



(51) International Patent Classification:

G06F 16/71 (2019.01) H04N 21/2343 (2011.01)
H04N 19/70 (2014.01) H04N 21/4402 (2011.01)

(21) International Application Number:

PCT/US2023/026358

(22) International Filing Date:

27 June 2023 (27.06.2023)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

63/367,148 28 June 2022 (28.06.2022) US

(71) Applicant: BYTEDANCE INC. [US/US]; 12655 West Jefferson Boulevard, Sixth Floor, Suite 137, Los Angeles, California 90066 (US).

(72) Inventor: WANG, Ye-Kui; 12655 West Jefferson Boulevard, Sixth Floor, Suite 137, Los Angeles, California 90066 (US).

(74) Agent: HOWELL, Brandt D. et al.; Conley Rose, P.C., 4965 Preston Park Boulevard, Suite 195E, Plano, Texas 75093 (US).

(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, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, 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, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, 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, ME, 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))

(54) Title: EDRAP SUPPORT IN ISOBMFF FOR ALL MEDIA TYPES

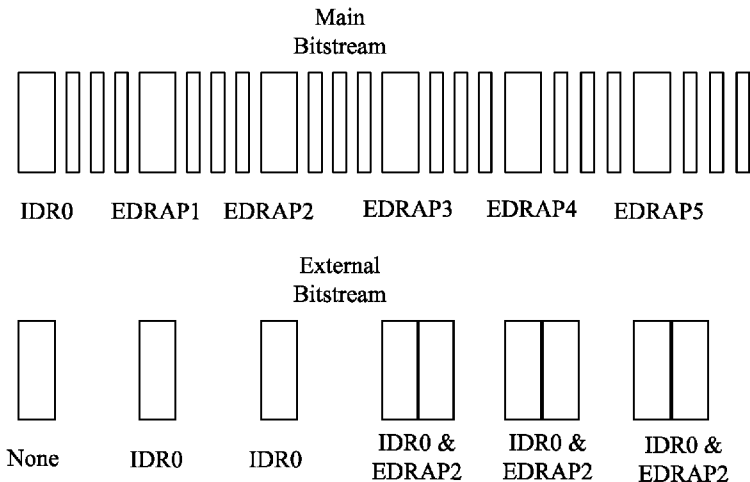


FIG. 4

(57) Abstract: A mechanism for processing video data is disclosed. A media track is determined to include a track reference type of associated external stream track (aest) A conversion is performed between a media data and a media data file based on a media track. The media track with the track reference type of aest may include an extended dependent random access point (EDRAP) sample.

EDRAP Support In ISOBMFF For All Media Types

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the priority to and benefits of U.S. Provisional Application No. 63/367,148, filed on June 28, 2022. All the aforementioned patent applications are hereby incorporated by reference in their entireties.

TECHNICAL FIELD

[0002] This patent document relates to generation, storage, and consumption of digital audio video media information in a file format.

BACKGROUND

[0003] Digital video accounts for the largest bandwidth used on the Internet and other digital communication networks. As the number of connected user devices capable of receiving and displaying video increases, the bandwidth demand for digital video usage is likely to continue to grow.

SUMMARY

[0004] A first aspect relates to a method for processing video data comprising: determining a media track includes a track reference type of associated external stream track (aest); and performing a conversion between a media data and a media data file based on the media track.

[0005] A second aspect relates to a non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method performed by a video processing apparatus, wherein the method comprises: determining a media track includes a track reference type of associated external stream track (aest); and generating a bitstream based on the determining.

[0006] A third aspect relates to an apparatus for processing video data comprising: a processor; and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to perform any of the preceding aspects.

[0007] A fourth aspect relates to non-transitory computer readable medium comprising a computer program product for use by a video coding device, the computer program product comprising computer executable instructions stored on the non-transitory computer readable medium such that when executed by a processor cause the video coding device to perform the method of any of the preceding aspects.

[0008] A fifth aspect relates to a method for storing bitstream of a video comprising: determining a media track includes a track reference type of associated external stream track (aest); generating a bitstream based on the determining; and storing the bitstream in a non-transitory computer-readable recording medium.

[0009] A sixth aspect relates to a method, apparatus or system described in the present document.

[0010] For the purpose of clarity, any one of the foregoing embodiments may be combined with any one or more of the other foregoing embodiments to create a new embodiment within the scope of the present disclosure.

[0011] These and other features will be more clearly understood from the following detailed description taken in conjunction with the accompanying drawings and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] For a more complete understanding of this disclosure, reference is now made to the following brief description, taken in connection with the accompanying drawings and detailed description, wherein like reference numerals represent like parts.

[0013] FIG. 1 is a schematic diagram of an example mechanism for random access when decoding a bitstream using intra random access points (IRAP) pictures.

[0014] FIG. 2 is a schematic diagram of an example mechanism for random access when decoding a bitstream using dependent random access point (DRAP) pictures.

[0015] FIG. 3 is a schematic diagram of an example mechanism for random access when decoding a bitstream using EDRAP pictures.

[0016] FIG. 4 is a schematic diagram of an example mechanism for signaling an external bitstream to support EDRAP based random access.

[0017] FIG. 5 illustrates an example of EDRAP based random access.

[0018] FIG. 6 is a block diagram showing an example video processing system.

[0019] FIG. 7 is a block diagram of an example video processing apparatus.

[0020] FIG. 8 is a flowchart for an example method of video processing.

[0021] FIG. 9 is a block diagram that illustrates an example video coding system.

[0022] FIG. 10 is a block diagram that illustrates an example encoder.

[0023] FIG. 11 is a block diagram that illustrates an example decoder.

[0024] FIG. 12 is a schematic diagram of an example encoder.

[0025] FIG. 13 is a flowchart for an example method of video processing.

DETAILED DESCRIPTION

[0026] It should be understood at the outset that although an illustrative implementation of one or more embodiments are provided below, the disclosed systems and/or methods may be implemented using any number of techniques, whether currently known or yet to be developed. The disclosure should in no way be limited to the illustrative implementations, drawings, and techniques illustrated below, including the exemplary designs and implementations illustrated and described herein, but may be modified within the scope of the appended claims along with their full scope of equivalents.

[0027] Section headings are used in the present document for ease of understanding and do not limit the applicability of techniques and embodiments disclosed in each section only to that section. Furthermore, H.266 terminology is used in some description only for ease of understanding and not for limiting scope of the disclosed techniques. As such, the techniques described herein are applicable to other video codec protocols and designs also. In the present document, editing changes are shown to text by bold italics indicating cancelled text and bold underline indicating added text, with respect to a draft of the Versatile Video Coding (VVC) specification or International Organization for Standardization (ISO) based media file format (ISOBMFF) file format specification.

1. Initial discussion

[0028] This document is related to media file formats. Specifically, it is related to support of extended dependent random access point (EDRAP) signaling in the International Organization for Standardization (ISO) base media file format (ISOBMFF) for all media types, including video, audio, etc. The ideas may be applied individually or in various combination, to media files according to any media file formats, such as the ISOBMFF and file formats derived from the ISOBMFF.

2. Video coding introduction

2.1 Video coding standards

[0029] Video coding standards have evolved primarily through the development of the International Telecommunication Union (ITU) telecommunication standardization sector (ITU-T) and International Organization for Standardization (ISO)/ International Electrotechnical Commission (IEC) standards. The ITU-T produced H.261 and H.263, ISO/IEC produced motion picture experts group (MPEG)-1 and MPEG-4 Visual, and the two organizations jointly produced

the H.262/MPEG-2 Video and H.264/MPEG-4 Advanced Video Coding (AVC) and H.265/ high efficiency video coding (HEVC) [1] standards. Since H.262, the video coding standards are based on the hybrid video coding structure wherein temporal prediction plus transform coding are utilized. To explore the future video coding technologies beyond HEVC, the Joint Video Exploration Team (JVET) was founded by video coding experts group (VCEG) and MPEG jointly. Many methods have been adopted by JVET and put into the reference software named Joint Exploration Model (JEM) [2]. The JVET was renamed to be the Joint Video Experts Team (JVET) when the Versatile Video Coding (VVC) project officially started. VVC [3] is a coding standard, targeting at 50% bitrate reduction as compared to HEVC.

[0030] The Versatile Video Coding (VVC) standard (ITU-T H.266 | ISO/IEC 23090-3) [3][4] and the associated Versatile Supplemental Enhancement Information (VSEI) standard (ITU-T H.274 | ISO/IEC 23002-7) [5][6] is designed for use in a maximally broad range of applications, including both the traditional uses such as television broadcast, video conferencing, or playback from storage media, and also newer and more advanced use cases such as adaptive bit rate streaming, video region extraction, composition and merging of content from multiple coded video bitstreams, multiview video, scalable layered coding, and viewport-adaptive 360° immersive media.

2.2 File format standards

[0031] Media streaming applications are based on the internet protocol (IP), Transmission Control Protocol (TCP), and Hypertext Transfer Protocol (HTTP) transport methods, and rely on a file format such as the ISO base media file format (ISOBMFF) [7]. One such streaming system is dynamic adaptive streaming over HTTP (DASH) [8]. For using a video format with ISOBMFF and DASH, a file format specification specific to the video format, such as the AVC file format and the HEVC file format in [9], would be needed for encapsulation of the video content in ISOBMFF tracks and in DASH representations and segments. Important information about the video bitstreams, e.g., the profile, tier, and level, and many others, would need to be exposed as file format level metadata and/or DASH media presentation description (MPD) for content selection purposes, e.g., for selection of appropriate media segments both for initialization at the beginning of a streaming session and for stream adaptation during the streaming session.

[0032] Similarly, for using an image format with ISOBMFF, a file format specification specific to the image format, such as the AVC image file format and the HEVC image file format in [10], would be needed.

2.3 Random access and its supports in HEVC and VVC

[0033] Random access refers to starting access and decoding of a bitstream from a picture that is not the first picture of the bitstream in decoding order. To support tuning in and channel switching in broadcast, multicast, and multiparty video conferencing, seeking in local playback and streaming, as well as stream adaptation in streaming, the bitstream should include frequent random-access points. Such random-access points may be intra coded pictures, but may also be inter-coded pictures, for example in the case of gradual decoding refresh.

[0034] HEVC includes signaling of intra random access points (IRAP) pictures in a NAL unit header through NAL unit types. Three types of IRAP pictures are supported in HEVC. These are instantaneous decoder refresh (IDR), clean random access (CRA), and broken link access (BLA) pictures. IDR pictures constrain the inter-picture prediction structure to not reference any picture before the current group-of-pictures (GOP). The reference pictures in the current GOP may be referred to as closed-GOP random access points. CRA pictures are less restrictive by allowing certain pictures to reference pictures before the current GOP, all of which are discarded in case of a random access. CRA pictures may be referred to as open-GOP random access points. BLA pictures usually originate from splicing of two bitstreams or part thereof at a CRA picture, for example during stream switching. To enable better systems usage of IRAP pictures, six different NAL units are defined to signal the properties of the IRAP pictures. Such properties can be used to better match the stream access point types as defined in the ISO/BMFF [7], which are utilized for random access support in dynamic adaptive streaming over hypertext transfer protocol (DASH) [8].

[0035] VVC supports three types of IRAP pictures, two types of IDR pictures (one type with and the other type without associated random access decodable leading (RADL) pictures) and one type of CRA picture. These are used in a similar manner as in HEVC. The BLA picture types in HEVC are not included in VVC for two reasons. First, the basic functionality of BLA pictures can be realized by CRA pictures plus the end of sequence NAL unit, the presence of which indicates that the subsequent picture starts a new CVS in a single-layer bitstream. Second, there is a desire in specifying fewer NAL unit types than HEVC during the development of VVC, as indicated by the use of five instead of six bits for the NAL unit type field in the NAL unit header.

[0036] Another difference in random access support between VVC and HEVC is the support of gradual decoding refresh (GDR) in a more normative manner in VVC. In GDR, the decoding of a bitstream can start from an inter-coded picture. At the beginning of an access, the entire picture

region can not be correctly decoded. However, after a number of pictures the entire picture region is correctly decoded. AVC and HEVC also support GDR by using a recovery point supplemental enhancement information (SEI) message for signaling of GDR random access points and recovery points. In VVC, a NAL unit type is specified for indication of GDR pictures and the recovery point is signaled in the picture header syntax structure. A coded video sequence (CVS) and a bitstream are allowed to start with a GDR picture. This means that an entire bitstream is allowed to contain only inter-coded pictures without a single intra-coded picture. The main benefit of specifying GDR support this way is to provide a conforming behavior for GDR. GDR enables encoders to smooth the bit rate of a bitstream by distributing intra-coded slices or blocks in multiple pictures as opposed intra coding entire pictures. This allows significant end-to-end delay reduction, which is considered more important in many cases as ultralow delay applications like wireless display, online gaming, and drone-based applications become more popular.

[0037] Another GDR related feature in VVC is virtual boundary signaling. The boundary between the refreshed region, which is the correctly decoded region, and the unrefreshed region at a picture between a GDR picture and a corresponding recovery point can be signaled as a virtual boundary. When signaled, in-loop filtering across the boundary is not applied. Thus, a decoding mismatch for some samples at or near the boundary would not occur. This can be useful when the application determines to display the correctly decoded regions during the GDR process. IRAP pictures and GDR pictures can be collectively referred to as random access point (RAP) pictures.

2.4 Extended dependent random access point (EDRAP) based video coding, storage, and streaming

2.4.1 The concept and the standard support

[0038] The concept of EDRAp based video coding, storage, and streaming is described herein. As shown in FIG. 1, the application (e.g., adaptive streaming) determines the frequency of random access points (RAPs), e.g., RAP period 1s or 2s. In an example, RAPs are provided by coding of IRAP pictures. Note that inter prediction references for the non-key pictures between RAP pictures are not shown, and from left to right is the output order. When random accessing from clean random access CRA4, the decoder receives and correctly decodes CRA4, CRA5, etc. and related inter predicted pictures.

[0039] FIG. 2 illustrates the DRAP approach, which provides improved coding efficiency by allowing a DRAP picture (and subsequent pictures) to refer to the previous IRAP picture for inter

prediction. Note that inter prediction for the non-key pictures between RAP pictures are not shown, and from left to right is the output order. When random accessing from DRAP4, the decoder receives and correctly decodes IDR0, DRAP4, DRAP5, etc. and related inter predicted pictures.

[0040] FIG. 3 illustrates the EDRAP approach, which provides a bit more flexibility by allowing an EDRAP picture (and subsequent pictures) to refer to a few of the earlier RAP pictures (IRAP or EDRAP). Note that inter prediction for the non-key pictures between RAP pictures are not shown, and from left to right is the output order. When random accessing from EDRAP4, the decoder receives and correctly decodes IDR0, EDRAP2, EDRAP4, EDRAP5, etc. and related inter predicted pictures.

[0041] FIG. 4 illustrates an example of the EDRAP approach using MSR segments and external stream representation (ESR) segments. FIG. 5 illustrates an example of random access from EDRAP4. When random accessing from or switching to the segment starting at EDRAP4, the decoder receives and decodes segments including IDR0, EDRAP2, EDRAP4, EDRAP5, etc. and related inter predicted pictures.

[0042] EDRAP based video coding is supported by the EDRAP indication SEI message included in [11], an amendment to the VSEI standard; the storage part is supported by the EDRAP sample group and the associated external stream track reference included in [12], an amendment to the ISO BMFF standard; and the streaming part is supported by the main stream representation (MSR) and ESR descriptors included in [13], an amendment to the DASH standard. These standard supports are described below.

2.4.2 The EDRAP indication SEI message

[0043] An amendment to the VSEI standard is under development. An example draft specification of this amendment is included in [11], which includes the specification of the EDRAP indication SEI message.

[0044] The syntax and semantics of the EDRAP indication SEI message are as follows.

extended_drap_indication(payloadSize) {	Descriptor
edrap_rap_id_minus1	u(16)
edrap_leading_pictures_decodable_flag	u(1)
edrap_reserved_zero_12bits	u(12)

edrap_num_ref_rap_pics_minus1	u(3)
for(i = 0; i <= edrap_num_ref_rap_pics_minus1; i++)	
edrap_ref_rap_id[i]	u(16)
}	

[0045] The picture associated with an extended DRAP (EDRAP) indication SEI message is referred to as an EDRAP picture.

[0046] The presence of the EDRAP indication SEI message indicates that the constraints on picture order and picture referencing specified in this subclause apply. These constraints can enable a decoder to properly decode the EDRAP picture and the pictures that are in the same layer and follow it in both decoding order and output order without needing to decode any other pictures in the same layer except the list of pictures referenceablePictures, which includes a list of IRAP or EDRAP pictures in decoding order that are within the same coded layer video sequence (CLVS) and identified by the edrap_ref_rap_id[i] syntax elements.

[0047] The constraints indicated by the presence of the EDRAP indication SEI message, which shall all apply, are as follows: The EDRAP picture is a trailing picture. The EDRAP picture has a temporal sublayer identifier equal to 0. The EDRAP picture does not include any pictures in the same layer in the active entries of its reference picture lists except the referenceablePictures. Any picture that is in the same layer and follows the EDRAP picture in both decoding order and output order does not include, in the active entries of its reference picture lists, any picture that is in the same layer and precedes the EDRAP picture in decoding order or output order, with the exception of the referenceablePictures. Any picture in the list referenceablePictures does not include, in the active entries of its reference picture lists, any picture that is in the same layer and is not a picture at an earlier position in the list referenceablePictures. NOTE – Consequently, the first picture in referenceablePictures, even when it is an EDRAP picture instead of an IRAP picture, does not include any picture from the same layer in the active entries of its reference picture lists.

[0048] edrap_rap_id_minus1 plus 1 specifies the RAP picture identifier, denoted as RapPicId, of the EDRAP picture.

[0049] Each IRAP or EDRAP picture is associated with a RapPicId value. The RapPicId value for an IRAP picture is inferred to be equal to 0. The RapPicId values for any two EDRAP pictures associated with the same IRAP picture shall be different.

[0050] edrap_leading_pictures_decodable_flag equal to 1 specifies that both of the following constraints apply: Any picture that is in the same layer and follows the EDRAP picture in decoding order shall follow, in output order, any picture that is in the same layer and precedes the EDRAP picture in decoding order. Any picture that is in the same layer and follows the EDRAP picture in decoding order and precedes the EDRAP picture in output order shall not include, in the active entries of its reference picture lists, any picture that is in the same layer and precedes the EDRAP picture in decoding order, with the exception of the referenceablePictures.

[0051] edrap_leading_pictures_decodable_flag equal to 0 does not impose such constraints.

[0052] edrap_reserved_zero_12bits shall be equal to 0 in bitstreams conforming to this version of this Specification. Other values for edrap_reserved_zero_12bits are reserved for use by ITU-T | ISO/IEC. Decoders shall ignore the value of edrap_reserved_zero_12bits.

[0053] edrap_num_ref_rap_pics_minus1 plus 1 indicates the number of IRAP or EDRAP pictures that are within the same CLVS as the EDRAP picture and may be included in the active entries of the reference picture lists of the EDRAP picture.

[0054] edrap_ref_rap_id[i] indicates RapPicId of the i-th RAP picture that may be included in the active entries of the reference picture lists of the EDRAP picture. The i-th RAP picture shall be either the IRAP picture associated with the current EDRAP picture or an EDRAP picture associated with the same IRAP picture as the current EDRAP picture.

2.4.3 The EDRAP sample group and associated external stream track reference

[0055] An amendment to the ISO/BMFF standard is under development. A draft specification of this amendment is included in [14], which includes the specifications of the EDRAP sample group and the associated external stream track reference.

[0056] The specifications of these two ISO/BMFF features are as follows.

3.1 Definitions

...

EDRAP sample

[0057] A sample for which all subsequent samples in both decoding and output order can be correctly decoded provided that the closest preceding SAP sample of type 1, 2, or 3 and zero or more preceding EDRAP samples are available when decoding the sample and the subsequent samples. Note 1 to entry. The closest preceding SAP sample of type 1, 2, or 3 and the zero or more preceding

EDRAP samples as described above are referred to as the required preceding SAP and EDRAP samples of the EDRAP sample.

...

random access

[0058] A decoding of an elementary stream starting from a particular access unit without decoding of any access unit in the elementary stream earlier in decoding order. Note 1 to entry. Sync samples and SAPs provide random accessing capabilities.

...

3.2 Abbreviated terms

...

EDRAP extended dependent random access point

...

8.3.3.4 Associated external stream track reference

[0059] A track reference of type 'aest' (meaning "associated external stream track") may be included in a video track. When a video track has a track reference of type 'aest', the following applies: The video track should have at least one sample associated with an EDRAP sample group. The referenced track shall comply to the following constraints: Each sample in the referenced track shall be identified as a sync sample. The referenced track shall have both header flags `track_in_movie` and `track_in_preview` equal to 0. The referenced track shall use a restricted scheme, as follows: The `scheme_type` field in the `SchemeTypeBox`, which is in the `RestrictedSchemeInfoBox`, is equal to 'tspt' and the value of the `mode` field in the `SamplePackingInformationBox` is equal to 1. Bit 0 of the `flags` field of the `SchemeTypeBox` is equal to 0, such that the value of $(\text{flags} \& 0x000001)$ is equal to 0. For each sample `sampleA` in the video track associated with an EDRAP sample group, there shall be one and only one sample `sampleB` in the referenced track that has the same decoding time as `sampleA`, and a number of consecutive samples in the referenced track, starting from `sampleB`, shall contain all media data of the required preceding SAP and EDRAP samples of `sampleA`. The consecutive samples in the referenced track shall precede `sampleC` corresponding to another sample in the video track associated with an EDRAP sample group.

8.20 Transformed sample-packed tracks

8.20.1 Introduction

[0060] When a restricted scheme with `SchemeType` 'tspt' is in use for a track, a sample associated with the sample entry may contain more than one sample of an original track, on which a

transformation has been applied to produce the current track. Such a current track is referred to as a sample-packed track.

8.20.2 Transformed sample packing information box

8.20.2.1 Definition

Box Type: 'tspi'
 Container: SchemeInformationBox
 Mandatory: Yes (when the SchemeType is 'tspt')
 Quantity: One

8.20.2.2 Syntax

aligned(8) class TransformedSamplePackingInformationBox extends extends FullBox('tspi',
 version = 0, flags = 0)
 {
 unsigned int(8) mode;
 }

8.20.2.3 Semantics

[0061] Mode equal to 0 specifies that all the samples of the original streams have been preserved in the transformation. The value 1 specifies that only some samples have been preserved. All other values are reserved for future use.

10.11 Extended DRAP (EDRAP) sample group

10.11.1 Definition

[0062] The EDRAP sample group documents the EDRAP samples in a track. This sample group is similar to the DRAP sample group as specified in subclause 10.8; however, it enables signalling additional samples, that can also be used for random access but that have more complex dependencies. NOTE 1: Similarly as for DRAP samples, EDRAP samples can only be used in combination with SAP samples of type 1, 2 and 3. NOTE 2: A DRAP sample is always an EDRAP sample.

10.11.2 Syntax

```

class VisualEdrapEntry()
extends VisualSampleGroupEntry('edrp') {
    unsigned int(3) edrap_type;
    unsigned int(3) num_ref_edrap_pics;
    unsigned int(26) reserved = 0;
    for(i=0; i<num_ref_edrap_pics; i++)
        unsigned int(16) ref_edrap_idx_delta[i];
}

```

10.11.3 Semantics

[0063] `edrap_type` is a non-negative integer. When `edrap_type` is in the range of 1 to 3 it indicates the `SAP_type` (as specified in Annex I) that the EDRAP sample would have corresponded to, had it not depended on the closest preceding SAP or other EDRAP samples. Other type values are reserved. `num_ref_edrap_pics` indicates the number of other EDRAP samples that are earlier in decoding order than the EDRAP sample and are needed for reference to be able to correctly decode the EDRAP sample and all samples following the EDRAP sample in both decoding and output order when starting decoding from the EDRAP sample. NOTE: an EDRAP sample which is also a DRAP sample would have `num_ref_edrap_pics` set to 0.

[0064] `reserved` shall be equal to 0. The semantics of this subclause only apply to sample group description entries with `reserved` equal to 0. Parsers shall allow and ignore sample group description entries with `reserved` greater than 0 when parsing this sample group. `ref_edrap_idx_delta[i]` indicates the *i*-th required preceding EDRAP sample of the current EDRAP sample. Let the list of EDRAP samples associated with a SAP sample of type 1, 2 or 3 be all the EDRAP samples following the SAP sample and preceding the next SAP sample, when present. The EDRAP sample index is defined as the index to this list of EDRAP samples. The value of `ref_edrap_idx_delta[i]` is equal to the EDRAP sample index of the current EDRAP sample and the EDRAP sample index of the *i*-th required preceding EDRAP sample. The value 1 indicates that the *i*-th EDRAP sample is the last EDRAP sample preceding this EDRAP sample in decoding order, the value 2 indicates that the *i*-th EDRAP sample is the second last EDRAP sample preceding this EDRAP sample in decoding order, and so on.

3. Technical problems solved by disclosed technical solutions

Existing design for the storage part of EDRAP based media coding, storage and streaming are associated with the following problems:

- 1) The EDRAP mechanism is only specified for video, not other media types.
- 2) It is allowed for an EDRAP sample in a track containing the EDRAP sample group signalling to be not a member of any EDRAP sample group.
- 3) It is allowed for a track containing a track reference of type 'aest' to have no sample to be a member of an EDRAP sample group (i.e., associated with an EDRAP sample group).

4. A listing of solutions and embodiments

[0065] To solve the above-described problem, methods as summarized below are disclosed. The examples should be considered as examples to explain the general concepts and should not be interpreted in a narrow way. Furthermore, these examples can be applied individually or combined in any manner.

Example 1

[0066] In one example, to solve the first problem, the EDRAP sample group and the 'aest' track reference are specified to be applicable to other media type(s), including video, audio, etc.

Example 2

[0067] In one example, to solve the second problem, an EDRAP sample in a track containing the EDRAP sample group signaling is disallowed from not being a member of any EDRAP sample group. In an example, each EDRAP sample in a track containing the EDRAP sample group signaling is required to be a member of an EDRAP sample group. In an example, when a track has a SampleToGroupBox with grouping_type equal to 'edrp', each EDRAP sample in the track shall be a member of an EDRAP sample group.

Example 3

[0068] In an example, to solve the second problem, a track containing a track reference of type 'aest' is disallowed from having no sample that is a member of an EDRAP sample group (e.g., associated with an EDRAP sample group). In an example, a track containing a track reference of type 'aest' shall have at least one sample that is a member of an EDRAP sample group. In an example, when a media track has a track reference of type 'aest', the media track shall have at least one sample associated with an EDRAP sample group.

5. Embodiments

[0069] Below are some example embodiments for some of the disclosure items summarized above in section 4. Most relevant parts that have been added or modified are shown in underlined bold font, and some of the deleted parts are shown in italicized bold fonts. There may be some other changes that are editorial in nature and thus not highlighted.

5.1 First embodiment

[0070] This embodiment is for items 1, 2, and 3.

8.3.3.4 Associated external stream track reference

[0071] A track reference of type 'aest' (meaning "associated external stream track") may be included in a media video track. When a media track has a track reference of type 'aest', the following applies: The media video track shall *should* have at least one sample associated with an EDRAP sample group. The referenced track shall comply to the following constraints: Each sample in the referenced track shall be identified as a sync sample. The referenced track shall have both header flags track_in_movie and track_in_preview equal to 0.

[0072] The referenced track shall use a restricted scheme, as follows: The scheme_type field in the SchemeTypeBox, which is in the RestrictedSchemeInfoBox, is equal to 'tspt' and the value of the mode field in the SamplePackingInformationBox is equal to 1. Bit 0 of the flags field of the SchemeTypeBox is equal to 0, such that the value of (flags & 0x000001) is equal to 0.

[0073] For each EDRAP sample sampleA in the media video track *associated with an EDRAP sample group*, there shall be one and only one sample sampleB in the referenced track that has the same decoding time as sampleA, and a number of consecutive samples in the referenced track, starting from sampleB, shall contain all media data of the required preceding SAP and EDRAP samples of sampleA. The consecutive samples in the referenced track shall precede sampleC corresponding to another sample in the media video track associated with an EDRAP sample group.

...

10.11 Extended DRAP (EDRAP) sample group

10.11.1 Definition

[0074] The EDRAP sample group documents the EDRAP samples in a track. This sample group is similar to the DRAP sample group as specified in subclause 10.8; however, it enables signalling

additional samples that can also be used for random access but have more flexible dependencies.

When a track has a SampleToGroupBox with grouping_type equal to 'edrp', each EDRAP sample in the track shall be a member of an EDRAP sample group.

NOTE 1: Similarly as for DRAP samples, EDRAP samples can only be used in combination with SAP samples of type 1, 2 and 3. NOTE 2: A DRAP sample is always an EDRAP sample.

10.11.2 Syntax

```
class VisualEdrapEntry()
extends VisualSampleGroupEntry('edrp') {
    unsigned int(3) edrap_type;
    unsigned int(3) num_ref_edrap_samples;
    unsigned int(26) reserved = 0;
    for(i=0; i<num_ref_edrap_samples; i++)
        unsigned int(16) ref_edrap_idx_delta[i];
}
```

10.11.3 Semantics

[0075] edrap_type is a non-negative integer. When edrap_type is in the range of 1 to 3 it indicates the SAP_type (as specified in Annex I) that the EDRAP sample would have corresponded to, had it not depended on the closest preceding SAP or other EDRAP samples. Other type values are reserved. num_ref_edrap_samples indicates the number of other EDRAP samples that are earlier in decoding order than the EDRAP sample and are needed for reference to be able to correctly decode the EDRAP sample and all samples following the EDRAP sample in both decoding and output order when starting decoding from the EDRAP sample. NOTE: An EDRAP sample that is also a DRAP sample would have num_ref_edrap_samples equal to 0.

[0076] reserved shall be equal to 0. The semantics of this subclause only apply to sample group description entries with reserved equal to 0. Parsers shall allow and ignore sample group description entries with reserved greater than 0 when parsing this sample group. ref_edrap_idx_delta[i] indicates the i-th required preceding EDRAP sample of the current EDRAP sample. Let the list of EDRAP samples associated with a SAP sample of type 1, 2 or 3 be all the EDRAP samples following the SAP sample and preceding the next SAP sample, when present. The EDRAP sample index is defined as the index to this list of EDRAP samples. The value of ref_edrap_idx_delta[i] is equal to the

EDRAP sample index of the current EDRAP sample and the EDRAP sample index of the i-th required preceding EDRAP sample. The value 1 indicates that the i-th EDRAP sample is the last EDRAP sample preceding this EDRAP sample in decoding order, the value 2 indicates that the i-th EDRAP sample is the second last EDRAP sample preceding this EDRAP sample in decoding order, and so on.

7. References

- [1] ITU-T and ISO/IEC, "High efficiency video coding", Rec. ITU-T H.265 | ISO/IEC 23008-2 (in force edition).
- [2] J. Chen, E. Alshina, G. J. Sullivan, J.-R. Ohm, J. Boyce, "Algorithm description of Joint Exploration Test Model 7 (JEM7)," JVET-G1001, Aug. 2017.
- [3] Rec. ITU-T H.266 | ISO/IEC 23090-3, "Versatile Video Coding", 2020.
- [4] B. Bross, J. Chen, S. Liu, Y.-K. Wang (editors), "Versatile Video Coding (Draft 10)," JVET-S2001.
- [5] Rec. ITU-T Rec. H.274 | ISO/IEC 23002-7, "Versatile Supplemental Enhancement Information Messages for Coded Video Bitstreams", 2020.
- [6] J. Boyce, V. Drugeon, G. J. Sullivan, Y.-K. Wang (editors), "Versatile supplemental enhancement information messages for coded video bitstreams (Draft 5)," JVET-S2007.
- [7] ISO/IEC 14496-12: "Information technology — Coding of audio-visual objects — Part 12: ISO base media file format".
- [8] ISO/IEC 23009-1: "Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats".
- [9] ISO/IEC 14496-15: "Information technology — Coding of audio-visual objects — Part 15: Carriage of network abstraction layer (NAL) unit structured video in the ISO base media file format".
- [10] ISO/IEC 23008-12: "Information technology — High efficiency coding and media delivery in heterogeneous environments — Part 12: Image File Format".
- [11] J. Boyce, G. J. Sullivan, Y.-K. Wang (editors), "Additional SEI messages for VSEI (Draft 6)," JVET-Y2006.
- [12] ISO/IEC JTC 1/SC 29/WG 03 output document N0550, "Potential Improvements of Text of CDAM ISO/IEC 14496-12:2021 AMD 1 Improved brand documentation and other improvements", Apr. 2022.

[13] ISO/IEC JTC 1/SC 29/WG 03 output document N0559, "WD of ISO/IEC 23009-1 5th edition AMD 2 EDRAp streaming and other extensions", Apr. 2022.

[0077] FIG. 6 is a block diagram showing an example video processing system 4000 in which various techniques disclosed herein may be implemented. Various implementations may include some or all of the components of the system 4000. The system 4000 may include input 4002 for receiving video content. The video content may be received in a raw or uncompressed format, e.g., 8 or 10 bit multi-component pixel values, or may be in a compressed or encoded format. The input 4002 may represent a network interface, a peripheral bus interface, or a storage interface. Examples of network interface include wired interfaces such as Ethernet, passive optical network (PON), etc. and wireless interfaces such as Wi-Fi or cellular interfaces.

[0078] The system 4000 may include a coding component 4004 that may implement the various coding or encoding methods described in the present document. The coding component 4004 may reduce the average bitrate of video from the input 4002 to the output of the coding component 4004 to produce a coded representation of the video. The coding techniques are therefore sometimes called video compression or video transcoding techniques. The output of the coding component 4004 may be either stored, or transmitted via a communication connected, as represented by the component 4006. The stored or communicated bitstream (or coded) representation of the video received at the input 4002 may be used by a component 4008 for generating pixel values or displayable video that is sent to a display interface 4010. The process of generating user-viewable video from the bitstream representation is sometimes called video decompression. Furthermore, while certain video processing operations are referred to as "coding" operations or tools, it will be appreciated that the coding tools or operations are used at an encoder and corresponding decoding tools or operations that reverse the results of the coding will be performed by a decoder.

[0079] Examples of a peripheral bus interface or a display interface may include universal serial bus (USB) or high definition multimedia interface (HDMI) or Displayport, and so on. Examples of storage interfaces include serial advanced technology attachment (SATA), peripheral component interconnect (PCI), integrated drive electronics (IDE) interface, and the like. The techniques described in the present document may be embodied in various electronic devices such as mobile phones, laptops, smartphones or other devices that are capable of performing digital data processing and/or video display.

[0080] FIG. 7 is a block diagram of an example video processing apparatus 4100. The apparatus 4100 may be used to implement one or more of the methods described herein. The apparatus 4100 may be embodied in a smartphone, tablet, computer, Internet of Things (IoT) receiver, and so on. The apparatus 4100 may include one or more processors 4102, one or more memories 4104 and video processing circuitry 4106. The processor(s) 4102 may be configured to implement one or more methods described in the present document. The memory (memories) 4104 may be used for storing data and code used for implementing the methods and techniques described herein. The video processing circuitry 4106 may be used to implement, in hardware circuitry, some techniques described in the present document. In some embodiments, the video processing circuitry 4106 may be at least partly included in the processor 4102, e.g., a graphics co-processor.

[0081] FIG. 8 is a flowchart for an example method 4200 of video processing. The method 4200 determines an extended dependent random access point (EDRAP) sample group references audio data at step 4202. A conversion is performed between a media data and the media data file based on the EDRAp sample at step 4204. In some examples, the EDRAp sample group may reference any media data, such a video, audio, closed captioning, or any type of media data. The conversion may include encoding at an encoder, decoding at a decoder, or combinations thereof.

[0082] It should be noted that the method 4200 can be implemented in an apparatus for processing video data comprising a processor and a non-transitory memory with instructions thereon, such as video encoder 4400, video decoder 4500, and/or encoder 4600. In such a case, the instructions upon execution by the processor, cause the processor to perform the method 4200. Further, the method 4200 can be performed by a non-transitory computer readable medium comprising a computer program product for use by a video coding device. The computer program product comprises computer executable instructions stored on the non-transitory computer readable medium such that when executed by a processor cause the video coding device to perform the method 4200.

[0083] FIG. 9 is a block diagram that illustrates an example video coding system 4300 that may utilize the techniques of this disclosure. The video coding system 4300 may include a source device 4310 and a destination device 4320. Source device 4310 generates encoded video data which may be referred to as a video encoding device. Destination device 4320 may decode the encoded video data generated by source device 4310 which may be referred to as a video decoding device.

[0084] Source device 4310 may include a video source 4312, a video encoder 4314, and an input/output (I/O) interface 4316. Video source 4312 may include a source such as a video capture device, an interface to receive video data from a video content provider, and/or a computer graphics system for generating video data, or a combination of such sources. The video data may comprise one or more pictures. Video encoder 4314 encodes the video data from video source 4312 to generate a bitstream. The bitstream may include a sequence of bits that form a coded representation of the video data. The bitstream may include coded pictures and associated data. The coded picture is a coded representation of a picture. The associated data may include sequence parameter sets, picture parameter sets, and other syntax structures. I/O interface 4316 may include a modulator/demodulator (modem) and/or a transmitter. The encoded video data may be transmitted directly to destination device 4320 via I/O interface 4316 through network 4330. The encoded video data may also be stored onto a storage medium/server 4340 for access by destination device 4320.

[0085] Destination device 4320 may include an I/O interface 4326, a video decoder 4324, and a display device 4322. I/O interface 4326 may include a receiver and/or a modem. I/O interface 4326 may acquire encoded video data from the source device 4310 or the storage medium/ server 4340. Video decoder 4324 may decode the encoded video data. Display device 4322 may display the decoded video data to a user. Display device 4322 may be integrated with the destination device 4320, or may be external to destination device 4320, which can be configured to interface with an external display device.

[0086] Video encoder 4314 and video decoder 4324 may operate according to a video compression standard, such as the High Efficiency Video Coding (HEVC) standard, Versatile Video Coding (VVC) standard and other current and/or further standards.

[0087] FIG. 10 is a block diagram illustrating an example of video encoder 4400, which may be video encoder 4314 in the system 4300 illustrated in FIG. 9. Video encoder 4400 may be configured to perform any or all of the techniques of this disclosure. The video encoder 4400 includes a plurality of functional components. The techniques described in this disclosure may be shared among the various components of video encoder 4400. In some examples, a processor may be configured to perform any or all of the techniques described in this disclosure.

[0088] The functional components of video encoder 4400 may include a partition unit 4401, a prediction unit 4402 which may include a mode select unit 4403, a motion estimation unit 4404, a motion compensation unit 4405, an intra prediction unit 4406, a residual generation unit 4407, a

transform processing unit 4408, a quantization unit 4409, an inverse quantization unit 4410, an inverse transform unit 4411, a reconstruction unit 4412, a buffer 4413, and an entropy encoding unit 4414.

[0089] In other examples, video encoder 4400 may include more, fewer, or different functional components. In an example, prediction unit 4402 may include an intra block copy (IBC) unit. The IBC unit may perform prediction in an IBC mode in which at least one reference picture is a picture where the current video block is located.

[0090] Furthermore, some components, such as motion estimation unit 4404 and motion compensation unit 4405 may be highly integrated, but are represented in the example of video encoder 4400 separately for purposes of explanation.

[0091] Partition unit 4401 may partition a picture into one or more video blocks. Video encoder 4400 and video decoder 4500 may support various video block sizes.

[0092] Mode select unit 4403 may select one of the coding modes, intra or inter, e.g., based on error results, and provide the resulting intra or inter coded block to a residual generation unit 4407 to generate residual block data and to a reconstruction unit 4412 to reconstruct the encoded block for use as a reference picture. In some examples, mode select unit 4403 may select a combination of intra and inter prediction (CIIP) mode in which the prediction is based on an inter prediction signal and an intra prediction signal. Mode select unit 4403 may also select a resolution for a motion vector (e.g., a sub-pixel or integer pixel precision) for the block in the case of inter prediction.

[0093] To perform inter prediction on a current video block, motion estimation unit 4404 may generate motion information for the current video block by comparing one or more reference frames from buffer 4413 to the current video block. Motion compensation unit 4405 may determine a predicted video block for the current video block based on the motion information and decoded samples of pictures from buffer 4413 other than the picture associated with the current video block.

[0094] Motion estimation unit 4404 and motion compensation unit 4405 may perform different operations for a current video block, for example, depending on whether the current video block is in an I slice, a P slice, or a B slice.

[0095] In some examples, motion estimation unit 4404 may perform uni-directional prediction for the current video block, and motion estimation unit 4404 may search reference pictures of list 0 or list 1 for a reference video block for the current video block. Motion estimation unit 4404 may then generate a reference index that indicates the reference picture in list 0 or list 1 that contains the

reference video block and a motion vector that indicates a spatial displacement between the current video block and the reference video block. Motion estimation unit 4404 may output the reference index, a prediction direction indicator, and the motion vector as the motion information of the current video block. Motion compensation unit 4405 may generate the predicted video block of the current block based on the reference video block indicated by the motion information of the current video block.

[0096] In other examples, motion estimation unit 4404 may perform bi-directional prediction for the current video block, motion estimation unit 4404 may search the reference pictures in list 0 for a reference video block for the current video block and may also search the reference pictures in list 1 for another reference video block for the current video block. Motion estimation unit 4404 may then generate reference indexes that indicate the reference pictures in list 0 and list 1 containing the reference video blocks and motion vectors that indicate spatial displacements between the reference video blocks and the current video block. Motion estimation unit 4404 may output the reference indexes and the motion vectors of the current video block as the motion information of the current video block. Motion compensation unit 4405 may generate the predicted video block of the current video block based on the reference video blocks indicated by the motion information of the current video block.

[0097] In some examples, motion estimation unit 4404 may output a full set of motion information for decoding processing of a decoder. In some examples, motion estimation unit 4404 may not output a full set of motion information for the current video. Rather, motion estimation unit 4404 may signal the motion information of the current video block with reference to the motion information of another video block. For example, motion estimation unit 4404 may determine that the motion information of the current video block is sufficiently similar to the motion information of a neighboring video block.

[0098] In one example, motion estimation unit 4404 may indicate, in a syntax structure associated with the current video block, a value that indicates to the video decoder 4500 that the current video block has the same motion information as another video block.

[0099] In another example, motion estimation unit 4404 may identify, in a syntax structure associated with the current video block, another video block and a motion vector difference (MVD). The motion vector difference indicates a difference between the motion vector of the current video block and the motion vector of the indicated video block. The video decoder 4500 may use the

motion vector of the indicated video block and the motion vector difference to determine the motion vector of the current video block.

[00100] As discussed above, video encoder 4400 may predictively signal the motion vector. Two examples of predictive signaling techniques that may be implemented by video encoder 4400 include advanced motion vector prediction (AMVP) and merge mode signaling.

[00101] Intra prediction unit 4406 may perform intra prediction on the current video block. When intra prediction unit 4406 performs intra prediction on the current video block, intra prediction unit 4406 may generate prediction data for the current video block based on decoded samples of other video blocks in the same picture. The prediction data for the current video block may include a predicted video block and various syntax elements.

[00102] Residual generation unit 4407 may generate residual data for the current video block by subtracting the predicted video block(s) of the current video block from the current video block. The residual data of the current video block may include residual video blocks that correspond to different sample components of the samples in the current video block.

[00103] In other examples, there may be no residual data for the current video block for the current video block, for example in a skip mode, and residual generation unit 4407 may not perform the subtracting operation.

[00104] Transform processing unit 4408 may generate one or more transform coefficient video blocks for the current video block by applying one or more transforms to a residual video block associated with the current video block.

[00105] After transform processing unit 4408 generates a transform coefficient video block associated with the current video block, quantization unit 4409 may quantize the transform coefficient video block associated with the current video block based on one or more quantization parameter (QP) values associated with the current video block.

[00106] Inverse quantization unit 4410 and inverse transform unit 4411 may apply inverse quantization and inverse transforms to the transform coefficient video block, respectively, to reconstruct a residual video block from the transform coefficient video block. Reconstruction unit 4412 may add the reconstructed residual video block to corresponding samples from one or more predicted video blocks generated by the prediction unit 4402 to produce a reconstructed video block associated with the current block for storage in the buffer 4413.

[00107] After reconstruction unit 4412 reconstructs the video block, the loop filtering operation may be performed to reduce video blocking artifacts in the video block.

[00108] Entropy encoding unit 4414 may receive data from other functional components of the video encoder 4400. When entropy encoding unit 4414 receives the data, entropy encoding unit 4414 may perform one or more entropy encoding operations to generate entropy encoded data and output a bitstream that includes the entropy encoded data.

[00109] FIG. 11 is a block diagram illustrating an example of video decoder 4500 which may be video decoder 4324 in the system 4300 illustrated in FIG. 9. The video decoder 4500 may be configured to perform any or all of the techniques of this disclosure. In the example shown, the video decoder 4500 includes a plurality of functional components. The techniques described in this disclosure may be shared among the various components of the video decoder 4500. In some examples, a processor may be configured to perform any or all of the techniques described in this disclosure.

[00110] In the example shown, video decoder 4500 includes an entropy decoding unit 4501, a motion compensation unit 4502, an intra prediction unit 4503, an inverse quantization unit 4504, an inverse transformation unit 4505, a reconstruction unit 4506, and a buffer 4507. Video decoder 4500 may, in some examples, perform a decoding pass generally reciprocal to the encoding pass described with respect to video encoder 4400.

[00111] Entropy decoding unit 4501 may retrieve an encoded bitstream. The encoded bitstream may include entropy coded video data (e.g., encoded blocks of video data). Entropy decoding unit 4501 may decode the entropy coded video data, and from the entropy decoded video data, motion compensation unit 4502 may determine motion information including motion vectors, motion vector precision, reference picture list indexes, and other motion information. Motion compensation unit 4502 may, for example, determine such information by performing the AMVP and merge mode.

[00112] Motion compensation unit 4502 may produce motion compensated blocks, possibly performing interpolation based on interpolation filters. Identifiers for interpolation filters to be used with sub-pixel precision may be included in the syntax elements.

[00113] Motion compensation unit 4502 may use interpolation filters as used by video encoder 4400 during encoding of the video block to calculate interpolated values for sub-integer pixels of a reference block. Motion compensation unit 4502 may determine the interpolation filters used by

video encoder 4400 according to received syntax information and use the interpolation filters to produce predictive blocks.

[00114] Motion compensation unit 4502 may use some of the syntax information to determine sizes of blocks used to encode frame(s) and/or slice(s) of the encoded video sequence, partition information that describes how each macroblock of a picture of the encoded video sequence is partitioned, modes indicating how each partition is encoded, one or more reference frames (and reference frame lists) for each inter coded block, and other information to decode the encoded video sequence.

[00115] Intra prediction unit 4503 may use intra prediction modes for example received in the bitstream to form a prediction block from spatially adjacent blocks. Inverse quantization unit 4504 inverse quantizes, i.e., de-quantizes, the quantized video block coefficients provided in the bitstream and decoded by entropy decoding unit 4501. Inverse transform unit 4505 applies an inverse transform.

[00116] Reconstruction unit 4506 may sum the residual blocks with the corresponding prediction blocks generated by motion compensation unit 4502 or intra prediction unit 4503 to form decoded blocks. If desired, a deblocking filter may also be applied to filter the decoded blocks in order to remove blockiness artifacts. The decoded video blocks are then stored in buffer 4507, which provides reference blocks for subsequent motion compensation/intra prediction and also produces decoded video for presentation on a display device.

[00117] FIG. 12 is a schematic diagram of an example encoder 4600. The encoder 4600 is suitable for implementing the techniques of VVC. The encoder 4600 includes three in-loop filters, namely a deblocking filter (DF) 4602, a sample adaptive offset (SAO) 4604, and an adaptive loop filter (ALF) 4606. Unlike the DF 4602, which uses predefined filters, the SAO 4604 and the ALF 4606 utilize the original samples of the current picture to reduce the mean square errors between the original samples and the reconstructed samples by adding an offset and by applying a finite impulse response (FIR) filter, respectively, with coded side information signaling the offsets and filter coefficients. The ALF 4606 is located at the last processing stage of each picture and can be regarded as a tool trying to catch and fix artifacts created by the previous stages.

[00118] The encoder 4600 further includes an intra prediction component 4608 and a motion estimation/motion compensation (ME/MC) component 4610 configured to receive input video. The intra prediction component 4608 is configured to perform intra prediction, while the ME/MC

component 4610 is configured to utilize reference pictures obtained from a reference picture buffer 4612 to perform inter prediction. Residual blocks from inter prediction or intra prediction are fed into a transform (T) component 4614 and a quantization (Q) component 4616 to generate quantized residual transform coefficients, which are fed into an entropy coding component 4618. The entropy coding component 4618 entropy codes the prediction results and the quantized transform coefficients and transmits the same toward a video decoder (not shown). Quantization components output from the quantization component 4616 may be fed into an inverse quantization (IQ) components 4620, an inverse transform component 4622, and a reconstruction (REC) component 4624. The REC component 4624 is able to output images to the DF 4602, the SAO 4604, and the ALF 4606 for filtering prior to those images being stored in the reference picture buffer 4612.

[00119] FIG. 13 is a flowchart for an example method 4700 of video processing. The method 4700 determines a media track includes a track reference type of associated external stream track (aest) at step 4702. A conversion is performed between a media data and the media data file based on the media track at step 4704. The conversion may include encoding at an encoder, decoding at a decoder, or combinations thereof.

[00120] It should be noted that the method 4700 can be implemented in an apparatus for processing video data comprising a processor and a non-transitory memory with instructions thereon, such as video encoder 4400, video decoder 4500, and/or encoder 4600. In such a case, the instructions upon execution by the processor, cause the processor to perform the method 4700. Further, the method 4700 can be performed by a non-transitory computer readable medium comprising a computer program product for use by a video coding device. The computer program product comprises computer executable instructions stored on the non-transitory computer readable medium such that when executed by a processor cause the video coding device to perform the method 4700.

[00121] A listing of solutions preferred by some examples is provided next.

[00122] The following solutions show examples of techniques discussed herein.

[00123] The following solutions show example embodiments of techniques discussed in the previous section (e.g., item 1).

[00124] 1. A method for processing media data comprising: determining (4202) an extended dependent random access point (EDRAP) sample group references audio data; and performing (4204) a conversion between a media data and a media data file based on the EDRAP sample.

[00125] 2. The method of solution 1, wherein an associated external stream track (aest) track contains the audio data.

[00126] The following solutions show example embodiments of techniques discussed in the previous section (e.g., item 2).

[00127] 3. A method for processing video data comprising: determining an extended dependent random access point (EDRAP) sample in a track containing EDRAP group signaling is disallowed from being omitted from EDRAP sample groups; and performing a conversion between a media data and the media data file based on the EDRAP sample.

[00128] 4. The method of any of solutions 1-3, wherein the EDRAP sample is a member of an EDRAP sample group.

[00129] 5. The method of any of solutions 1-4, wherein each sample in a track is required to be a member of an EDRAP sample group when the track has a sample to group box (SampleToGroupBox) with a grouping type (grouping_type) equal to EDRAP type (edrp).

[00130] The following solutions show example embodiments of techniques discussed in the previous section (e.g., item 3).

[00131] 6. A method for processing video data comprising: determining a track containing an associated external stream track (aest) reference type is disallowed from containing no sample that is a member of an extended dependent random access point (EDRAP) sample group; and performing a conversion between a media data and a media data file based on the EDRAP sample group.

[00132] 7. The method of any of solutions 1-6, wherein the track containing the aest reference type is required to contain at least one sample that is a member of an EDRAP sample group.

[00133] 8. The method of any of solutions 1-7, wherein the track containing the aest reference type is required to contain at least one sample that is associated with an EDRAP sample group.

[00134] 9. An apparatus for processing video data comprising: a processor; and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to perform the method of any of solutions 1-8.

[00135] 10. A non-transitory computer readable medium comprising a computer program product for use by a video coding device, the computer program product comprising computer executable instructions stored on the non-transitory computer readable medium such that when executed by a processor cause the video coding device to perform the method of any of solutions 1-8.

[00136] 11. A non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method performed by a video processing apparatus, wherein the method comprises: determining a track containing an associated external stream track (aest) reference type is disallowed from containing no sample that is a member of an extended dependent random access point (EDRAP) sample group; and generating a bitstream based on the determining.

[00137] 12. A method for storing bitstream of a video comprising: determining a track containing an associated external stream track (aest) reference type is disallowed from containing no sample that is a member of an extended dependent random access point (EDRAP) sample group; generating a bitstream based on the determining; and storing the bitstream in a non-transitory computer-readable recording medium.

[00138] 13. A method, apparatus or system described in the present document.

[00139] The following solutions show example embodiments of techniques discussed in the previous sections (e.g., FIG. 13).

[00140] 1. A method for processing media data comprising: determining a media track includes a track reference type of associated external stream track (aest); and performing a conversion between a media data and a media data file based on the media track.

[00141] 2. The method of solution 1, wherein the media track with the track reference type of aest includes an extended dependent random access point (EDRAP) sample.

[00142] 3. The method of any of solutions 1-2, wherein the EDRAP sample is associated with an EDRAP sample group.

[00143] 4. The method of any of solutions 1-3, wherein the EDRAP sample in the media data track is a video sample.

[00144] 5. The method of any of solutions 1-3, wherein the EDRAP sample in the media data track is an audio sample.

[00145] 6. The method of any of solutions 1-5, wherein any EDRAP sample in a track containing EDRAP group signaling is disallowed from being omitted from EDRAP sample groups.

[00146] 7. The method of any of solutions 1-6, wherein each EDRAP sample in a track containing EDRAP group signaling is required to be a member of an EDRAP sample group.

[00147] 8. The method of any of solutions 1-7, wherein each sample in a track is required to be a member of an EDRAP sample group when the track has a sample to group box (SampleToGroupBox) with a grouping type (grouping_type) equal to EDRAP type (edrp).

- [00148] 9. The method of any of solutions 1-8, wherein the media track containing the aest reference type is disallowed from containing no sample that is a member of an EDRAP sample group.
- [00149] 10. The method of any of solutions 1-9, wherein the media track containing the aest reference type is required to contain at least one sample that is a member of an EDRAP sample group.
- [00150] 11. The method of any of solutions 1-10, wherein the track containing the aest reference type is required to contain at least one sample that is associated with an EDRAP sample group.
- [00151] 12. The method of any of solutions 1-11, wherein the conversion comprises generating the media data file from the media data.
- [00152] 13. The method of any of solutions 1-11, wherein the conversion comprises parsing the media data from the media data file.
- [00153] 14. A non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method performed by a video processing apparatus, wherein the method comprises: determining a media track includes a track reference type of associated external stream track (aest); and generating a bitstream based on the determining.
- [00154] 15. The method of solution 14, wherein the media track with the track reference type of aest includes an extended dependent random access point (EDRAP) sample.
- [00155] 16. The method of any of solutions 14-15, wherein the EDRAP sample is associated with an EDRAP sample group.
- [00156] 17. The method of any of solutions 14-16, wherein the EDRAP sample in the media data track is a video sample.
- [00157] 18. The method of any of solutions 14-17, wherein the EDRAP sample in the media data track is an audio sample.
- [00158] 19. The method of any of solutions 14-18, wherein any EDRAP sample in a track containing EDRAP group signaling is disallowed from being omitted from EDRAP sample groups.
- [00159] 20. An apparatus for processing video data comprising: at least one processor; and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the at least one processor, cause the at least one processor to perform the method of any of solutions 1-13.
- [00160] 21. A non-transitory computer readable medium comprising a computer program product for use by a video coding device, the computer program product comprising computer

executable instructions stored on the non-transitory computer readable medium such that when executed by a processor cause the video coding device to perform the method of any of solutions 1-13.

[00161] 22. A method for storing bitstream of a video comprising: determining a media track includes a track reference type of associated external stream track (aest); generating a bitstream based on the determining; and storing the bitstream in a non-transitory computer-readable recording medium.

[00162] 23. A method, apparatus or system described in the present document.

[00163] In the solutions described herein, an encoder may conform to the format rule by producing a coded representation according to the format rule. In the solutions described herein, a decoder may use the format rule to parse syntax elements in the coded representation with the knowledge of presence and absence of syntax elements according to the format rule to produce decoded video.

[00164] In the present document, the term “video processing” may refer to video encoding, video decoding, video compression or video decompression. For example, video compression algorithms may be applied during conversion from pixel representation of a video to a corresponding bitstream representation or vice versa. The bitstream representation of a current video block may, for example, correspond to bits that are either co-located or spread in different places within the bitstream, as is defined by the syntax. For example, a macroblock may be encoded in terms of transformed and coded error residual values and also using bits in headers and other fields in the bitstream. Furthermore, during conversion, a decoder may parse a bitstream with the knowledge that some fields may be present, or absent, based on the determination, as is described in the above solutions. Similarly, an encoder may determine that certain syntax fields are or are not to be included and generate the coded representation accordingly by including or excluding the syntax fields from the coded representation.

[00165] The disclosed and other solutions, examples, embodiments, modules and the functional operations described in this document can be implemented in digital electronic circuitry, or in computer software, firmware, or hardware, including the structures disclosed in this document and their structural equivalents, or in combinations of one or more of them. The disclosed and other embodiments can be implemented as one or more computer program products, i.e., one or more modules of computer program instructions encoded on a computer readable medium for execution

by, or to control the operation of, data processing apparatus. The computer readable medium can be a machine-readable storage device, a machine-readable storage substrate, a memory device, a composition of matter effecting a machine-readable propagated signal, or a combination of one or more them. The term “data processing apparatus” encompasses all apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. The apparatus can include, in addition to hardware, code that creates an execution environment for the computer program in question, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of one or more of them. A propagated signal is an artificially generated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal, that is generated to encode information for transmission to suitable receiver apparatus.

[00166] A computer program (also known as a program, software, software application, script, or code) can be written in any form of programming language, including compiled or interpreted languages, and it can be deployed in any form, including as a stand-alone program or as a module, component, subroutine, or other unit suitable for use in a computing environment. A computer program does not necessarily correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data (e.g., one or more scripts stored in a markup language document), in a single file dedicated to the program in question, or in multiple coordinated files (e.g., files that store one or more modules, sub programs, or portions of code). A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

[00167] The processes and logic flows described in this document can be performed by one or more programmable processors executing one or more computer programs to perform functions by operating on input data and generating output. The processes and logic flows can also be performed by, and apparatus can also be implemented as, special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit).

[00168] Processors suitable for the execution of a computer program include, by way of example, both general and special purpose microprocessors, and any one or more processors of any kind of digital computer. Generally, a processor will receive instructions and data from a read only memory or a random-access memory or both. The essential elements of a computer are a processor for performing instructions and one or more memory devices for storing instructions and data.

Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto optical disks, or optical disks. However, a computer need not have such devices. Computer readable media suitable for storing computer program instructions and data include all forms of non-volatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., erasable programmable read-only memory (EPROM), electrically erasable programmable read-only memory (EEPROM), and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto optical disks; and compact disc read-only memory (CD ROM) and Digital versatile disc-read only memory (DVD-ROM) disks. The processor and the memory can be supplemented by, or incorporated in, special purpose logic circuitry.

[00169] While this patent document contains many specifics, these should not be construed as limitations on the scope of any subject matter or of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments of particular techniques. Certain features that are described in this patent document in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

[00170] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. Moreover, the separation of various system components in the embodiments described in this patent document should not be understood as requiring such separation in all embodiments.

[00171] Only a few implementations and examples are described and other implementations, enhancements and variations can be made based on what is described and illustrated in this patent document.

[00172] A first component is directly coupled to a second component when there are no intervening components, except for a line, a trace, or another medium between the first component and the second component. The first component is indirectly coupled to the second component when

there are intervening components other than a line, a trace, or another medium between the first component and the second component. The term “coupled” and its variants include both directly coupled and indirectly coupled. The use of the term “about” means a range including $\pm 10\%$ of the subsequent number unless otherwise stated.

[00173] While several embodiments have been provided in the present disclosure, it should be understood that the disclosed systems and methods might be embodied in many other specific forms without departing from the spirit or scope of the present disclosure. The present examples are to be considered as illustrative and not restrictive, and the intention is not to be limited to the details given herein. For example, the various elements or components may be combined or integrated in another system or certain features may be omitted, or not implemented.

[00174] In addition, techniques, systems, subsystems, and methods described and illustrated in the various embodiments as discrete or separate may be combined or integrated with other systems, modules, techniques, or methods without departing from the scope of the present disclosure. Other items shown or discussed as coupled may be directly connected or may be indirectly coupled or communicating through some interface, device, or intermediate component whether electrically, mechanically, or otherwise. Other examples of changes, substitutions, and alterations are ascertainable by one skilled in the art and could be made without departing from the spirit and scope disclosed herein.

CLAIMS

What is claimed is:

1. A method for processing media data comprising:
determining a media track includes a track reference type of associated external stream track (aest); and
performing a conversion between a media data and a media data file based on the media track.
2. The method of claim 1, wherein the media track with the track reference type of aest includes an extended dependent random access point (EDRAP) sample.
3. The method of any of claims 1-2, wherein the EDRAP sample is associated with an EDRAP sample group.
4. The method of any of claims 1-3, wherein the EDRAP sample in the media data track is a video sample.
5. The method of any of claims 1-3, wherein the EDRAP sample in the media data track is an audio sample.
6. The method of any of claims 1-5, wherein any EDRAP sample in a track containing EDRAP group signaling is disallowed from being omitted from EDRAP sample groups.
7. The method of any of claims 1-6, wherein each EDRAP sample in a track containing EDRAP group signaling is required to be a member of an EDRAP sample group.
8. The method of any of claims 1-7, wherein each sample in a track is required to be a member of an EDRAP sample group when the track has a sample to group box (SampleToGroupBox) with a grouping type (grouping_type) equal to EDRAP type (edrp).

9. The method of any of claims 1-8, wherein the media track containing the aest reference type is disallowed from containing no sample that is a member of an EDRAP sample group.
10. The method of any of claims 1-9, wherein the media track containing the aest reference type is required to contain at least one sample that is a member of an EDRAP sample group.
11. The method of any of claims 1-10, wherein the track containing the aest reference type is required to contain at least one sample that is associated with an EDRAP sample group.
12. The method of any of claims 1-11, wherein the conversion comprises generating the media data file from the media data.
13. The method of any of claims 1-11, wherein the conversion comprises parsing the media data from the media data file.
14. A non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method performed by a video processing apparatus, wherein the method comprises:
 - determining a media track includes a track reference type of associated external stream track (aest); and
 - generating a bitstream based on the determining.
15. The method of claim 14, wherein the media track with the track reference type of aest includes an extended dependent random access point (EDRAP) sample.
16. The method of any of claims 14-15, wherein the EDRAP sample is associated with an EDRAP sample group.
17. The method of any of claims 14-16, wherein the EDRAP sample in the media data track is a video sample.

18. The method of any of claims 14-17, wherein the EDRAP sample in the media data track is an audio sample.
19. The method of any of claims 14-18, wherein any EDRAP sample in a track containing EDRAP group signaling is disallowed from being omitted from EDRAP sample groups.
20. An apparatus for processing video data comprising: at least one processor; and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the at least one processor, cause the at least one processor to perform the method of any of claims 1-13.
21. A non-transitory computer readable medium comprising a computer program product for use by a video coding device, the computer program product comprising computer executable instructions stored on the non-transitory computer readable medium such that when executed by a processor cause the video coding device to perform the method of any of claims 1-13.
22. A method for storing bitstream of a video comprising:
 - determining a media track includes a track reference type of associated external stream track (aest);
 - generating a bitstream based on the determining; and
 - storing the bitstream in a non-transitory computer-readable recording medium.
23. A method, apparatus or system described in the present document.

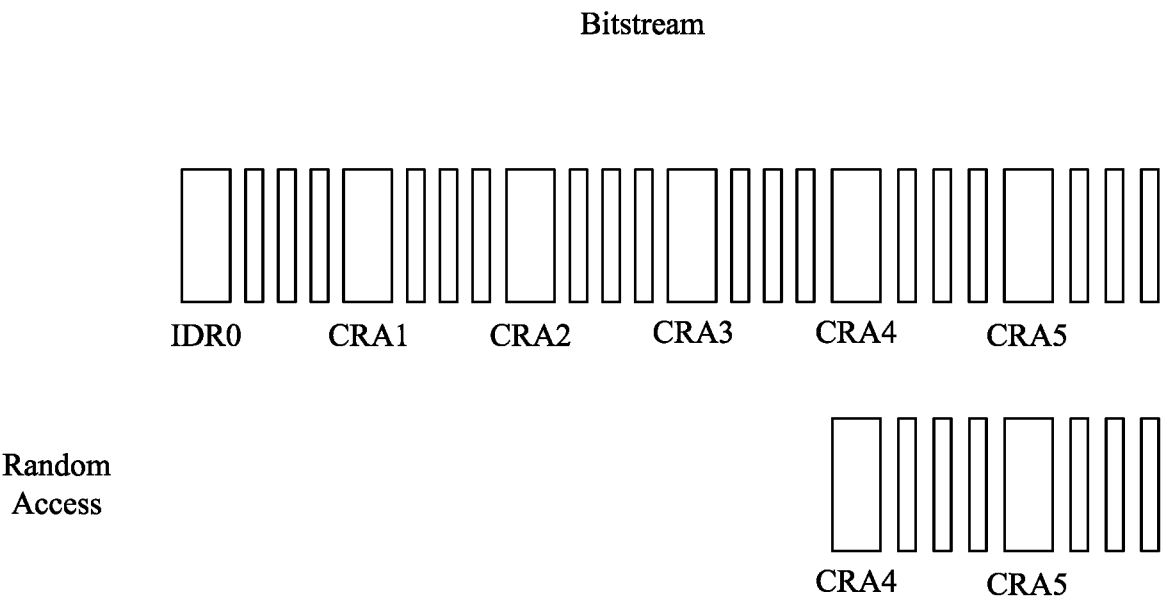


FIG. 1

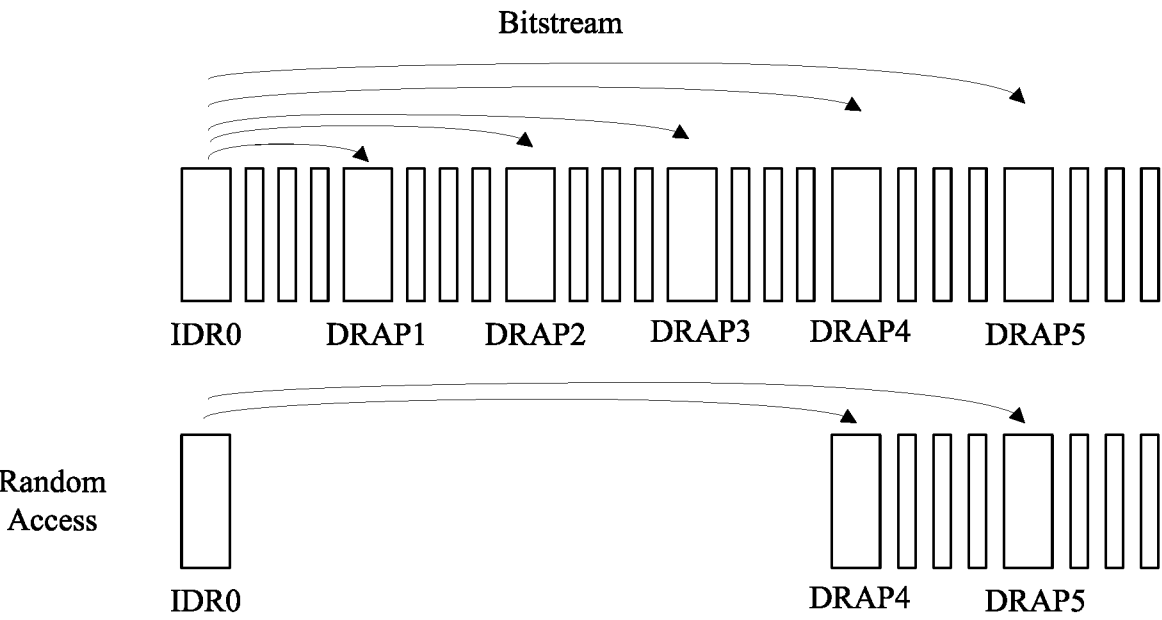


FIG. 2

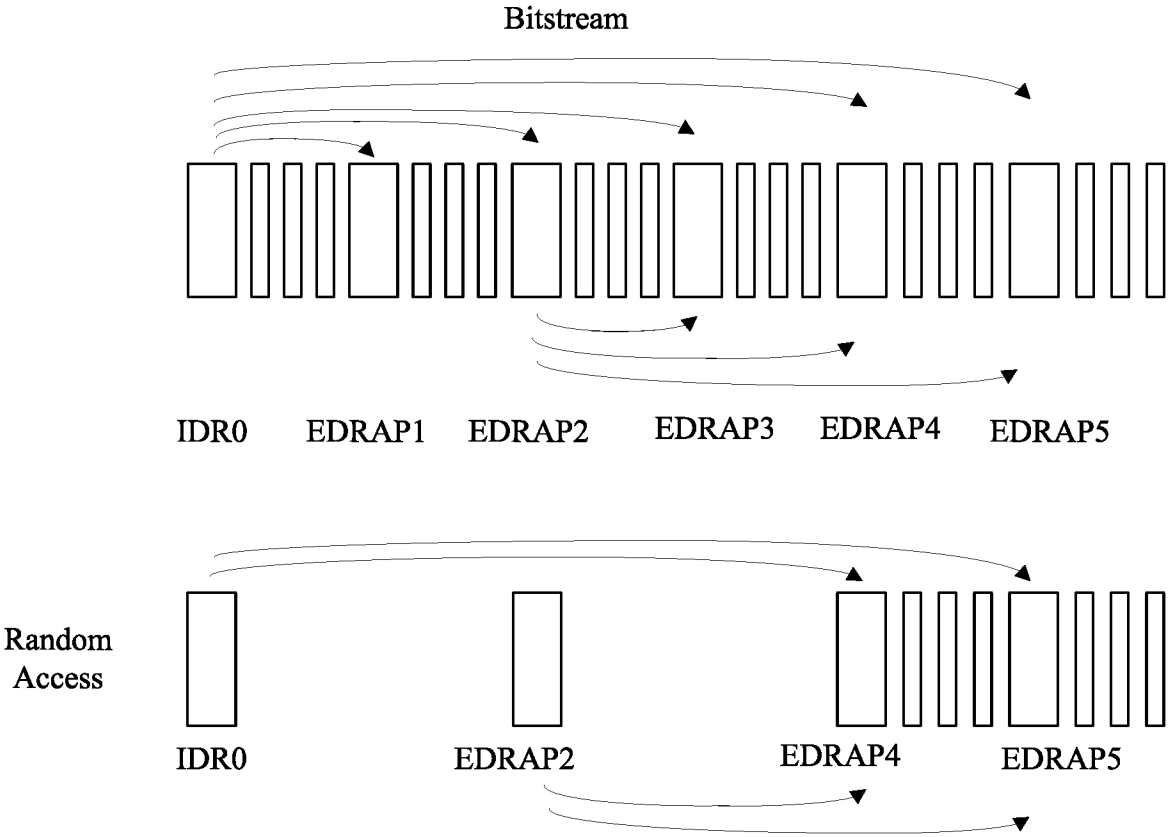


FIG. 3

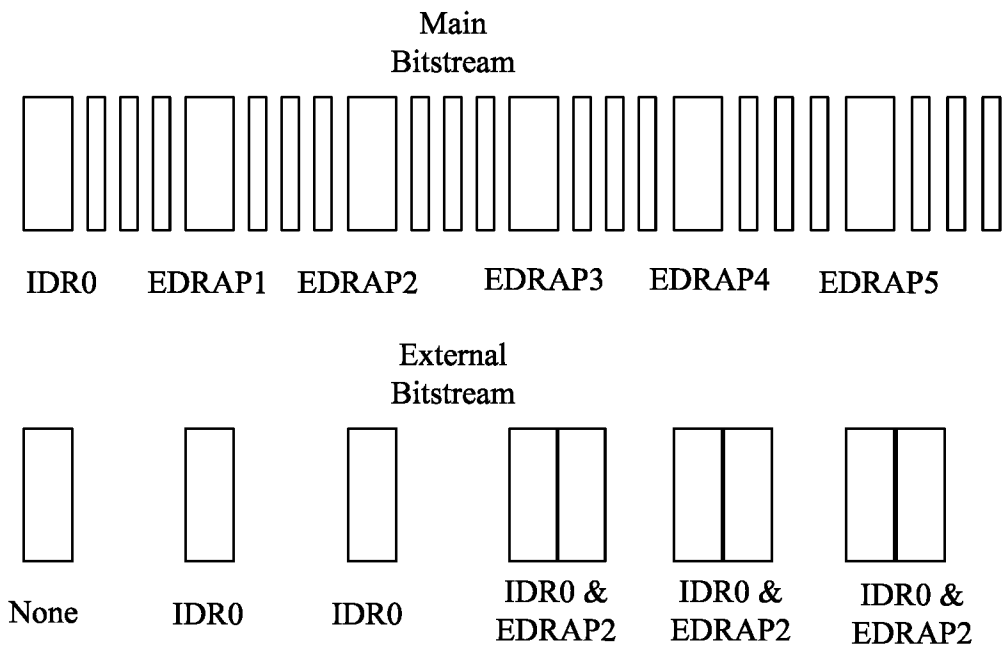


FIG. 4

Random Access from EDRAP4

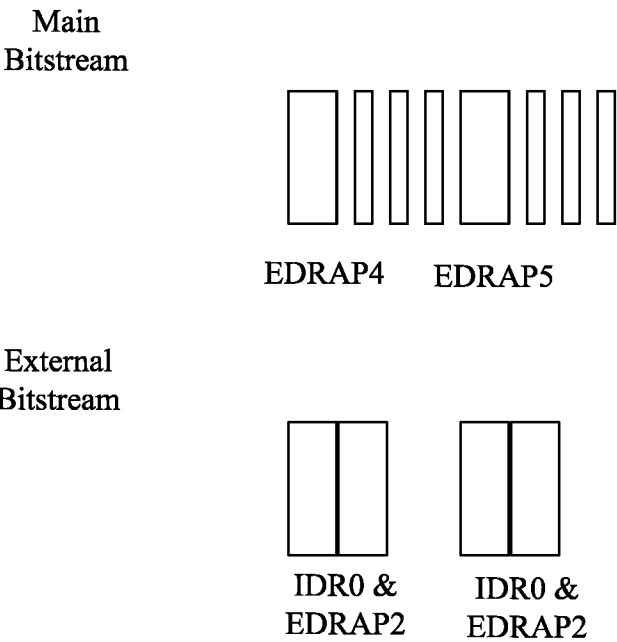


FIG. 5

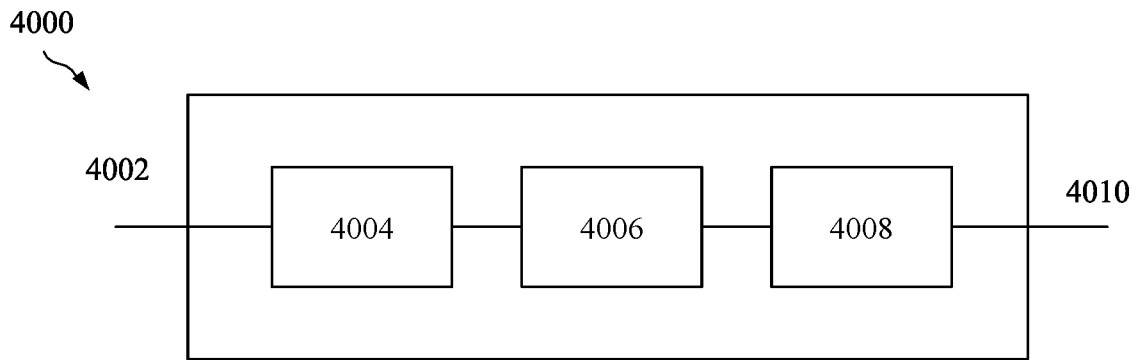


FIG. 6

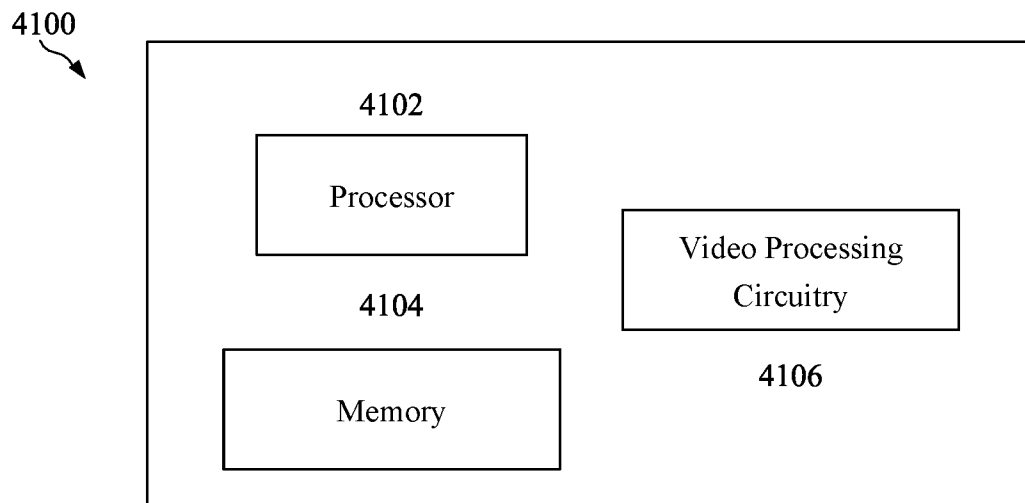


FIG. 7

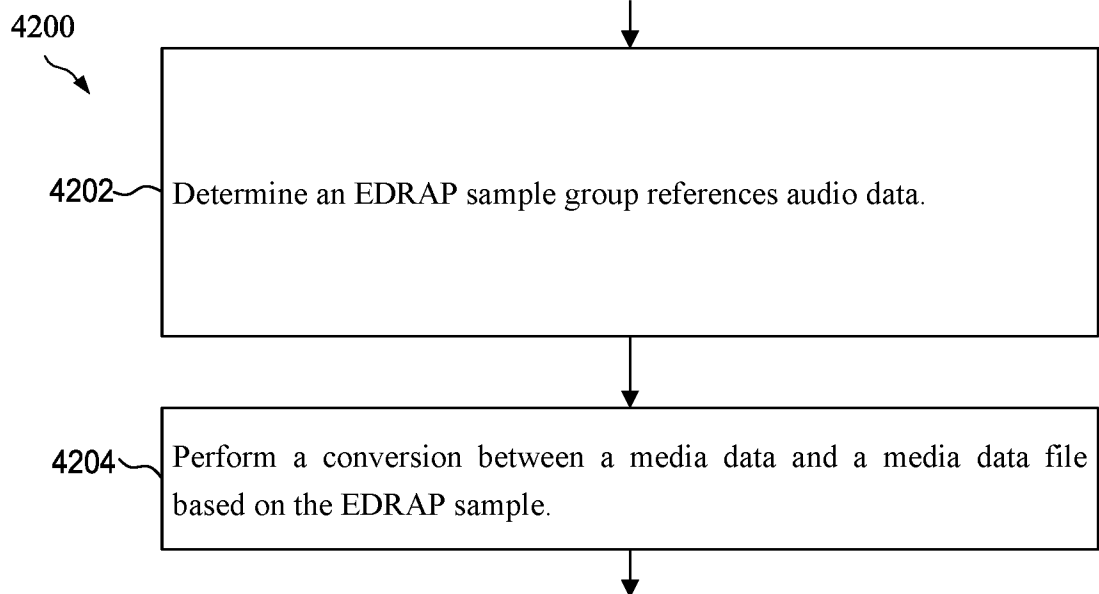


FIG. 8

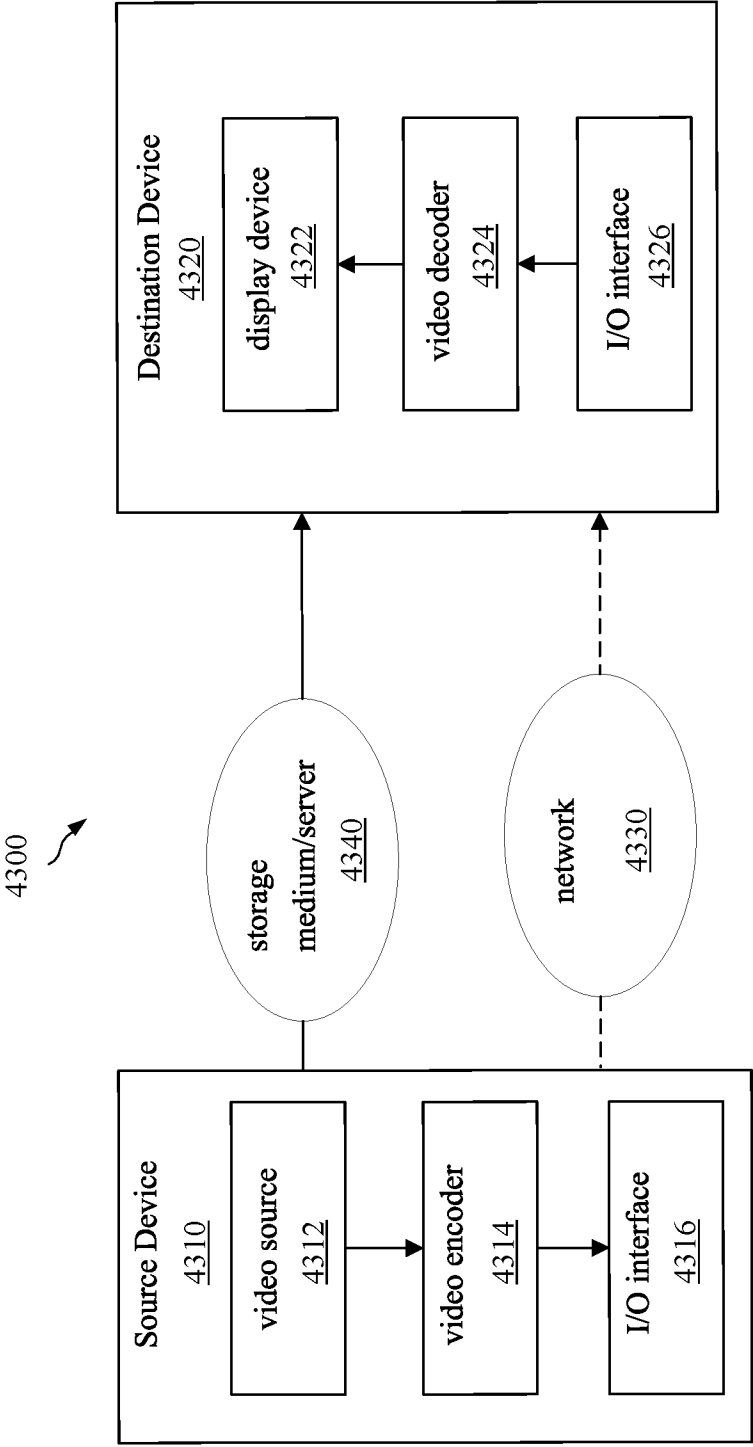


FIG. 9

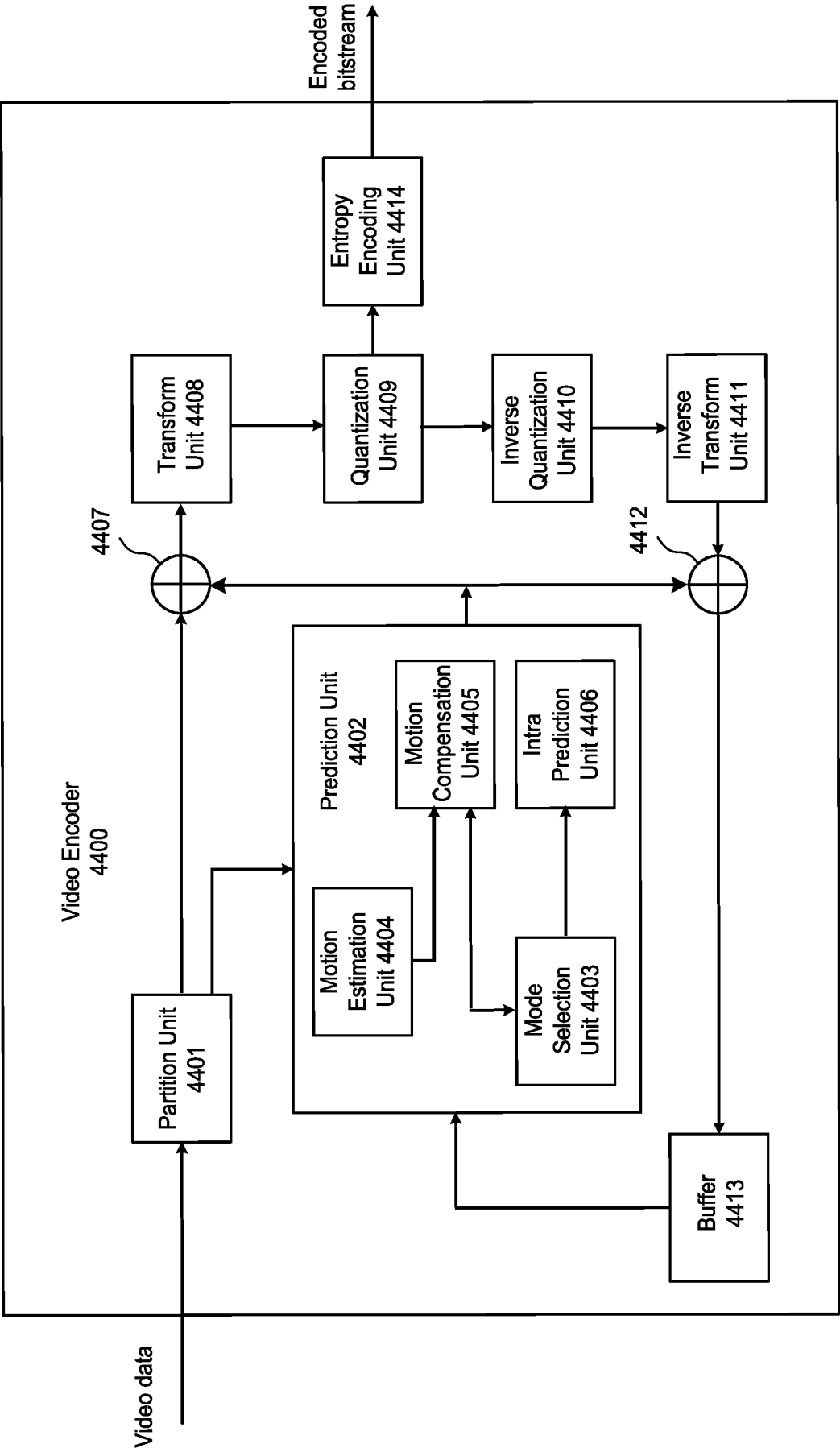


FIG. 10

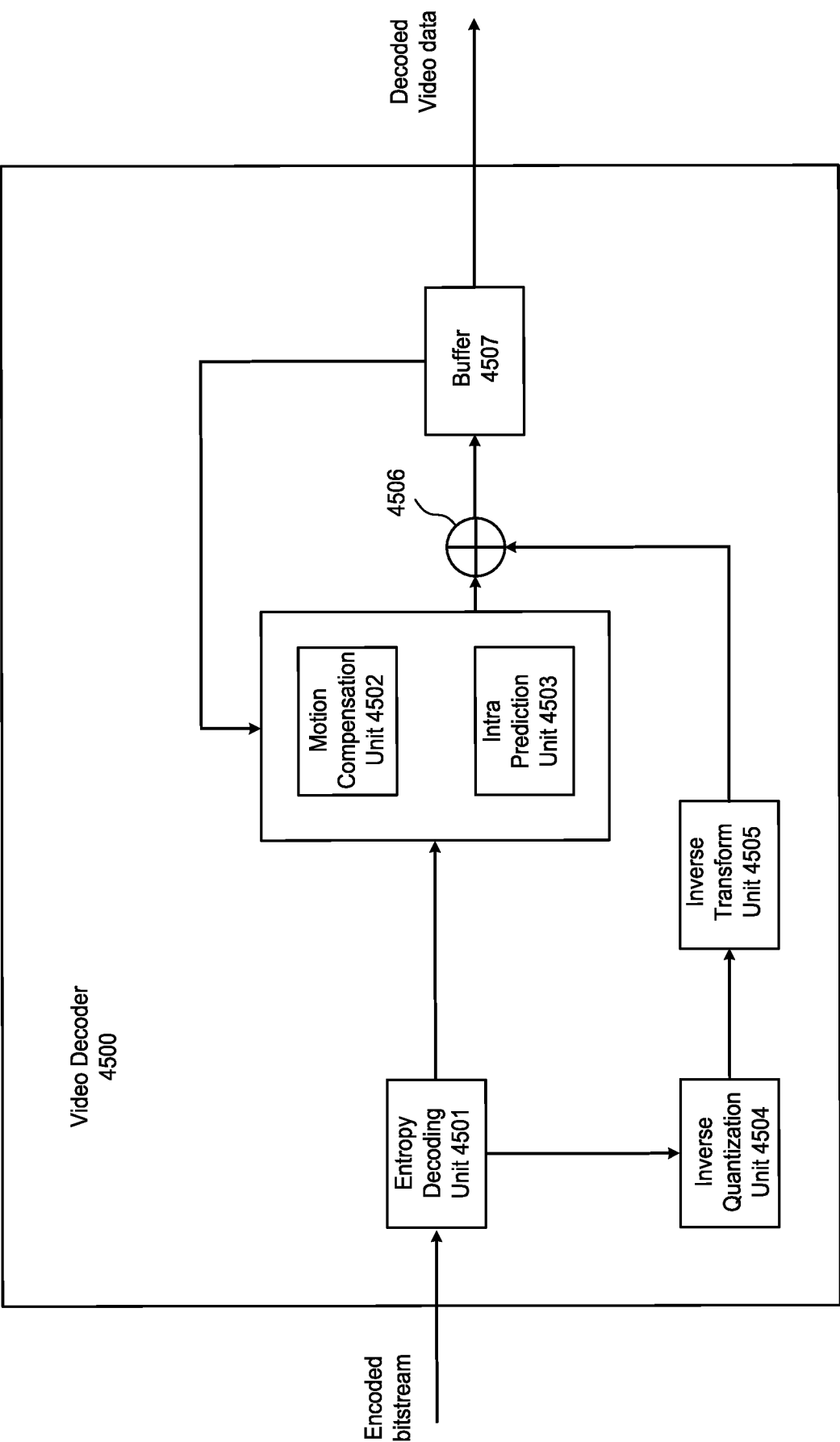


FIG. 11

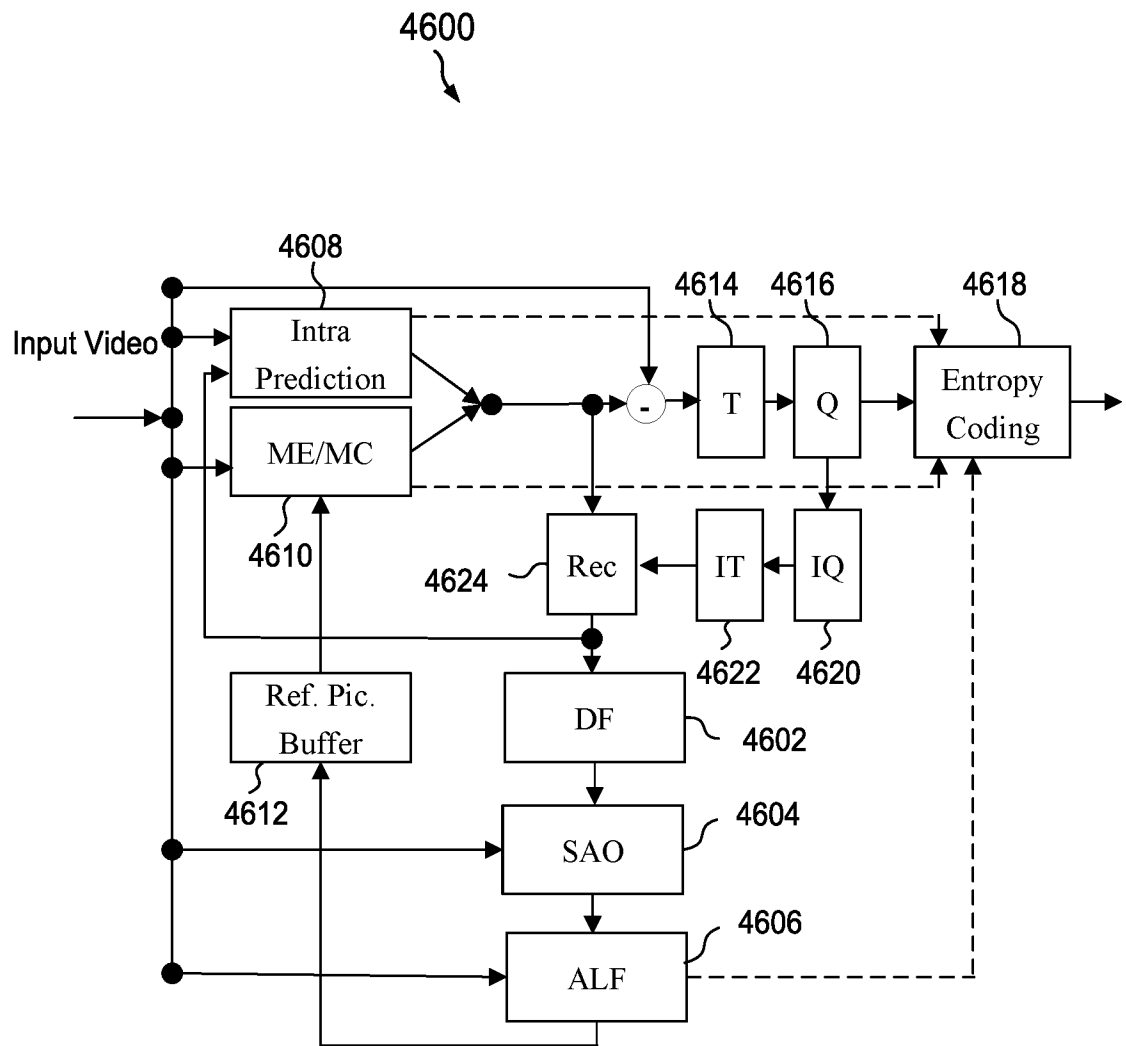


FIG. 12

9/9

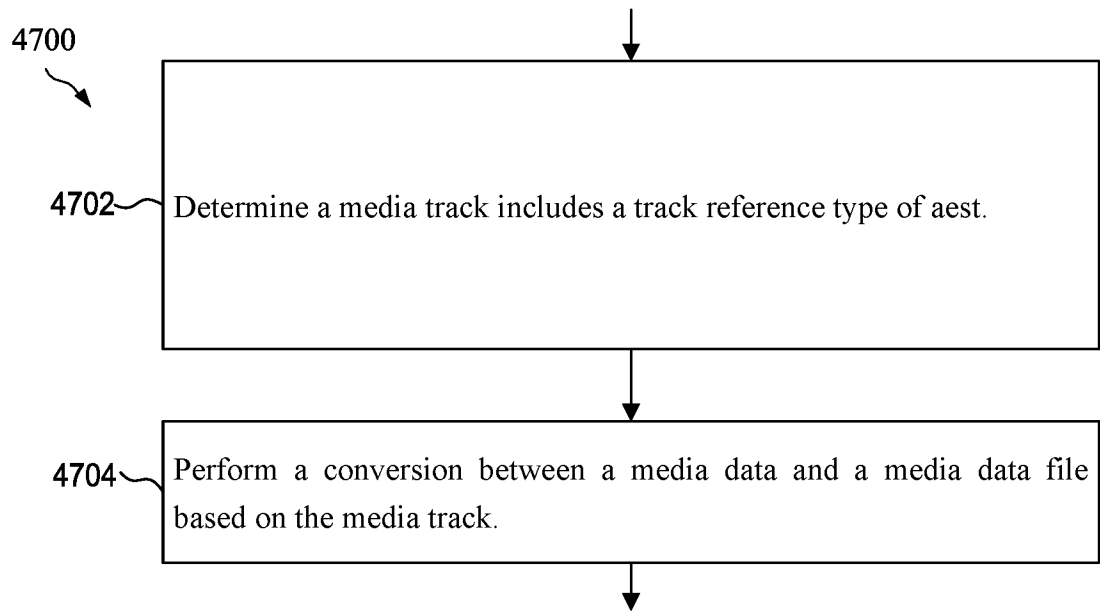


FIG. 13

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US 23/26358

A. CLASSIFICATION OF SUBJECT MATTER

IPC - INV. G06F 16/71 (2023.01)

ADD. H04N 19/70, H04N 21/2343, H04N 21/4402 (2023.01)

CPC - INV. G06F 16/71

ADD. H04N 19/70, H04N 21/23439, H04N 21/4402

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)

See Search History document

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

See Search History document

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

See Search History document

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2022/0103847 A1 (LEMON INC.), 31 March 2022 (31.03.2022), entire document, especially Abstract; para [0006], [0150], [0170]-[0171], [0195], [0305]	1-3, 14-16, 20-23
T	US 2022/0353317 A1 (LEMON INC.), 03 November 2022 (03.11.2022), entire document, especially Abstract; para [0038], [0049]-[0050], [0052]	1-3, 14-16, 20-23
T	WO 2023/137321 A2 (BYTEDANCE INC.), 20 July 2023 (20.07.2023), entire document, especially Abstract; para [0078], [0082]-[0083]	1-3, 14-16, 20-23

☐ 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

"D" document cited by the applicant in the international application

"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

01 September 2023

Date of mailing of the international search report

OCT 05 2023

Name and mailing address of the ISA/US

Mail Stop PCT, Attn: ISA/US, Commissioner for Patents
P.O. Box 1450, Alexandria, Virginia 22313-1450

Facsimile No. 571-273-8300

Authorized officer

Kari Rodriguez

Telephone No. PCT Helpdesk: 571-272-4300

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US 23/26358

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:
2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
3. ☒ Claims Nos.: 4-13, 17-19
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. ☐ As all searchable claims could be searched without effort justifying additional fees, this Authority did not invite payment of additional fees.
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

- ☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- ☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- ☐ No protest accompanied the payment of additional search fees.