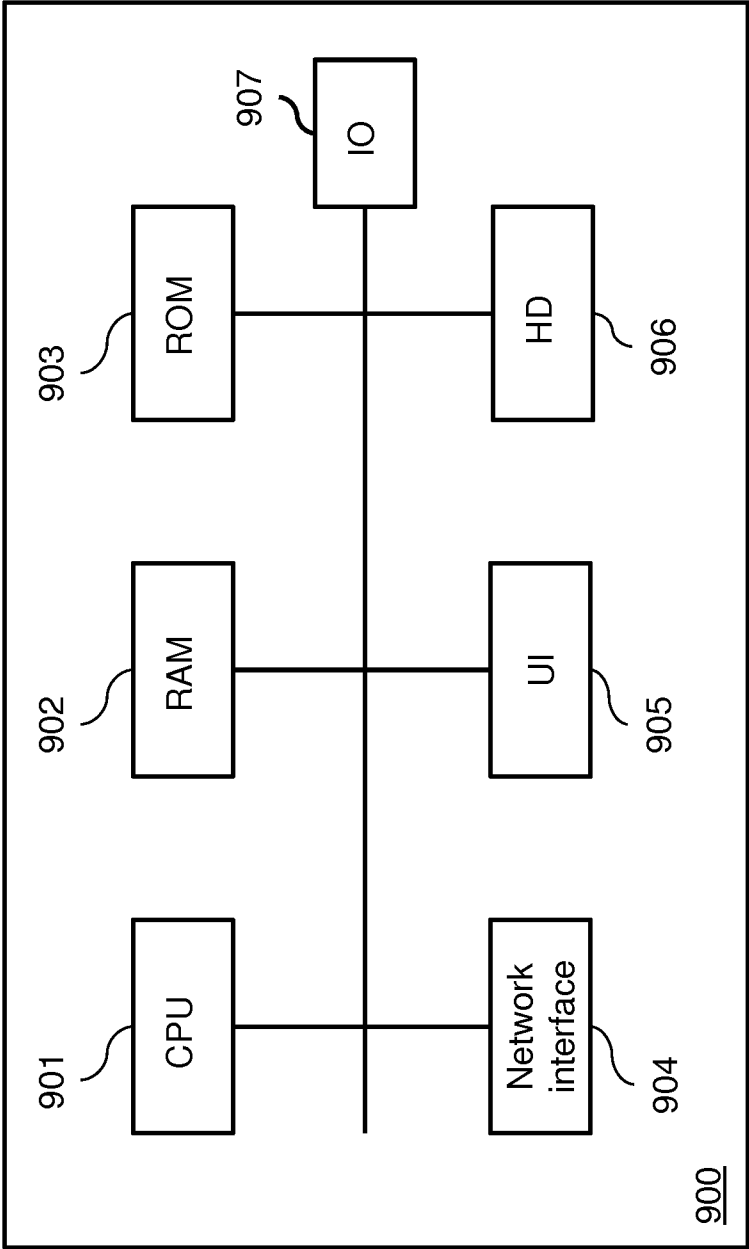


Fig. 9

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2014/050704

A. CLASSIFICATION OF SUBJECT MATTER

INV. H04N21/854 H04N21/4728 H04N21/2343 H04N19/17
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	ADITYA MAVLANKAR ET AL: "Optimal slice size for streaming regions of high resolution video with virtual pan/tilt/zoom functionality", PROC. OF 15TH EUROPEAN SIGNAL PROCESSING CONFERENCE (EUSIPCO}, 30 September 2007 (2007-09-30), XP055034766, DOI: 10.1.1.75.1021 Sections 3, 4 and 6 ----- -/--	1-48



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

7 May 2014

Date of mailing of the international search report

15/05/2014

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Sanahuja, Francesc

INTERNATIONAL SEARCH REPORT

International application No

PCT/EP2014/050704

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	ADITYA MAVLANKAR ET AL: "Peer-to-peer multicast live video streaming with interactive virtual pan/tilt/zoom functionality", IMAGE PROCESSING, 2008. ICIP 2008. 15TH IEEE INTERNATIONAL CONFERENCE, IEEE, PISCATAWAY, NJ, USA, 12 October 2008 (2008-10-12), pages 2296-2299, XP031374497, ISBN: 978-1-4244-1765-0 Section 3 -----	1-48
A	NGO QUANG MINH KHIEM ET AL: "Supporting zoomable video streams with dynamic region-of-interest cropping", PROCEEDINGS OF THE FIRST ANNUAL ACM SIGMM CONFERENCE ON MULTIMEDIA SYSTEMS, MMSYS '10, 22 February 2010 (2010-02-22), pages 259-270, XP055115422, New York, New York, USA DOI: 10.1145/1730836.1730868 ISBN: 978-1-60-558914-5 Sections 1, 2 and 4. -----	1-48
A	EP 1 335 603 A2 (MATSUSHITA ELECTRIC IND CO LTD [JP]) 13 August 2003 (2003-08-13) paragraphs [0181] - [0218] -----	1-48
A	AMON P ET AL: "File Format for Scalable Video Coding", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE SERVICE CENTER, PISCATAWAY, NJ, US, vol. 17, no. 9, 1 September 2007 (2007-09-01), pages 1174-1185, XP011193013, ISSN: 1051-8215, DOI: 10.1109/TCSVT.2007.905521 the whole document -----	1-48

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/EP2014/050704

Patent document cited in search report	Publication date	Patent family member(s)	Publication date	
EP 1335603	A2	13-08-2003	CN 1436003 A	13-08-2003
			EP 1335603 A2	13-08-2003
			JP 2003224846 A	08-08-2003
			US 2003142872 A1	31-07-2003
