



(51) International Patent Classification:

H04N 21/2343 (2011.01) *H04N 21/845* (2011.01)
H04N 21/238 (2011.01)

(21) International Application Number:

PCT/US2018/019575

(22) International Filing Date:

23 February 2018 (23.02.2018)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

62/462,873	23 February 2017 (23.02.2017)	US
62/534,170	18 July 2017 (18.07.2017)	US
62/550,517	25 August 2017 (25.08.2017)	US
15/902,971	22 February 2018 (22.02.2018)	US

(71) Applicant: **NETFLIX, INC.** [US/US]; 100 Winchester Circle, Los Gatos, California 95032 (US).

(72) Inventor: **KATSAVOUNIDIS, Ioannis**; 100 Winchester Circle, Los Gatos, California 95032 (US).

(74) Agent: **CAREY, John C.** et al.; Artegis Law Group, LLP, 7710 Cherry Park Drive, Suite T104, Houston, Texas 77095 (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, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, 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, 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: TECHNIQUES FOR SELECTING RESOLUTIONS FOR ENCODING DIFFERENT SHOT SEQUENCES

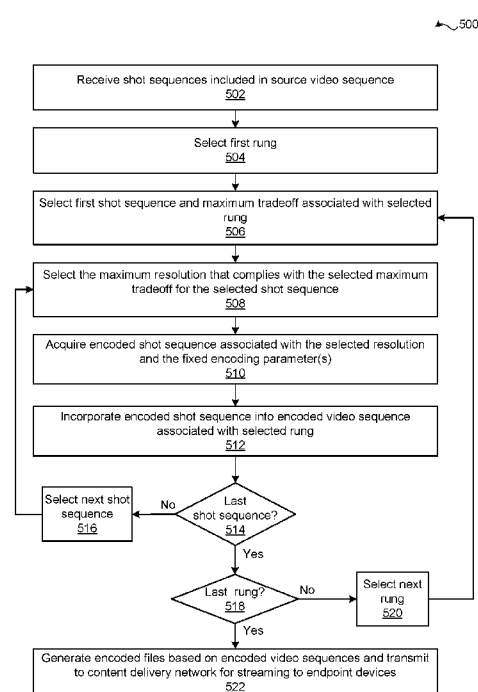


FIGURE 5

(57) Abstract: In various embodiments, an encoding optimization application encodes a source video sequence. The encoding optimization application partitions a source video sequence associated with a media title into multiple subsequences that include a first subsequence and a second subsequence. The encoding optimization application then generates a first encoded subsequence based on the first subsequence and a first resolution. The encoding optimization application generates a second encoded subsequence based on the second subsequence and a second resolution. Notably, the second resolution is higher than the first resolution. The encoding optimization application then aggregates the first encoded subsequence and the second encoded subsequence to generate an encoded video sequence. At least a portion of the encoded video sequence is subsequently streamed to an endpoint device during a playback of the media title.

TECHNIQUES FOR SELECTING RESOLUTIONS FOR ENCODING DIFFERENT SHOT SEQUENCES

CROSS-REFERENCE TO RELATED APPLICATIONS

5 [0001] This application claims the priority benefit of the United States Provisional Patent Application having Serial Number 62/462,873 (Attorney Docket Number NETF/0144USL) and filed on February 23, 2017, the United States Provisional Patent Application having Serial Number 62/534,170 (Attorney Docket Number NETF/0176USL) and filed on July 18, 2017, the United States Provisional Patent
10 Application having Serial Number 62/550,517 (Attorney Docket Number NETF/0176USL02) and filed on August 25, 2017, and the United States Patent Application having Serial Number 15/902,971 (Attorney Docket Number NETF0144US2) and filed on February 22, 2018. The subject matter of these related applications is hereby incorporated herein by reference.

15 **BACKGROUND**

Field of the Invention

[0002] Embodiments of the present invention relate generally to video technology and, more specifically, to techniques for selecting resolutions for encoding different shot sequences.

20 **Description of the Related Art**

[0003] A typical video streaming service provides access to a library of media titles that can be viewed on a range of different endpoint devices. Each endpoint device may connect to the video streaming service under different connection conditions. Some notable connection conditions include, without limitation, bandwidth and
25 latency. In addition, each different endpoint device may include different hardware for outputting the media title to end user(s). For example, a given endpoint device could include a display screen having a particular screen size and a particular screen resolution.

[0004] In many implementations, an endpoint device that connects to a video
30 streaming service executes an endpoint application that determines, for a given media title, an appropriate encoded version of the media title to stream to the endpoint device based on the connection conditions and the properties of the endpoint device. More specifically, the endpoint application attempts to select a

particular encoded version of the media title that provides the best visual quality during playback of the media title on the endpoint device while avoiding playback interruptions due to buffering or rebuffering.

[0005] In some implementations, the endpoint application selects the particular encoded version of the media title based on a bitrate ladder. The bitrate ladder is designed to achieve a target visual quality during playback of a media title based on an available bitrate. Each rung in the bitrate ladder specifies a different bitrate-resolution pair corresponding to a different pre-generated encoded version of the media title. To generate the encoded version of a media title corresponding to a given bitrate-resolution pair, the video service provider samples the video content associated with the media title at the resolution to generate sampled video content. The video service provider then encodes the sampled video content based on encoding parameters associated with the bitrate.

[0006] One drawback of the above encoding technique is that the complexity of the video content associated with a given media title oftentimes varies across the media title, but the resolution and encoding parameters used to encode the video content do not vary. As a result, encoding relatively simple portions of the media title may consume more computational and storage resources than are necessary to meet the target visual quality. For example, a relatively simple portion of a media title could have the same visual quality regardless of whether that portion of video content is encoded using a bitrate of 8 megabits per second or encoded using a bitrate of 20 megabits per second. Such encoding inefficiencies not only needlessly waste computational and storage resources, these types of inefficiencies also unnecessarily increase the bandwidth required to stream encoded versions of media titles to endpoint devices.

[0007] As the foregoing illustrates, what is needed in the art are more effective techniques for encoding video content for streaming.

SUMMARY OF THE EMBODIMENTS

[0008] One embodiment of the present invention sets forth a computer-implemented encoding video sequences. The method includes partitioning a source video sequence associated with a media title into multiple subsequences that include a first subsequence and a second subsequence; generating a first encoded subsequence

based on the first subsequence and a first resolution; generating a second encoded subsequence based on the second subsequence and a second resolution that is higher than the first resolution; and aggregating the first encoded subsequence and the second encoded subsequence to generate a first encoded video sequence, where
5 at least a portion of the first encoded video sequence is subsequently streamed to an endpoint device during a playback of the media title.

[0009] At least one technological improvement of the disclosed techniques relative to prior art is that varying the resolution when encoding individual shot sequences included in an encoded video sequence reduces encoding inefficiencies associated
10 with conventional encoding techniques. Such encoding inefficiencies not only needlessly waste computational and storage resources, these types of inefficiencies also unnecessarily increase the bandwidth required to stream encoded versions of media titles to endpoint devices.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] So that the manner in which the above recited features of the present invention can be understood in detail, a more particular description of the invention, briefly summarized above, may be had by reference to embodiments, some of which are illustrated in the appended drawings. It is to be noted, however, that the appended drawings illustrate only typical embodiments of this invention and are therefore not to
20 be considered limiting of its scope, for the invention may admit to other equally effective embodiments.

[0011] Figure 1 is a conceptual illustration of a system configured to implement one or more aspects of the present invention;

[0012] Figure 2 illustrates an example of a key frame location list that is generated by
25 the key frame optimizer of Figure 1, according to various embodiments of the present invention;

[0013] Figure 3 is a flow diagram of method steps for positioning key frames within encoded video sequences, according to various embodiments of the present invention;

[0014] Figure 4 is a more detailed illustration of the ladder optimizer of Figure 1,
30 according to various embodiments of the present invention;

[0015] Figure 5 is a flow diagram of method steps for encoding a source video sequence, according to various embodiments of the present invention;

[0016] Figure 6 is a more detailed illustration of the ladder optimizer of Figure 1 configured to vary a quantization parameter (QP) across a source video sequence,
5 according to various embodiments of the present invention; and

[0017] Figure 7 is a flow diagram of method steps for varying an encoding parameter when encoding a source video sequence, according to various embodiments of the present invention.

DETAILED DESCRIPTION

10 [0018] In the following description, numerous specific details are set forth to provide a more thorough understanding of the present invention. However, it will be apparent to one of skilled in the art that the present invention may be practiced without one or more of these specific details.

[0019] The disclosed techniques optimize operations associated with streaming
15 encoded video sequences for viewing an associated source video sequence associated with a media title. Notably, an encoding optimization application implements one or more encoding optimization techniques based on shot changes. Each shot change separates a different pair of shot sequences. Each shot sequence includes a set of frames that usually have similar spatial-temporal properties and run
20 for an uninterrupted period of time. First, a key frame optimizer partitions a source video sequence into multiple shot sequences based on the shot changes. In some embodiments, the key frame optimizer also determines optimized locations for key frames based on the shot changes. Each key frame is encoded interdependently of preceding frames. Subsequently, a ladder optimizer generates one or more encoded
25 video sequences based on resolution and/or an encoding parameter that is individually optimized for each shot sequence. Finally, a content engine generates encoded files based on the encoded video sequences and a file granularity (e.g., video sequence or shot sequence) and stores the encoded files in a content distribution network for distribution to endpoint devices.

30 [0020] Advantageously, during playback, the media title is switchable between decoded versions of different encoded video sequences at the key frames to optimize

a viewing experience based on any number of relevant criteria. Examples of relevant criteria include the current connection bandwidth, the current connection latency, the content of the upcoming shot sequence, and the like. Further, encoding each shot sequence based on a resolution and encoding parameter(s) that are optimized for the shot sequence reduces inefficiencies associated with conventional encoding techniques. Finally, storing encoded individual shot sequences as individual files may reduce the total resources required to store the encoded video sequences.

System Overview

[0021] Figure 1 is a conceptual illustration of a system 100 configured to implement one or more aspects of the present invention. As shown, the system 100 includes, without limitation, any number of compute instances 110 and a cloud 160. In general, the cloud 160 contains encapsulated shared resources, software, data, etc. For explanatory purposes, multiple instances of like objects are denoted with reference numbers identifying the object and parenthetical numbers identifying the instance where needed.

[0022] Any number of the compute instances 110 may reside outside the cloud 160 while other compute instances 110 may reside inside the cloud 160. In various embodiments, any number of the components of the system 100 may be distributed across multiple geographic locations. Further, any number of the components of the system 100 may be included in or distributed across one or more stand-alone device, distributed computing environments, or clouds 160 in any combination.

[0023] As shown, each of the compute instances 110 includes, without limitation, a processor 112 and a memory 116. The processor 112 may be any instruction execution system, apparatus, or device capable of executing instructions. For example, the processor 112 could comprise a central processing unit (CPU), a graphics processing unit (GPU), a controller, a microcontroller, a state machine, or any combination thereof. The memory 116 stores content, such as software applications and data, for use by the processor 112 of the compute instance 110.

[0024] The memory 116 may be one or more of a readily available memory, such as random access memory (RAM), read only memory (ROM), floppy disk, hard disk, or any other form of digital storage, local or remote. In some embodiments, a storage (not shown) may supplement or replace the memory 116. The storage may include

any number and type of external memories that are accessible to the processor 112. For example, and without limitation, the storage may include a Secure Digital Card, an external Flash memory, a portable compact disc read-only memory (CD-ROM), an optical storage device, a magnetic storage device, or any suitable combination of the foregoing.

[0025] In general, the compute instances 110 are configured to implement one or more applications. For explanatory purposes only, each application is depicted as residing in the memory 116 of a single compute instance 110 and executing on a processor 112 of the single compute instance 110. However, as persons skilled in the art will recognize, the functionality of each application may be distributed across any number of other applications that reside in the memories 116 of any number of compute instances 110 and execute on the processors 112 of any number of compute instances 110 in any combination. Further, the functionality of any number of applications may be consolidated into a single application or subsystem.

[0026] In particular, the compute instances 110 are configured to generate any number of encoded video sequences 142 for the source video sequence 122. The source video sequence 122 includes, without limitation, any amount and type of video content that is associated with a media title. Examples of video content include, without limitation, any portion (including all) of feature length films, episodes of television programs, and music videos, to name a few. Each of the encoded video sequences 142 includes, without limitation, encoded video content derived from the video content included in the associated source video sequence 122.

[0027] Although not shown, a video streaming service provides access to a library of media titles that can be viewed on a range of different endpoint devices. The library of media titles includes without limitation, the media title associated with the source video sequence 122. Each endpoint device may connect to the video streaming service under different connection conditions. Some notable connection conditions include, without limitation, bandwidth and latency. In addition, each different endpoint device may include different hardware for outputting the media title to end user(s). For example, a given endpoint device could include a display screen having a particular screen size and a particular screen resolution.

[0028] In many implementations, an endpoint device that connects to a video streaming service executes an endpoint application that determines, for a given media title, an appropriate encoded version of the media title to stream to the endpoint device based on the connection conditions and the properties of the endpoint device. More specifically, the endpoint application attempts to select a particular encoded version of the media title that provides the best visual quality during playback of the media title on the endpoint device while avoiding playback interruptions due to buffering or rebuffering.

[0029] In some conventional implementations, the endpoint application selects the particular encoded version of the media title based on a bitrate ladder. The bitrate ladder is designed to achieve a target visual quality during playback of a media title based on an available bitrate. Each rung in the bitrate ladder specifies a different bitrate-resolution pair corresponding to a different pre-generated encoded version of the media title. To generate the encoded version of a media title corresponding to a given bitrate-resolution pair, the video service provider samples the video content associated with the media title at the resolution to generate sampled video content. The video service provider then encodes the sampled video content based on encoding parameters associated with the bitrate.

[0030] One drawback of the above conventional encoding technique is that the complexity of the video content associated with a given media title oftentimes varies across the media title, but the resolution and encoding parameters used to encode the video content do not vary. As a result, encoding relatively simple portions of the media title may consume more computational and storage resources than are necessary to meet the target visual quality. For example, a relatively simple portion of a media title could have the same visual quality regardless of whether that portion of video content is encoded using a bitrate of 8 megabits per second or encoded using a bitrate of 20 megabits per second. Such encoding inefficiencies not only needlessly waste computational and storage resources, these types of inefficiencies also unnecessarily increase the bandwidth required to stream encoded versions of media titles to endpoint devices.

Optimizing Encoding Operations for Individual Shot Sequences

[0031] To address the above problems, the system 100 includes, without limitation, an encoding optimization application 120 that optimizes encoding operations for each shot sequence 132 included in the source video sequence 122. The source video sequence 122 includes, without limitation, any number of contiguous and non-overlapping shot sequences 132. Each of the shot sequences 132 includes a set of frames that usually have similar spatial-temporal properties and run for an uninterrupted period of time. The encoding optimization application 120 resides in one of the memories 116 and executes on one of the processors 112.

[0032] As shown, the encoding optimization application 120 includes, without limitation, a key frame optimizer 130, a ladder optimizer 140, and a content engine 150. Upon receiving the source video sequence 122, the key frame optimizer 130 determines one or more shot changes 172 included in the source video sequence 122. Each of the shot changes 172 specifies a boundary between a different pair of the shot sequences 132. The key frame optimizer 130 may determine the one or more shot changes 172 in any technically feasible fashion.

[0033] For instance, in some embodiments, the key frame optimizer 130 transmits the source video sequence 122 to a shot detector 170 that resides in the cloud 160. To determine the shot changes 172, the shot detector 170 executes any number of shot detection algorithms based on the source video sequence 122. Some examples of shot detection algorithms include, without limitation, a multi-scale sum-of-absolute-differences algorithm, a motion-compensated residual energy algorithm, a histogram of differences algorithm, a difference of histograms algorithm, and so forth. The shot detector 170 then transmits the shot changes 172 to the key frame optimizer 130. In alternate embodiments, the key frame optimizer 130 may perform any number of shot detection operations on the source video sequence 122 to determine the shot changes 172.

[0034] The key frame optimizer 130 performs partitioning operations on the source video sequence 122 based on the shot changes 172 to determine the shot sequences 132. In some embodiments, the key frame optimizer 130 also generates a key frame location list 134 based on the shot sequences 132. Each key frame location (not shown in Figure 1) included in the key frame location list 134 specifies a location of a

frame that is to be encoded as a key frame. A “key frame” and all subsequent frames from the same shot sequence included in the encoded video sequence 142 are decoded independently of any proceeding frames included in the encoded video sequence 142.

5 **[0035]** In operation, if the key frame optimizer 130 generates the key frame location list 134, then the ladder optimizer 140 ensures that the frames at the key frame locations included in the key frame location list 134 are encoded as key frames for all of the encoded videos sequences 142. The ladder optimizer 140 may ensure the frames at the key frame locations included in the key frame location list 134 are
10 encoded as key frames in any technically feasible fashion. For instance, in some embodiments, the ladder optimizer 140 configures an encoding application 180 to encode frames as key frames based on the key frame location list 132 when encoding video content. In other embodiments, the ladder optimizer 140 may perform any number of encoding operations to encode frames as key frames based on the key
15 frame location list 134 when encoding video content.

[0036] In this fashion, the key frame location list 134 ensures that each of the encoded video sequences 142 includes key frames that are aligned with the other encoded video sequences 142. As persons skilled in the art will recognize, during playback, the media title associated with the source video sequence 122 is switchable
20 between decoded versions of the encoded video sequences 142 at aligned key frames to optimize a viewing experience based on any number of relevant criteria. Examples of relevant criteria include the current connection bandwidth, the current connection latency, the content of the upcoming shot sequence, and the like.

[0037] As described in detail in conjunction with Figures 2 and 3, for each of the shot
25 sequences 132, the key frame optimizer 130 adds the location of the initial frame of the shot sequence 132 in the key frame location list 134. Further, for each of the shot sequences 132 that exceed a maximum group of picture (GOP) size 124, the key frame optimizer 130 adds the location(s) of one or more additional frames included in the shot sequence 132 to the frame location list 134. The key frame optimizer 130
30 may determine the maximum GOP size 124 in any technically feasible fashion. For example, the key frame optimizer 130 could receive the maximum GOP size 124 via a graphical user interface (GUI).

[0038] Upon receiving the shot sequences 132, the ladder optimizer 140 generates one or more encoded video sequences 142. For each of the shot sequences 132, a given encoded video sequence 142 includes, without limitation, an encoded shot sequence (not shown) that is associated with the shot sequence 132. Each of the encoded shot sequences includes encoded video content derived from the video content included in the associated shot sequence 132.

[0039] Each of the encoded video sequences 142 is associated with a different rung of an encoding ladder (not shown). Each of the rungs is associated with a different rung specification 144 and, consequently, each of the encoded video sequences 142 is associated with a different rung specification 144. Each of the rung specifications 144 includes, without limitation, any number of constraints. At least one of the constraints specifies a rung-specific threshold for a metric that varies based on at least one of the resolution and the encoding parameters. The ladder optimizer 140 ensures that each of the encoded shot sequences 142 included in the encoded video sequence 142 individually satisfies the associated rung specification 144.

[0040] To generate the encoded video sequence 142(x), for each of the shot sequences 132, the ladder optimizer 140 generates an encoded shot sequence that meets the rung specification 144(x). More precisely, for each of the shot sequences 132, the ladder optimizer 140 individually optimizes at least one of the resolution and the encoding parameter to ensure that the associated encoded shot sequence meets the rung specification 144(x). Subsequently, the ladder optimizer 140 includes each of the encoded shot sequence in the encoded video sequence 142(x).

[0041] In some “resolution optimization” embodiments, to generate each encoded shot sequence for the encoded video sequence 142(x), the ladder optimizer 140 selects a potentially different resolution from a resolution list. The ladder optimizer 140 selects the resolution based on the rung specification 144(x) and the video content included in the associated shot sequence 132. Consequently, the encoded video sequence 142(x) could include one encoded shot sequence that is associated with a first resolution, seven encoded shot sequences that are associated with a second resolution, twelve encoded shot sequences that are associated with a third resolution, etc. The rung specification 144(x) includes, without limitation, one or more encoding parameters and a constraint (also referred to herein as an “optimization constraint”) specifying a rung-specific threshold for a metric that varies based on the

selected resolution. The constraint may be specified in any technically feasible fashion based on any technically feasible metric. Further, the ladder optimizer 140 may select the resolution based on the constraint in any technically feasible fashion. Some resolution optimization embodiments are described in greater detail in

5 conjunction with Figures 4 and 5.

[0042] In some “encoding parameter optimization” embodiments, to generate each encoded shot sequence for the encoded video sequence 142(x), the ladder optimizer 140 selects a potentially different encoding parameter based on the rung specification 144(x) and the video content included the associated shot sequence 132.

10 Consequently, the encoded video sequence 142(x) could include seven encoded shot sequences that are associated with a first encoding parameter, five encoded shot sequences that are associated with a second encoding parameter, and twelve encoded shot sequences that are associated with a third encoding parameter, etc. The rung specification 144(x) includes, without limitation, a rung resolution and a
15 constraint specifying a rung-specific threshold for a metric that varies based on the selected encoding parameter. The constraint may be specified in any technically feasible fashion based on any technically feasible metric. Further, the ladder optimizer 140 may select the encoding parameter based on the constraint in any technically feasible fashion. Some encoding parameter optimization embodiments are
20 described in greater detail in conjunction with Figures 6 and 7.

[0043] As a general matter, the ladder optimizer 140 may generate encoded video content derived from video content based on a given resolution and given encoding parameters in any technically feasible fashion. For instance, in some embodiments, the ladder optimizer 140 performs sampling operations on the video content based on
25 the resolution to generate sampled video content. Subsequently, the ladder optimizer 140 configures the encoding application 180 to perform encoding operations on the sampled video content based on the encoding parameters to generate the encoded shot sequence.

[0044] As shown, the encoding application 180 resides in the cloud 160 and is
30 configured to efficiently perform encoding operations via one or more parallel encoders 182. Each of the parallel encoders 182 may include any number of computer instances 110. In alternate embodiments, the ladder optimizer 140 may perform encoding operations and the system 100 may omit the encoding application

180. In the same or other embodiments, the system 100 may include a sampling application, and the ladder optimizer 140 may configure the sampling application to perform sampling operations.

[0045] Notably, the ladder optimizer 140 may perform sampling operations and encoding operations at any level of granularity (e.g., per frame, per shot sequence, per video sequence, etc.) in any combination and in any technically feasible fashion. For instance, in some embodiments, the ladder optimizer 140 may perform sampling operations on the source video sequence 122 based on a first resolution to generate a sampled shot sequence. Subsequently, the ladder optimizer 140 may configure the encoding application 180 to encode the sampled video content corresponding to the shot sequence 132(x) based on first encoding parameters. The resulting encoded video content is an encoded shot sequence associated with the shot sequence 132(x), the first resolution, and the first encoding parameters.

[0046] The ladder optimizer 140 generates the encoded video sequence 142 associated with the rung based on the encoded shot sequences associated with the rung. The ladder optimizer 140 may generate the encoded video sequence 142 in any technically feasible fashion. For instance, in some embodiments, the ladder optimizer 140 adds each of the encoded shot sequences into the encoded video sequence 142. In other embodiments, the ladder optimizer 140 generates a monolithic encoded video sequence based on a first resolution and first encoding parameters. Subsequently, the ladder optimizer 140 replaces encoded video content included in the monolithic encoded video sequence with encoded shot sequence(s) generated based on different resolutions or encoding parameters.

[0047] The content engine 150 generates encoded files 192 based on the encoded video sequences 142 and a file granularity (not shown) that is one of video sequence or shot sequence. In alternate embodiments, the file granularity may specify any granularity in any technically feasible fashion. If the file granularity is shot sequence, then the content engine 150 analyzes all of the encoded video sequences 142 to identify the unique encoded shot sequences. For each unique encoded shot sequence, the content engine 150 generates a separate file and includes the file in the encoded files 182. In this fashion, the content engine 150 eliminates duplicate files. By contrast, if the file granularity is video sequence, then the content engine 150 generates a separate file for each of the encoded video sequences 142 and

includes the files in the encoded files 182. Finally, the content engine 150 transmits the encoded files 192 to a content delivery network (CDN) 190 included in the cloud 160 for on-demand delivery to endpoint devices. In alternate embodiments, the content engine 150 may generate the encoded files 192 and cause the encode files 192 to be delivered to endpoint devices in any technically feasible fashion.

[0048] In some embodiments, the content engine 150 configures the CDN 190 to deliver metadata to the endpoint devices along with the encoded files 192. An example of metadata includes, without limitation, quality scores associated with encoded video content at any level of granularity. For example, the metadata could include, without limitation, quality scores associated with one or more encoded shot sequences and/or quality scores associated with one or more of the encoded video sequences 142. The content engine 150 may configure CDN 190 to deliver the quality scores to the endpoint devices in any technically feasible fashion. For instance, in some embodiments, the content engine 150 may provide metadata to the CDN 190 along with each encoded file included in the encoded files 192. Subsequently, as part of streaming an encoded file to an endpoint device, the CDN 190 may embed the metadata associated with the encoded file in the stream. In other embodiments, the content engine 150 may generate a manifest file that specifies the metadata and then transmit the manifest file to the CDN 190. Subsequently, when initializing a streaming session, the CDN 190 may transmit the manifest file to the endpoint device.

[0049] The endpoint devices may perform any type and amount of operations based on the metadata. In particular, the endpoint devices may perform any number and type of adaptive streaming operations in any technically feasible fashion. For example, an endpoint device could compute an average VMAFh score based on VMAFh scores specified in a manifest. Subsequently, the endpoint device could analyze the VMAFh scores in conjunction with any other relevant criteria (e.g., bitrate, available bandwidth, type of media title, viewer preferences, etc) to select a particular encoded video sequence 142 or encoded shot sequence or to switch between encoded video sequences 142 or encoded shot sequences.

[0050] Note that the techniques described herein are illustrative rather than restrictive, and may be altered without departing from the broader spirit and scope of the invention. Many modifications and variations on the functionality provided by the

encoding optimization application 120, the key frame optimizer 130, the ladder optimizer 140, the content engine 150, the shot detector 170, the encoding application 180, and the content delivery network 190 will be apparent to those of ordinary skill in the art without departing from the scope and spirit of the described embodiments.

- 5 **[0051]** For instance, in some alternate embodiments, the key frame optimizer 130 may be configured to partition the source video sequence 122 into subsequences of frames that are not necessarily equivalent to the shot sequences 132. A subsequence of frames is also referred to herein as a "subsequence." A subsequence may represent a variety of different constructs, including a group of pictures (GOP), a sequence of one or more frames, a plurality of subsequences of frames, and so forth. In a complementary fashion, the key frame optimizer 130, the ladder optimizer 140, and the content engine 150 may be configured to operate on subsequences and encoded subsequences instead of the shot sequences 132 and encoded shot sequences. In general, the key frame optimizer 130 may be configured to identify subsequences in any technically feasible fashion. For example, the key frame optimizer 130 could be configured to identify and operate on subsequences for which a consistency metric lies within a specified range. The ladder optimizer 140 could then encode each subsequence at a different resolution or a different encoding parameter.
- 20 **[0052]** As persons skilled in the art will recognize, the techniques described herein may be modified to optimize audio encoding instead of video encoding. For example, in some embodiments, an audio track may be partitioned into "audio scenes." The audio scenes may be sampled via audio rendering hardware. The sampled audio scenes may be encoded via an audio codec that is configured via a quantization parameter and/or bitrate settings. The quality scores of the encoded audio scenes may be computed via a Perceptual Evaluation of Audio Quality (PEAQ) algorithm. Notably, the resolution and/or any number of encoding parameters may be optimized for each audio scene based on any of the techniques described herein in any combination.
- 30 **[0053]** It will be appreciated that the system 100 shown herein is illustrative and that variations and modifications are possible. For example the functionality provided by the encoding optimization application 120, the key frame optimizer 130, the ladder optimizer 140, the content engine 150, the shot detector 170, the encoding application

180, and the content delivery network 190, as described herein may be integrated into or distributed across any number of software applications (including one), hardware devices (e.g., a hardware-based encoder), and any number of components of the system 100. Further, the connection topology between the various units in Figure 1
5 may be modified as desired.

[0054] Figure 2 illustrates an example of a key frame location list 134 that is generated by the key frame optimizer 130 of Figure 1, according to various embodiments of the present invention. As described in conjunction with Figure 1, for each of the shot sequences 132, the key frame optimizer 130 adds a key frame
10 location 234 specifying the location of the initial frame of the shot sequence 132 to the key frame location list 134. In addition, for each of the shot sequences 132 that exceeds the maximum group of picture (GOP) size 124, the key frame optimizer 130 adds one or more additional key frame locations 234 to the key frame location list 134,

15 [0055] For each shot sequence 132 that exceeds the maximum GOP size 124, the key frame optimizer 130 adds the minimum number of additional key frames locations 234 to the key frame location list 234 to ensure that associated encoded shot sequences comply with the maximum GOP size 124. The size of the shot sequence 132 equals the number of frames included in the shot sequence 132. An encoded
20 video sequence 142 that complies with the maximum GOP size includes no more than the maximum GOP size 124 of frames between any two adjacent key frames. Further, the key frame optimizer 130 spaces the additional key frame locations 234 at approximately equal intervals across the shot sequence 132.

[0056] More precisely, the key frame optimizer 130 divides the total number of frames
25 included the shot sequence 132 by the maximum GOP size 124 to determine a quotient. The key frame optimizer 130 then performs a ceiling operation on the quotient to determine an optimized GOP size 224. The key frame optimizer 130 sets a current location equal to the location of the initial frame included in the shot sequence. The key frame optimizer 130 increments the current location by the
30 optimized GOP size 224 and adds the key frame location 234 specifying the new current location to the key frame list 134. In this fashion, the key frame optimizer 130 continues to add key frame locations 234 to the key frame list 134 at intervals of the optimized GOP size 224 for the remainder of the shot sequence 132.

[0057] Notably, the optimized GOP size 224 may vary based on the shot sequence 132. However, within each shot sequence 132 that exceeds the maximum GOP size 124, the key frame optimizer 130 spaces the additional key frame locations 234 in a uniform fashion. Spacing the additional key frame locations 243 in a uniform fashion may decrease the maximum number of frames during which switching between the encoded video sequences 142 is not feasible.

[0058] For explanatory purposes only, Figure 2 depicts an example source video sequence 122 that includes, without limitation, two shot changes 172(1)-172(2) and three shot sequences 132(1)-132(3). As shown, the size of the shot sequence 132(1) is smaller than the maximum GOP size 124. Accordingly, the key frame optimizer 130 adds the key frame location 234(1) specifying the location of the initial frame included in the shot sequence 132(1) to the key frame list 134.

[0059] The size of the shot sequence 132(2) exceeds the maximum GOP size 124. Accordingly, after adding the key frame location 234(2) specifying the location of the initial frame included in the shot sequence 132(2) to the key frame list 134, the key frame optimizer 130 computes the optimized GOP size 224 for the shot sequence 132(2). Based on the optimized GOP size 224, the key frame optimizer 130 adds the three key frame locations 234(3)-234(5) to the key frame list 134. Because the size of the shot sequence 132(3) is smaller than the maximum GOP size 124, the key frame optimizer adds a final key frame location 234(6) to the key frame list 134. The key frame location 234(6) specifies the location of the initial frame included in the shot sequence 132(3).

[0060] In alternate embodiments, the key frame optimizer 130 may determine any number of key frame locations 234 in any technically feasible fashion. Further, in some alternate embodiments, the key frame optimizer 130 partitions each of the shot sequences 132 that exceed the maximum GOP size 124 into multiple new shot sequences 132 based on the key frame locations 234. Accordingly, computing the optimized GOP size 224 results in an approximately even distribution of the frames across the new shot sequences 132.

Optimizing Key Frame Locations

[0061] Figure 3 is a flow diagram of method steps for positioning key frames within encoded video sequences, according to various embodiments of the present

invention. Although the method steps are described with reference to the systems of Figures 1-2, persons skilled in the art will understand that any system configured to implement the method steps, in any order, falls within the scope of the present invention.

5 **[0062]** As shown, a method 300 begins at step 302, where the key frame optimizer 130 determines the shot changes 172 included in the source video sequence 122. At step 304, the key frame optimizer 130 partitions the source video sequence 122 into the shot sequences 132 based on the shot changes 172. The key frame optimizer 130 then selects the first shot sequence 132.

10 **[0063]** At step 306, the key frame optimizer 130 adds a new key frame location 234 specifying the location of the initial frame included in the selected shot sequence 132 to the key frame location list 134. At step 308, the key frame optimizer 130 sets a current size equal to the size of the selected shot sequence 132. At step 310, the key frame optimizer 130 determines whether the current size exceeds the maximum GOP size 124. If, at step 310, the key frame optimizer 130 determines that the current size exceeds the maximum GOP size 124, then the method 300 proceeds to step 312.

15 **[0064]** At step 312, the key frame optimizer 130 computes the optimized GOP size 224 based on the current size and the maximum GOP size 124. At step 314, the key frame optimizer 130 adds one or more additional key frame locations 234 to the key frame location list 134. The additional key frame location(s) 234 specify location(s) at intervals of the optimized GOP size 224 within the selected shot sequence 132.

20 **[0065]** If, however, at step 310, the key frame optimizer 130 determines that the current size does not exceed the maximum GOP size 124, then the method 300 proceeds directly to step 316. At step 316, the key frame optimizer 130 determines whether the selected shot sequence 132 is the last shot sequence 132 included in the source video sequence 122. If, at step 316, the key frame optimizer 130 determines that the selected shot sequence 132 is not the last shot sequence 132 included in the source video sequence 122, then the method 300 proceeds to step 318. At step 318, the key frame optimizer 130 selects the next shot sequence 132 included in the source video sequence 122. The method 300 then returns to step 306, where the key frame optimizer 130 key frame optimizer 130 adds a new key frame location 234

specifying the location of the initial frame included in the selected shot sequence 132 to the key frame location list 134.

[0066] If, however, at step 316, the key frame optimizer 130 determines that the selected shot sequence 132 is the last shot sequence 132 included in the source video sequence 122, then the method 300 proceeds directly to step 320. At step 320, the ladder optimizer 140 generates the encoded video sequences 142. Each encoded video sequence 142 has key frames at the key frame locations 234 included in the key frame location list 134. At step 322, the content engine 150 generates the encoded files 192 based on the encoded video sequences 142. The content engine 150 then transmits the encoded video files 192 to the content delivery network 190 for streaming to endpoint devices. The method 300 then terminates.

Varying the Resolution When Generating Encoded Video Sequences

[0067] Figure 4 is a more detailed illustration of the ladder optimizer 140 of Figure 1, according to various embodiments of the present invention. As shown, the ladder optimizer 140 includes, without limitation, a tradeoff equation 410, a fixed quantization parameter (QP) 420, a resolution list 430, the rung specifications 144(1)-144(2), and the encoded video sequences 142(1)-144(2). As described in conjunction with Figure 1, each of the rung specifications 144 is associated with a different rung in the encoding ladder. The ladder optimizer 140 generates the encoded video sequences 142(1)-144(2) based, respectively, on the rung specifications 144(1)-144(2). In alternate embodiments, the ladder optimizer 140 may generate N encoded video sequences 142, where N is any integer greater than zero, based on N rung specifications 144 in any technically feasible fashion.

[0068] The ladder optimizer 140 determines a tradeoff score associated with increasing the resolution at which a given shot sequence 132 is encoded from a lower resolution to a higher resolution based on the tradeoff equation 410. The tradeoff equation 410 specifies the tradeoff score for the higher resolution as the ratio of an increase in bitrate to an increase in quality score when increasing the resolution at which a given shot sequence 132 is encoded. Consequently, a relatively low tradeoff score indicates a relatively high advantage associated with encoding the shot sequence 132 at the higher resolution instead of the lower resolution.

[0069] In general, the ladder optimizer 140 may compute the bitrate and quality scores for encoded video content in any technically feasible fashion. In some embodiments, to determine the quality score associated with an encoded shot sequence, the ladder optimizer 140 decodes the encoded shot sequence to generate a decoded shot sequence. The ladder optimizer 140 then re-samples (i.e., upsampled or downsampled) the decoded shot sequence to a target resolution to generate a re-constructed shot sequence that is relevant to the display characteristics of a class of endpoint devices. Subsequently, the ladder optimizer 140 analyzes the re-constructed shot sequence to generate the quality score for a visual quality metric.

[0070] The quality scores may be associated with any visual quality metric. Example of visual quality metrics include, without limitation, a peak signal-to-noise-ratio PSNR), a linear video multimethod assessment fusion (VMAF) metric, and a harmonic VMAF (VMAFh), to name a few. For instance, in some embodiments the ladder optimizer 140 implements a VMAF (or harmonic VMAF) algorithm to generate a VMAF score for each re-constructed shot sequence.

[0071] Although a multitude of video quality metrics, such as VMAF scores, can be calculated at different target resolutions, it should be clear that, when comparing quality scores among encoded shot sequences associated with different resolutions, applications (e.g., the ladder optimizer 140) need to use the same target resolution for re-sampling, after decoding. For instance, in some embodiments, the ladder optimizer 140 re-samples the decoded shot sequence to 1920x1080 to generate the re-constructed shot sequence for each encoded shot sequence. Subsequently the ladder optimizer 140 computes the quality scores associated with different encoded shot sequence based on the re-constructed shot sequences.

[0072] In some embodiments, to compute the quality score for a given encoded shot sequence, the ladder optimizer 140 first computes linear VMAF scores for each frame included in the encoded shot sequence. Techniques for computing linear VMAF scores are known in the art and omitted for clarity. The ladder optimizer 140 then computes a harmonic VMAF score for the encoded shot sequence based on Equation 1:

$$VMAF_{harmonic} = \frac{N}{\sum_{n=0}^{N-1} \frac{1}{1+VMAF_n}} - 1$$

Equation 1

Equation 1 assumes that the encoded shot sequence includes N frames, indexed between 0 and N-1 (inclusive). VMAF_n is the individual VMAF score for frame n, which has a value in the range [0,100]. The ladder optimizer 140 computes the final VMAF_h score for the encoded shot sequence to also have a value in the range [0,100].

[0073] The fixed QP 420 specifies a value for a quantization parameter that allows a monotonic performance in terms of bitrate and distortion when encoding a video content. The higher the QP value, the lower the resulting bitrate at the expense of lower quality. The fixed QP 420 may be specified in any technically feasible fashion. For instance, the fixed QP 420 may be specified via a GUI. In alternate embodiments, the ladder optimizer 140 may determine the fixed QP 420 in any technically feasible fashion. Further, in various embodiments, the ladder optimizer 140 may implement a different fixed QP 420 for each rung in the encoding ladder.

[0074] The resolution list 430 specifies the list of resolutions at which the ladder optimizer 140 may encode the shot sequences 132. Each of the rung specifications 144 includes, without limitation, a maximum tradeoff 420. The maximum tradeoff 420 is a constraint specifying a rung-specific threshold for the tradeoff metric. In general, to generate each of the encoded shot sequences for the encoded video sequence 144(x), the ladder optimizer 140 selects a potentially different resolution from the resolution list 430 based on the rung specification 144(x) and the video content included the associated shot sequence 132. More specifically, for each shot sequence 132, the ladder optimizer 140 identifies the maximum resolution included the resolution list 430 for which the associated encoded shot sequence complies with the maximum tradeoff 420(x). Note that, by default, the ladder optimizer 140 determines that the lowest resolution included in the resolution list 430 complies with all of the maximum tradeoffs 420. The ladder optimizer 140 may identify the maximum resolutions in any technically feasible fashion.

[0075] For example, to encode the shot sequence 132(3) for the encoded video sequence 144(1), the ladder optimizer 140 encodes the shot sequence 132(3) at

960x540 (the lowest resolution included in the resolution list 430) and the fixed QP 420 of 26 to generate a 960x540 encode. The ladder optimizer 140 then encodes the shot sequence 132(3) at 1280x720 and the fixed QP 420 of 26 to generate a 1280x720 encode. The ladder optimizer 240 computes the bitrate and VMAFh score for the 960x540 encode as well as the bitrate and VMAFh score the 1280x720 encode. Subsequently, the ladder optimizer 140 computes the tradeoff score for increasing the resolution from 960x540 to 1280x720 based on the tradeoff equation 410, the bitrates, and the VMAFh scores. The ladder optimizer 140 determines that the tradeoff score for increasing the resolution from 960x540 to 1280x720 is less than the maximum tradeoff 420(1) of 0.1. Because the ladder optimizer 140 has not yet determined whether 1280x720 is the highest resolution that complies with the maximum tradeoff 420(1), the ladder optimizer 140 sets the current encoding resolution equal to 1280x720.

[0076] The ladder optimizer 140 then encodes the shot sequence 123(3) at 1920x1080 and the fixed QP 420 of 26 to generate a 1920x1080 encode. The ladder optimizer 140 determines that the tradeoff score for increasing the resolution from 1280x720 to 1920x1080 exceeds the maximum tradeoff 420(1) of 0.1. Consequently, the ladder optimizer 140 selects the resolution of 1280x720 as the resolution for the encoded shot sequence derived from the shot sequence 132(3). Accordingly, the ladder optimizer 140 includes the 1280x720 encode in the encoded video sequence 142(1). The 1280x720 encode is depicted in the encoded video sequence 142(1) as a box with horizontal lines.

[0077] In this fashion, the ladder optimizer 140 generates the encoded video sequence 142(1) that includes, without limitation, six encoded shot sequences associated with 960x540 and one shot sequence associated with 1280x720. Since the rung specification 420(2) specifies the maximum tradeoff 420(2) of 0.15, the ladder optimizer 140 generates the encoded video sequence 142(2) that differs from the encoded video sequence 142(1). The encoded video sequence 142(2) includes, without limitation, four encoded shot sequences associated with 960x540, one encoded shot sequence associated with 1280x720, and two encoded shot sequences associated with 1920x1080.

[0078] In alternate embodiments, the ladder optimizer 140 may compute the tradeoff score in any technically feasible fashion. For instance, in some alternate

embodiments, the ladder optimizer 140 performs frequency spectrum analysis operations to determine the tradeoff score. For each shot sequence 132 included in the source video sequence, the ladder optimizer 140 computes a two-dimensional frequency spectrum for a sample shot included in the shot sequence. The ladder optimizer 140 may compute the frequency spectrum in any technically feasible fashion. For instance, in some embodiments, the ladder optimizer 140 generates a fast Fourier transform (FFT) of the sample shot. The ladder optimizer 140 then performs comparison operations between the amounts of energy included in the higher frequencies to the amount of energy included in the center frequency to determine the tradeoff scores for different resolutions included in the resolution list 430.

[0079] Note that the techniques described herein are illustrative rather than restrictive, and may be altered without departing from the broader spirit and scope of the invention. Many modifications and variations on the functionality provided by the ladder optimizer 140 will be apparent to those of ordinary skill in the art without departing from the scope and spirit of the described embodiments. For instance, in some embodiments, the fixed QP 410 is replaced by any number of fixed encoding parameters.

[0080] Figure 5 is a flow diagram of method steps for encoding a source video sequence, according to various embodiments of the present invention. Although the method steps are described with reference to the systems of Figures 1 and 4, persons skilled in the art will understand that any system configured to implement the method steps, in any order, falls within the scope of the present invention.

[0081] As shown, a method 500 begins at step 502, where the ladder optimizer 140 receives the shot sequences 132 included in the source video sequence 122. At step 504, the ladder optimizer 140 selects the first rung included in the encoding ladder. At step 506, the ladder optimizer 140 selects the maximum tradeoff 420 associated with the selected rung and selects the first shot sequence 132 included in the source video sequence 122.

[0082] At step 508, the ladder optimizer 140 identifies the maximum resolution included in the resolution list 320 for which the encoded shot sequence derived from the selected shot sequence 132 complies with the selected maximum tradeoff 420.

Notably, the encoded shot sequence is encoded at fixed encoding parameter(s) irrespective of the resolution. The ladder optimizer 140 may identify the maximum compliant resolution in any technically feasible fashion. For instance, in some embodiments, the ladder optimizer 140 may determine the maximum resolution based on the tradeoff equation 410. In other embodiments, the ladder optimizer 140 may perform frequency spectrum analysis operations on a sample frame included in the selected shot sequence 132 to determine the maximum resolution.

[0083] At step 510, the ladder optimizer 140 acquires the encoded shot sequence associated with the selected shot sequence 132, the selected resolution, and the fixed encoding parameter(s). The ladder optimizer 140 may acquire the encoded shot sequence in any technically feasible fashion. For instance, if the ladder optimizer 140 selects the resolution based on the tradeoff equation 410, then the ladder optimizer 140 re-uses the encoded shot sequence generated as part of step 508. By contrast, if the ladder optimizer 140 performs frequency spectrum analysis operations to select the resolution, then the ladder optimizer 140 samples the selected shot sequence 120 at the selected resolution to generate sampled video content. The ladder optimizer 140 then configures the encoding application 180 to encode the sampled video content at the fixed encoding parameter(s).

[0084] At step 512, the ladder optimizer 140 incorporates the encoded shot sequence into the encoded video sequence 142 associated with the selected rung. At step 514, the ladder optimizer 140 determines whether the selected shot sequence 132 is the last shot sequence 132 included in the source video sequence 122. If, at step 514, the ladder optimizer 140 determines that the selected shot sequence 132 is not the last shot sequence 132 included in the source video sequence 122, then the method 500 proceeds to step 516. At step 516, the ladder optimizer 140 selects the next shot sequence 132 included in the source video sequence 122. The method 500 then returns to step 508, where the ladder optimizer 140 selects the resolution for the new selected shot sequence 132.

[0085] If, however, at step 516, the ladder optimizer 140 determines that the selected shot sequence 132 is the last shot sequence 132 included in the source video sequence 122, then the method 500 proceeds directly to step 518. At step 518, the ladder optimizer 140 determines whether the selected rung is the last rung included in the encoding ladder. If, at step 518, the ladder optimizer 140 determines that the

selected rung is not the last rung included in the encoding ladder, then the method 500 proceeds to step 520. At step 520, the ladder optimizer 140 selects the next rung included in the encoding ladder. The method 500 then returns to step 506, where the source video sequence selects the first shot sequence 132 included in the source video sequence 122 and the maximum tradeoff 420 associated with the selected rung.

[0086] If, however, at step 518, the ladder optimizer 140 determines the selected rung is the last rung included in the encoding ladder, then the method 500 proceeds directly to step 522. At step 522, the content engine 150 generates the encoded files 192 based on the encoded video sequences 142. The content engine 150 transmits the encoded video files 192 to the content delivery network 190 for streaming to endpoint devices. The method 500 then terminates.

Varying an Encoding Parameter When Generating Encoded Video Sequences

[0087] Figure 6 is a more detailed illustration of the ladder optimizer 140 of Figure 1 configured to vary a quantization parameter (QP) across the source video sequence 122, according to various embodiments of the present invention. As shown, the ladder optimizer 140 includes, without limitation, the rung specifications 144(1)-144(2) and the encoded video sequences 142(1)-142(2).

[0088] As described in conjunction with Figure 1, each of the rung specifications 144 is associated with a different rung in the encoding ladder. The ladder optimizer 140 generates the encoded video sequences 142(1)-144(2) based, respectively, on the rung specifications 144(1)-144(2). In alternate embodiments, the ladder optimizer 140 may generate N encoded video sequences 142, where N is any integer greater than zero, based on N rung specifications 144 in any technically feasible fashion.

[0089] Each of the rung specifications 144 includes, without limitation, a rung resolution 620 and a minimum quality score 630. As shown, the rung specification 144(1) specifies a rung resolution 620(1) of 960x540 and a minimum quality score 630(1) of 80. The rung specification 144(2) specifies a rung resolution 620(2) of 1280x720 and a minimum quality score 630(2) of 92. The quality scores may be associated with any visual quality metric. Example of visual quality metrics include, without limitation, a peak signal-to-noise-ratio (PSNR), a linear video multimethod assessment fusion ((VMAF) metric, and a harmonic VMAF (VMAFh), to name a few.

The ladder optimizer 140 may compute the quality scores for encoded video content in any technically feasible fashion. One technique for generating VMAFh scores is described in conjunction with Figure 4.

[0090] In general, to generate each encoded shot sequence for the encoded video sequence 142(x), the ladder optimizer 140 selects a potentially different encoding parameter based on the rung specification 144(x) and the video content included the associated shot sequence 132. For explanatory purposes only, exemplary QP selection operations 610 illustrate operations that the ladder optimizer 140 performs to select the QP for the encoded shot sequence derived from the shot sequence 132(1) and included in the encoded video sequence 142(1).

[0091] First, the ladder optimizer 140 samples the source video content at the rung resolution 620(1) of 920x540 to generate sampled shot sequences for each of the shot sequences 132. The ladder optimizer 140 then selects the sampled shot sequence associated with the shot sequence 132(1). The ladder optimizer 140 sets a current QP to an initial maximum QP allowed by the video codec used. For example, in the case of H.264/AVC, the ladder optimizer 140 selects a current QP of 51, and encodes the selected shot sequence at the current QP of 51 to generate a first encoded shot sequence. In alternate embodiments, the ladder optimizer 140 may implement any initial QP in any technically feasible fashion. For instance, in some embodiments the ladder optimizer 140 may implement an initial QP of the middle QP allowed by the video codec used. For example, in the case of H.264/AVC, the middle QP allowed is 26.

[0092] The ladder optimizer 140 computes the VMAFh score of 74 for the first encoded shot sequence. Because the VMAFh score of 74 is smaller than the minimum quality score 630(1) of 80, the ladder optimizer 140 determines that the first encoded shot sequence does not satisfy the rung specification 144(1). The ladder optimizer 140 then sets the current QP to a new QP. More specifically, the ladder optimizer 140 decrements the current QP of 51 to generate the current QP of 50. In alternate embodiments, the ladder optimizer 140 may determine a new current QP in any technically feasible fashion.

[0093] For instance, in some embodiments, the ladder optimizer 140 subtracts 2 from the current QP to generate a new current QP. In other embodiments, the ladder

optimizer 140 sets the new current QP based on a binary search algorithm. In such embodiments, instead of setting the new current QP equal to 50, the ladder optimizer 140 may set the new current QP equal to 26.

[0094] The ladder optimizer 140 re-encodes the selected sampled shot sequence at the current QP of 50 to generate a second encoded shot sequence. Subsequently, the ladder optimizer 140 computes the VMAFh score of 79 for the second encoded shot sequence. Because the VMAFh score of 70 is less than the minimum quality score 630(1) of 80, the ladder optimizer 140 determines that the second encoded shot sequence does not satisfy the rung specification 144(1). The ladder optimizer 140 then sets the current QP to a new QP. More specifically, the ladder optimizer 140 decrements the current QP of 50 to generate the new QP of 49.

[0095] The ladder optimizer 140 re-encodes the selected sampled shot sequence at the current QP of 49 to generate a third encoded shot sequence. Subsequently, the ladder optimizer 140 computes the VMAFh score of 82 for the third encoded shot sequence. Because the VMAFh score of 82 is greater than the minimum quality score 630(1) of 80, the ladder optimizer 140 determines that the third encoded shot sequence satisfies the rung specification 144(1). Consequently, the ladder optimizer 140 adds the third encoded shot sequence to the encoded video sequence 142(1) as the encoded video sequence corresponding to the shot sequence 132(1).

[0096] In this fashion, the ladder optimizer 140 generates the encoded video sequence 142(1) that includes, without limitation, seven encoded shot sequences associated with 960x540 and QPs ranging from 6 to 49. Further, the ladder optimizer 140 generates the encoded video sequence 142(2) that includes, without limitation, seven encoded sequences associated with 1280x720 and QPs ranging from 10 to 51.

[0097] Note that the techniques described herein are illustrative rather than restrictive, and may be altered without departing from the broader spirit and scope of the invention. Many modifications and variations on the functionality provided by the ladder optimizer 140 will be apparent to those of ordinary skill in the art without departing from the scope and spirit of the described embodiments. For instance, in alternate embodiments the ladder optimizer 140 may vary any number of encoding parameters based on any number and type of constraints included in the rung specifications 144.

[0098] Figure 7 is a flow diagram of method steps for varying an encoding parameter when encoding a source video sequence, according to various embodiments of the present invention. Although the method steps are described with reference to the systems of Figures 1, and 6, persons skilled in the art will understand that any system configured to implement the method steps, in any order, falls within the scope of the present invention.

[0099] As shown, a method 700 begins at step 702, where the ladder optimizer 140 selects the first rung included in the encoding ladder and the associated minimum quality score 630. At step 704, the ladder optimizer 140 samples the source video sequence 122 at the rung resolution 620 associated with the selected rung and selects the first shot sequence 132 included in the source video sequence 122. At step 706, the ladder optimizer 140 selects the sampled shot sequence associated with the selected shot sequence 132 and initializes a current encoding parameter

[0100] At step 708, the ladder optimizer 140 encodes the selected sampled shot sequence at the current encoding parameter to generate a current encoded shot sequence. At step 710, the ladder optimizer 140 computes a quality score for the current encoded shot sequence. At step 712, the ladder optimizer 140 determines whether the quality score is less than the selected minimum quality score 620. If, at step 712, the ladder optimizer 140 determines that the quality score is less than the selected minimum quality score 620, then the method 700 proceeds to step 714. At step 714, the ladder optimizer 140 selects a new current encoding parameter. The method 700 then returns to the step 708, where the ladder optimizer 140 encodes the selected sampled shot sequence at the new current encoding parameter.

[0101] If, however, at step 712, the ladder optimizer 140 determines that the quality score is greater than or equal to the minimum quality score 620, then the method 700 proceeds directly to step 716. At step 716, the ladder optimizer 140 incorporates the current encoded shot sequence into the encoded video sequence 142 associated with the selected rung. At step 718, the ladder optimizer 140 determines whether the current shot sequence 132 is the last shot sequence 132 included in the source video sequence 122.

[0102] If, at step 718, the ladder optimizer 140 determines that the selected shot sequence 132 is not the last shot sequence 132 included in the source video

sequence 122, then the method 700 proceeds to step 720. At step 720, the ladder optimizer 140 selects the next shot sequence 132 included in the source video sequence 122. The method 700 then returns to step 706, where the ladder optimizer 140 selects the sampled shot sequence associated with the selected shot sequence 132 and initializes a current encoding parameter.

[0103] If, however, at step 718, the ladder optimizer 140 determines that the selected shot sequence 132 is the last shot sequence 132 included in the source video sequence 122, then the method 700 proceeds directly to step 722. At step 722, the ladder optimizer 140 determines whether the selected rung is the last rung included in the encoding ladder. If, at step 722, the ladder optimizer 140 determines that the selected rung is not the last rung included in the encoding ladder, then the method 700 proceeds to step 724. At step 724, the ladder optimizer 140 selects the next rung included in the encoding ladder and the associated minimum quality score 630. The method 700 then returns to step 704, where the ladder optimizer 140 samples the source video sequence 122 at the rung resolution 620 associated with the selected rung.

[0104] If however, at step 722, the ladder optimizer 140 determines that the selected rung is the last rung included in the encoding ladder, then the method 700 proceeds directly to step 726. At step 726, the content engine 170 generates the encoded files 192 based on the encoded video sequences 142. The content engine 170 then transmits the encoded video files 192 to the content delivery network 190 for streaming to endpoint devices. The method 700 then terminates.

[0105] In sum, the disclosed techniques enable optimization of a viewing experience when playing back a source video sequence via streaming. An encoding optimization application includes, without limitation, a key frame optimizer, a ladder optimizer, and a content engine. First, a key frame optimizer partitions a source video sequence into multiple shot sequences based on shot changes. Each shot change separates a different pair of shot sequences. Each shot sequence includes a set of frames that usually have similar spatial-temporal properties and run for an uninterrupted period of time. The key frame optimizer may determine the shot changes in any technically feasible fashion.

[0106] In some embodiments, the key frame optimizer also determines optimized locations for key frames based on the shot changes. Each key frame is encoded interdependently of preceding frames. For each shot sequence, the key frame optimizer adds the location of the initial frame of the shot sequence to a key frame location list. Further, for each shot sequence having a size greater than a maximum group of pictures (GOP) size, the key frame optimizer determines an optimal GOP size. To determine the optimal GOP size for a given shot sequence, the key frame optimizer divides the number of frames included in the shot sequence by the optimal GOP size to generate a quotient. The key frame optimizer then performs a ceiling operation on the quotient. The key frame optimizer then adds additional locations at intervals of the optimal GOP size within the shot sequence to the key frame location list. Subsequently, the ladder optimizer configures an encoding application to encode frames at the key frame locations as key frames when performing encoding operations on video content associated with the source video sequence.

[0107] After receiving the shot sequences from the key frame optimizer, the ladder optimizer generates one or more encoded video sequences based on a combination of resolution and encoding parameters that the ladder optimizer may vary for each shot sequence. Notably, each encoded video sequence is associated with a different rung included in an encoding ladder. Each rung is also associated with a rung specification that includes at least one constraint. The ladder optimizer ensures that each encoded shot sequence included in a given encoded video sequence individually meets the associated rung specification.

[0108] In some encoding parameter optimization embodiments, to generate each encoded shot sequence for a given encoded video sequence, the ladder optimizer selects a potentially different resolution based on the associated rung specification and the video content included the associated shot sequence. Consequently, the resulting encoded video sequence could include one encoded shot sequence that is associated with a first resolution, seven encoded shot sequences that are associated with a second resolution, twelve encoded shot sequences that are associated with a third resolution, etc. The rung specification specifies one or more encoding parameters and a constraint specifying a threshold for a metric that varies based on the selected resolution. The constraint may be specified in any technically feasible

fashion based on any technically feasible metric. Further, the ladder optimizer may select the resolution based on the constraint in any technically feasible fashion.

[0109] In some embodiments, to generate each encoded shot sequence for a given encoded video sequence, the ladder optimizer selects a potentially different encoding parameter based on the associated rung specification and the video content included the associated shot sequence. Consequently, the encoded video sequence could include seven encoded shot sequences that are associated with a first encoding parameter, five encoded shot sequences that are associated with a second encoding parameter, and twelve encoded shot sequences that are associated with a third encoding parameter, etc. The associated rung specification specifies a resolution and a constraint specifying a threshold for a metric that varies based on the selected encoding parameter. The constraint may be specified in any technically feasible fashion based on any technically feasible metric. Further, the ladder optimizer may select the encoding parameter based on the constraint in any technically feasible fashion.

[0110] For each rung, the ladder optimizes aggregates the different encoded shot sequences associated with the rung to generate the encoded video sequence. Finally, the content engine generates encoded files based on the encoded video sequences and a file granularity (e.g., video sequence or shot sequence) and stores the encoded files in a content distribution network for distribution to endpoint devices. If the content engine generates the encoded files at the file granularity of shot sequences, then the content engine eliminates duplicates and generates a single copy for each encoded shot sequence associated with the same resolution and encoding parameters.

[0111] At least one technological improvement relative to prior art is that varying the resolution and/or encoding parameters when encoding individual shot sequences included in an encoded video sequence reduces encoding inefficiencies associated with conventional encoding techniques. Such encoding inefficiencies not only needlessly waste computational and storage resources, these types of inefficiencies also unnecessarily increase the bandwidth required to stream encoded versions of media titles to endpoint devices. Another technological improvement relative to the prior art is that the encoding optimizer application positions key frames within encoded video sequences based on changes in the spatial-temporal properties of the

source video sequence. Consequently, the disclosed techniques reduce encoding inefficiencies attributable to encoding frames that immediately follow relatively similar frames as key frames. Yet another technological improvement relative to prior art is that storing each unique encoded individual shot sequences as an individual file may
5 reduce the total resources required to store the encoded video sequences.

[0112] 1. In some embodiments, a computer-implemented method comprises partitioning a source video sequence associated with a media title into a plurality of subsequences that includes a first subsequence and a second subsequence; generating a first encoded subsequence based on the first subsequence and a first
10 resolution; generating a second encoded subsequence based on the second subsequence and a second resolution that is higher than the first resolution; and aggregating the first encoded subsequence and the second encoded subsequence to generate a first encoded video sequence, wherein at least a portion of the first encoded video sequence is subsequently streamed to an endpoint device during a
15 playback of the media title.

[0113] 2. The computer-implemented method of clause 1, wherein each of the subsequences included in the plurality of subsequences comprises a shot sequence.

[0114] 3. The computer-implemented method of clauses 1 or 2, further comprising computing the first resolution based on a two-dimensional frequency spectrum of a
20 first frame included in the first subsequence.

[0115] 4. The computer-implemented method of any of clauses 1-3, further comprising, prior to aggregating the first encoded subsequence and the second encoded subsequence generating a third encoded subsequence based on the first subsequence and a third resolution that is lower than the first resolution; computing a
25 delta bitrate based on a first bitrate associated with the first encoded subsequence and a second bitrate associated with the third encoded subsequence; computing a delta visual quality score based on a first visual quality score associated with the first encoded subsequence and a second visual quality score associated with the third encoded subsequence, wherein both the first visual quality score and the second
30 visual quality score are associated with a target resolution; and determining that the first encoded subsequence, but not the third encoded subsequence, complies with a optimization constraint based on the delta bitrate and the delta visual quality score.

[0116] 5. The computer-implemented method of any of clauses 1-4, further comprising, prior to aggregating the first encoded subsequence and the second encoded subsequence generating a third encoded subsequence based on the first subsequence and the second resolution; computing a delta bitrate based on a first
5 bitrate associated with the first encoded subsequence and a second bitrate associated with the second encoded subsequence; computing a delta visual quality score based on a first visual quality score associated with the first encoded subsequence and a second visual quality score associated with the second encoded subsequence, wherein both the first visual quality score and the second visual quality
10 score are associated with a target resolution; determining that the third encoded subsequence does not comply with an optimization constraint based on the delta bitrate and the delta visual quality score; and discarding the third encoded subsequence.

[0117] 6. The computer-implemented method of any of clauses 1-5, wherein the first
15 visual quality score comprises a video multimethod assessment fusion score, a harmonic video multimethod assessment fusion score, or a peak-signal-to-noise ratio.

[0118] 7. The computer-implemented method of any of clauses 1-6, further comprising generating a third encoded subsequence based on the first subsequence and the second resolution; aggregating the third encoded subsequence and the
20 second encoded subsequence to generate a second encoded video sequence that is associated with a higher bitrate than the first encoded video sequence, wherein at least a portion of the second encoded video sequence, instead of the at least a portion of the first encoded video sequence, is subsequently streamed to the endpoint device during the playback of the media title.

[0119] 8. The computer-implemented method of any of clauses 1-7, wherein the at
25 least the portion of the first encoded video sequence comprises both the first encoded subsequence and the second encoded subsequence, and the first encoded video sequence is streamed to the endpoint device via a first file and the second encoded video sequence is streamed to the endpoint device via a second file.

[0120] 9. The computer-implemented method of any of clauses 1-8, further
30 comprising computing a first visual quality score based on the first encoded video sequence and a target resolution; specifying the first visual quality score via a

manifest file associated with the media title; and transmitting the manifest file to the endpoint device.

[0121] 10. In some embodiments, a computer-readable storage medium includes instructions that, when executed by a processor, cause the processor to perform the steps of generating a first encoded shot sequence based on a first shot sequence included in a source video sequence and a first resolution, wherein the source video sequence is associated with a media title; generating a second encoded shot sequence based on a second shot sequence included in the source video sequence and a second resolution that is higher than the first resolution; and generating a first encoded video sequence based on the first encoded shot sequence and the second encoded shot sequence, wherein at least a portion of the first encoded video sequence is subsequently streamed to an endpoint device during a playback of the media title.

[0122] 11. The computer-readable storage medium of clause 10, further comprising computing the first resolution based on a two-dimensional frequency spectrum of a first frame included in the first shot sequence.

[0123] 12. The computer-readable storage medium of clauses 10 or 11, further comprising computing the two-dimensional frequency spectrum based on a fast Fourier transform algorithm and the first frame.

[0124] 13. The computer-readable storage medium of any of clauses 10-12, further comprising, prior to generating the first encoded video sequence generating a third encoded shot sequence based on the first shot sequence and a third resolution that is lower than the first resolution; computing a delta bitrate based on a first bitrate associated with the first encoded shot sequence and a second bitrate associated with the third encoded shot sequence; computing a delta visual quality score based on a first visual quality score associated with the first encoded shot sequence and a second visual quality score associated with the third encoded shot sequence, wherein both the first visual quality score and the second visual quality score are associated with a target resolution; and determining that the first encoded shot sequence, but not the third encoded shot sequence, complies with a optimization constraint based on the delta bitrate and the delta visual quality score.

[0125] 14. The computer-readable storage medium of any of clauses 10-13, further comprising, prior to prior to generating the first encoded video sequence generating a third encoded shot sequence based on the first shot sequence and the second resolution; computing a delta bitrate based on a first bitrate associated with the first encoded shot sequence and a second bitrate associated with the second encoded shot sequence; computing a delta visual quality score based on a first visual quality score associated with the first encoded shot sequence and a second visual quality score associated with the second encoded shot sequence, wherein both the first visual quality score and the second visual quality score are associated with a target resolution; determining that the third encoded shot sequence does not comply with an optimization constraint based on the delta bitrate and the delta visual quality score; and discarding the third encoded shot sequence.

[0126] 15. The computer-readable storage medium of any of clauses 10-14, further comprising generating a third encoded shot sequence based on the first shot sequence and the second resolution; and generating a second encoded video sequence that is associated with a higher bitrate than the first encoded video sequence based on the third encoded shot sequence and the second encoded shot sequence, wherein at least a portion of the second encoded video sequence, instead of the at least a portion of the first encoded video sequence, is subsequently streamed to the endpoint device during the playback of the media title.

[0127] 16. The computer-readable storage medium of any of clauses 10-15, wherein the at least the portion of the first encoded video sequence is streamed to the endpoint device via a first file that includes the first encoded video sequence, and the at least the portion of the second encoded video sequence is streamed to the endpoint device via a second file that includes the second encoded video sequence.

[0128] 17. The computer-readable storage medium of any of clauses 10-16, further comprising computing a first visual quality score based on the first encoded video sequence and a target resolution; specifying the first visual quality score via a manifest file associated with the media title; and transmitting the manifest file to the endpoint device.

[0129] 18. The computer-readable storage medium of any of clauses 10-17, wherein the first visual quality score comprises a video multimethod assessment fusion score,

a harmonic video multimethod assessment fusion score, or a peak-signal-to-noise ratio.

[0130] 19. In some embodiments a system comprises a memory storing instructions; and a processor that is coupled to the memory and, when executing the instructions, is configured to generate a first encoded shot sequence based on a first shot sequence included in a source video sequence and a first resolution; generate a second encoded shot sequence based on a second shot sequence included in the source video sequence and a second resolution that is higher than the first resolution; and aggregate the first encoded shot sequence and the second encoded shot sequence to generate a first encoded video sequence, wherein at least a portion of the first encoded video sequence is subsequently streamed to an endpoint device during a playback of a media title that is associated with the source video sequence.

[0131] 20. The system of clause 19, wherein the processor is further configured to generate a third encoded shot sequence based on the first shot sequence and the second resolution; and aggregate the third encoded shot sequence and the second encoded shot sequence to generate a second encoded video sequence that is associated with a higher bitrate than the first encoded video sequence, wherein at least a portion of the second encoded video sequence, instead of the at least a portion of the first encoded video sequence, is subsequently streamed to the endpoint device during the playback of the media title.

[0132] Any and all combinations of any of the claim elements recited in any of the claims and/or any elements described in this application, in any fashion, fall within the contemplated scope of the present invention and protection.

[0133] The descriptions of the various embodiments have been presented for purposes of illustration, but are not intended to be exhaustive or limited to the embodiments disclosed. Many modifications and variations will be apparent to those of ordinary skill in the art without departing from the scope and spirit of the described embodiments.

[0134] Aspects of the present embodiments may be embodied as a system, method or computer program product. Accordingly, aspects of the present disclosure may take the form of an entirely hardware embodiment, an entirely software embodiment

(including firmware, resident software, micro-code, etc.) or an embodiment combining software and hardware aspects that may all generally be referred to herein as a “module” or “system.” Furthermore, aspects of the present disclosure may take the form of a computer program product embodied in one or more computer readable medium(s) having computer readable program code embodied thereon.

[0135] Any combination of one or more computer readable medium(s) may be utilized. The computer readable medium may be a computer readable signal medium or a computer readable storage medium. A computer readable storage medium may be, for example, but not limited to, an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system, apparatus, or device, or any suitable combination of the foregoing. More specific examples (a non-exhaustive list) of the computer readable storage medium would include the following: an electrical connection having one or more wires, a portable computer diskette, a hard disk, a random access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM or Flash memory), an optical fiber, a portable compact disc read-only memory (CD-ROM), an optical storage device, a magnetic storage device, or any suitable combination of the foregoing. In the context of this document, a computer readable storage medium may be any tangible medium that can contain, or store a program for use by or in connection with an instruction execution system, apparatus, or device.

[0136] Aspects of the present disclosure are described above with reference to flowchart illustrations and/or block diagrams of methods, apparatus (systems) and computer program products according to embodiments of the disclosure. It will be understood that each block of the flowchart illustrations and/or block diagrams, and combinations of blocks in the flowchart illustrations and/or block diagrams, can be implemented by computer program instructions. These computer program instructions may be provided to a processor of a general purpose computer, special purpose computer, or other programmable data processing apparatus to produce a machine. The instructions, when executed via the processor of the computer or other programmable data processing apparatus, enable the implementation of the functions/acts specified in the flowchart and/or block diagram block or blocks. Such processors may be, without limitation, general-purpose processors, special-purpose processors, application-specific processors, or field-programmable gate arrays.

[0137] The flowchart and block diagrams in the figures illustrate the architecture, functionality, and operation of possible implementations of systems, methods and computer program products according to various embodiments of the present disclosure. In this regard, each block in the flowchart or block diagrams may represent a module, segment, or portion of code, which comprises one or more executable instructions for implementing the specified logical function(s). It should also be noted that, in some alternative implementations, the functions noted in the block may occur out of the order noted in the figures. For example, two blocks shown in succession may, in fact, be executed substantially concurrently, or the blocks may sometimes be executed in the reverse order, depending upon the functionality involved. It will also be noted that each block of the block diagrams and/or flowchart illustration, and combinations of blocks in the block diagrams and/or flowchart illustration, can be implemented by special purpose hardware-based systems that perform the specified functions or acts, or combinations of special purpose hardware and computer instructions.

[0138] While the preceding is directed to embodiments of the present disclosure, other and further embodiments of the disclosure may be devised without departing from the basic scope thereof, and the scope thereof is determined by the claims that follow.

WHAT IS CLAIMED IS:

1. A computer-implemented method, comprising:
partitioning a source video sequence associated with a media title into a
plurality of subsequences that includes a first subsequence and a
5 second subsequence;
generating a first encoded subsequence based on the first subsequence and a
first resolution;
generating a second encoded subsequence based on the second
subsequence and a second resolution that is higher than the first
10 resolution; and
aggregating the first encoded subsequence and the second encoded
subsequence to generate a first encoded video sequence,
wherein at least a portion of the first encoded video sequence is subsequently
streamed to an endpoint device during a playback of the media title.
15
2. The computer-implemented method of claim 1, wherein each of the
subsequences included in the plurality of subsequences comprises a shot sequence.
3. The computer-implemented method of claim 1, further comprising computing
20 the first resolution based on a two-dimensional frequency spectrum of a first frame
included in the first subsequence.
4. The computer-implemented method of claim 1, further comprising, prior to
aggregating the first encoded subsequence and the second encoded subsequence:
25 generating a third encoded subsequence based on the first subsequence and
a third resolution that is lower than the first resolution;
computing a delta bitrate based on a first bitrate associated with the first
encoded subsequence and a second bitrate associated with the third
encoded subsequence;
30 computing a delta visual quality score based on a first visual quality score
associated with the first encoded subsequence and a second visual
quality score associated with the third encoded subsequence, wherein
both the first visual quality score and the second visual quality score are
associated with a target resolution; and

determining that the first encoded subsequence, but not the third encoded subsequence, complies with a optimization constraint based on the delta bitrate and the delta visual quality score.

- 5 5. The computer-implemented method of claim 1, further comprising, prior to aggregating the first encoded subsequence and the second encoded subsequence: generating a third encoded subsequence based on the first subsequence and the second resolution;
- 10 computing a delta bitrate based on a first bitrate associated with the first encoded subsequence and a second bitrate associated with the second encoded subsequence;
- 15 computing a delta visual quality score based on a first visual quality score associated with the first encoded subsequence and a second visual quality score associated with the second encoded subsequence, wherein both the first visual quality score and the second visual quality score are associated with a target resolution;
- 20 determining that the third encoded subsequence does not comply with an optimization constraint based on the delta bitrate and the delta visual quality score; and discarding the third encoded subsequence.

- 25 6. The computer-implemented method of claim 5, wherein the first visual quality score comprises a video multimethod assessment fusion score, a harmonic video multimethod assessment fusion score, or a peak-signal-to-noise ratio.

- 30 7. The computer-implemented method of claim 1, further comprising: generating a third encoded subsequence based on the first subsequence and the second resolution;
- 30 aggregating the third encoded subsequence and the second encoded subsequence to generate a second encoded video sequence that is associated with a higher bitrate than the first encoded video sequence, wherein at least a portion of the second encoded video sequence, instead of the at least a portion of the first encoded video sequence, is subsequently streamed to the endpoint device during the playback of

the media title.

8. The computer-implemented method of claim 1, wherein the at least the portion of the first encoded video sequence comprises both the first encoded subsequence and the second encoded subsequence, and the first encoded video sequence is streamed to the endpoint device via a first file and the second encoded video sequence is streamed to the endpoint device via a second file.

9. The computer-implemented method of claim 1, further comprising:
computing a first visual quality score based on the first encoded video sequence and a target resolution;
specifying the first visual quality score via a manifest file associated with the media title; and
transmitting the manifest file to the endpoint device.

10. A computer-readable storage medium including instructions that, when executed by a processor, cause the processor to perform the steps of:
generating a first encoded shot sequence based on a first shot sequence included in a source video sequence and a first resolution, wherein the source video sequence is associated with a media title;
generating a second encoded shot sequence based on a second shot sequence included in the source video sequence and a second resolution that is higher than the first resolution; and
generating a first encoded video sequence based on the first encoded shot sequence and the second encoded shot sequence,
wherein at least a portion of the first encoded video sequence is subsequently streamed to an endpoint device during a playback of the media title.

11. The computer-readable storage medium of claim 10, further comprising computing the first resolution based on a two-dimensional frequency spectrum of a first frame included in the first shot sequence.

12. The computer-readable storage medium of claim 11, further comprising computing the two-dimensional frequency spectrum based on a fast Fourier transform

algorithm and the first frame.

13. The computer-readable storage medium of claim 10, further comprising, prior to generating the first encoded video sequence:

- 5 generating a third encoded shot sequence based on the first shot sequence
 and a third resolution that is lower than the first resolution;
 computing a delta bitrate based on a first bitrate associated with the first
 encoded shot sequence and a second bitrate associated with the third
 encoded shot sequence;
- 10 computing a delta visual quality score based on a first visual quality score
 associated with the first encoded shot sequence and a second visual
 quality score associated with the third encoded shot sequence, wherein
 both the first visual quality score and the second visual quality score are
 associated with a target resolution; and
- 15 determining that the first encoded shot sequence, but not the third encoded
 shot sequence, complies with a optimization constraint based on the
 delta bitrate and the delta visual quality score.

14. The computer-readable storage medium of claim 10, further comprising, prior
20 to prior to generating the first encoded video sequence:

- generating a third encoded shot sequence based on the first shot sequence
 and the second resolution;
- computing a delta bitrate based on a first bitrate associated with the first
 encoded shot sequence and a second bitrate associated with the
25 second encoded shot sequence;
- computing a delta visual quality score based on a first visual quality score
 associated with the first encoded shot sequence and a second visual
 quality score associated with the second encoded shot sequence,
 wherein both the first visual quality score and the second visual quality
30 score are associated with a target resolution;
- determining that the third encoded shot sequence does not comply with an
 optimization constraint based on the delta bitrate and the delta visual
 quality score; and
- discarding the third encoded shot sequence.

15. The computer-readable storage medium of claim 10, further comprising:
generating a third encoded shot sequence based on the first shot sequence
and the second resolution; and
5 generating a second encoded video sequence that is associated with a higher
bitrate than the first encoded video sequence based on the third
encoded shot sequence and the second encoded shot sequence,
wherein at least a portion of the second encoded video sequence, instead of
the at least a portion of the first encoded video sequence, is
10 subsequently streamed to the endpoint device during the playback of
the media title.
16. The computer-readable storage medium of claim 15, wherein the at least the
portion of the first encoded video sequence is streamed to the endpoint device via a
15 first file that includes the first encoded video sequence, and the at least the portion of
the second encoded video sequence is streamed to the endpoint device via a second
file that includes the second encoded video sequence.
17. The computer-readable storage medium of claim 10, further comprising:
20 computing a first visual quality score based on the first encoded video
sequence and a target resolution;
specifying the first visual quality score via a manifest file associated with the
media title; and
transmitting the manifest file to the endpoint device.
- 25 18. The computer-readable storage medium of claim 17, wherein the first visual
quality score comprises a video multimethod assessment fusion score, a harmonic
video multimethod assessment fusion score, or a peak-signal-to-noise ratio.
- 30 19. A system, comprising:
a memory storing instructions; and
a processor that is coupled to the memory and, when executing the
instructions, is configured to:
generate a first encoded shot sequence based on a first shot sequence

included in a source video sequence and a first resolution;
generate a second encoded shot sequence based on a second shot
sequence included in the source video sequence and a second
resolution that is higher than the first resolution; and
5 aggregate the first encoded shot sequence and the second encoded
shot sequence to generate a first encoded video sequence,
wherein at least a portion of the first encoded video sequence is
subsequently streamed to an endpoint device during a playback
of a media title that is associated with the source video sequence.

10

20. The system of claim 19, wherein the processor is further configured to:
generate a third encoded shot sequence based on the first shot sequence and
the second resolution; and
aggregate the third encoded shot sequence and the second encoded shot
15 sequence to generate a second encoded video sequence that is
associated with a higher bitrate than the first encoded video sequence,
wherein at least a portion of the second encoded video sequence, instead of
the at least a portion of the first encoded video sequence, is
subsequently streamed to the endpoint device during the playback of
20 the media title.

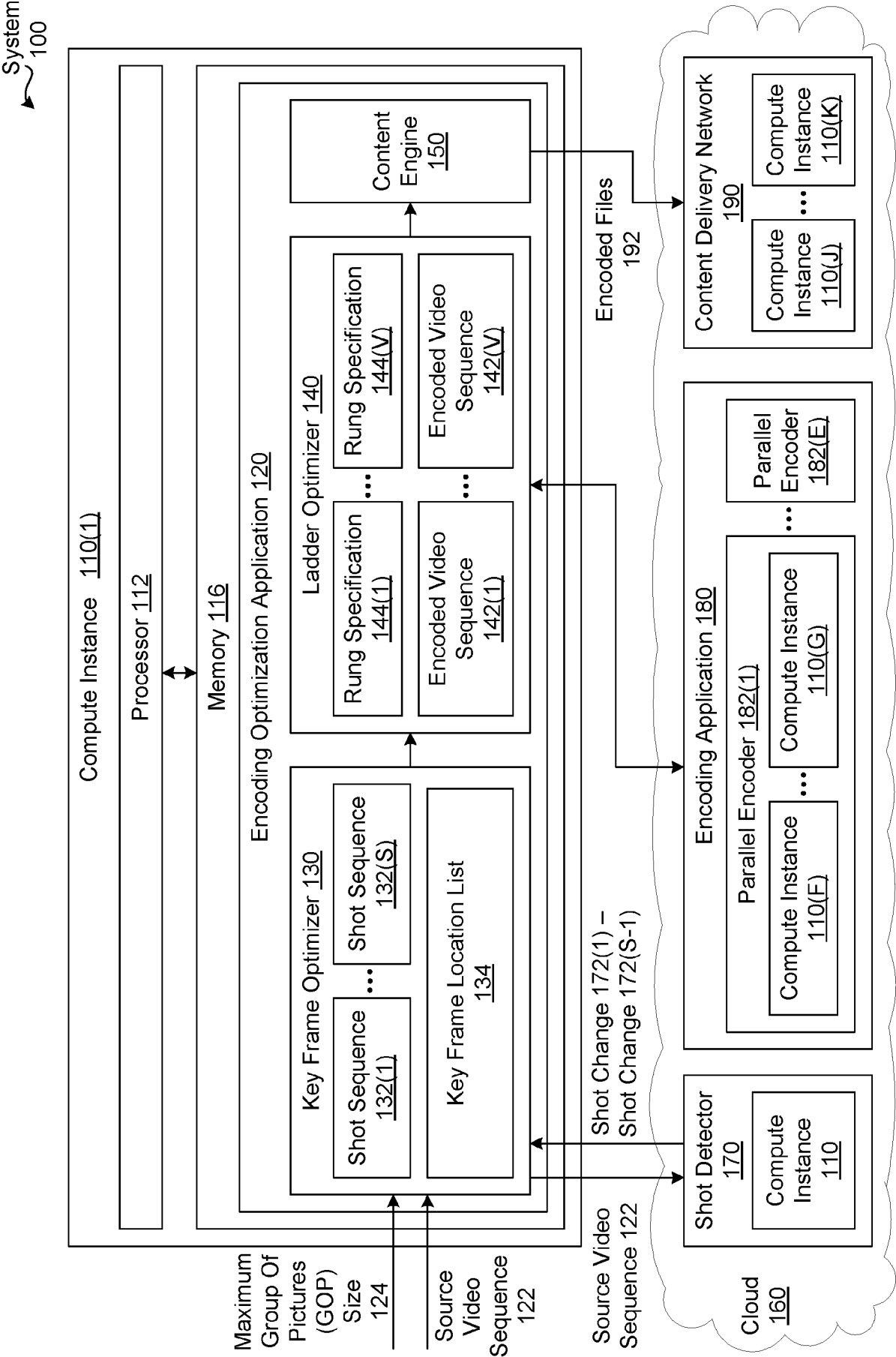


FIGURE 1

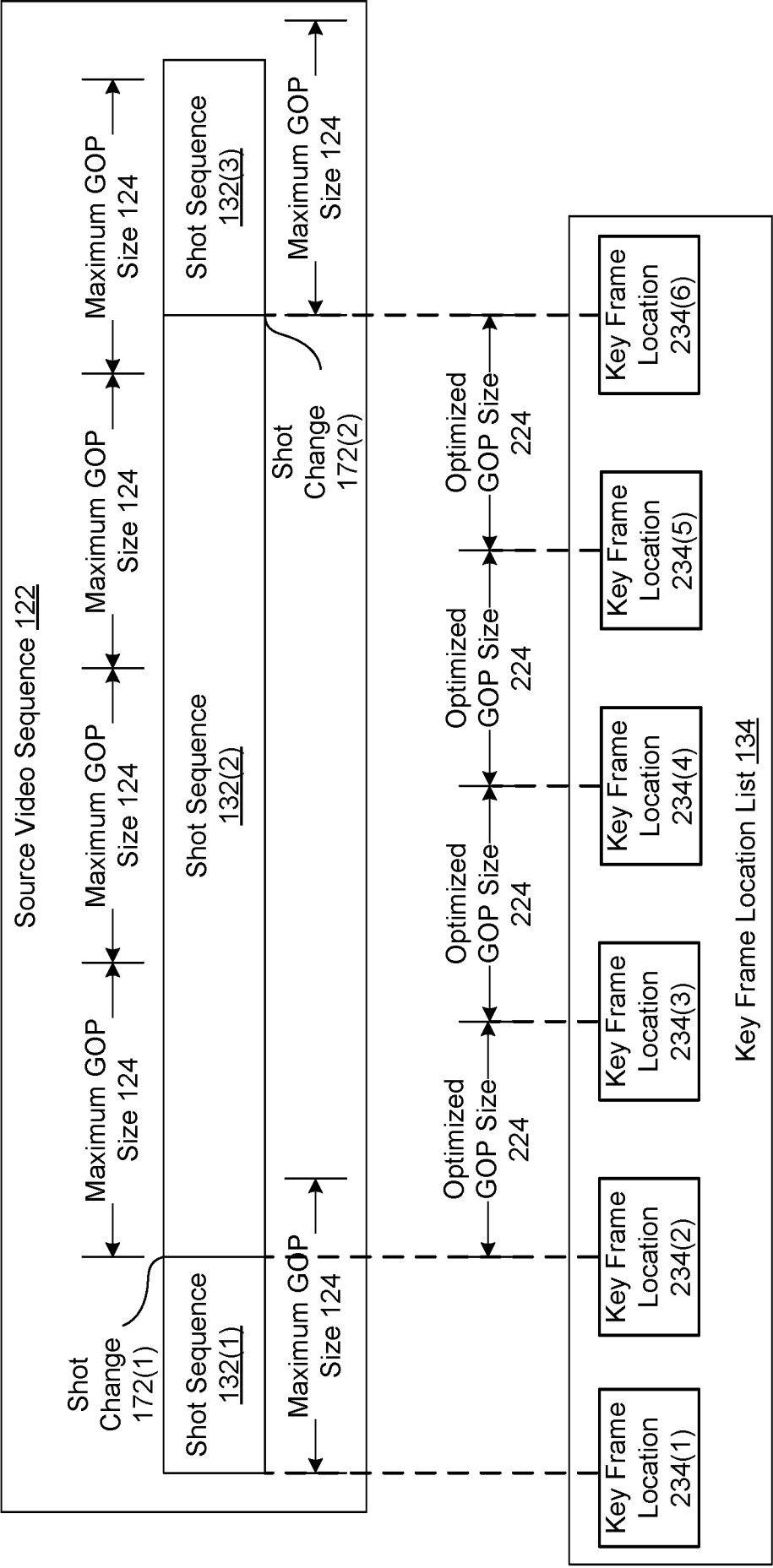


FIGURE 2

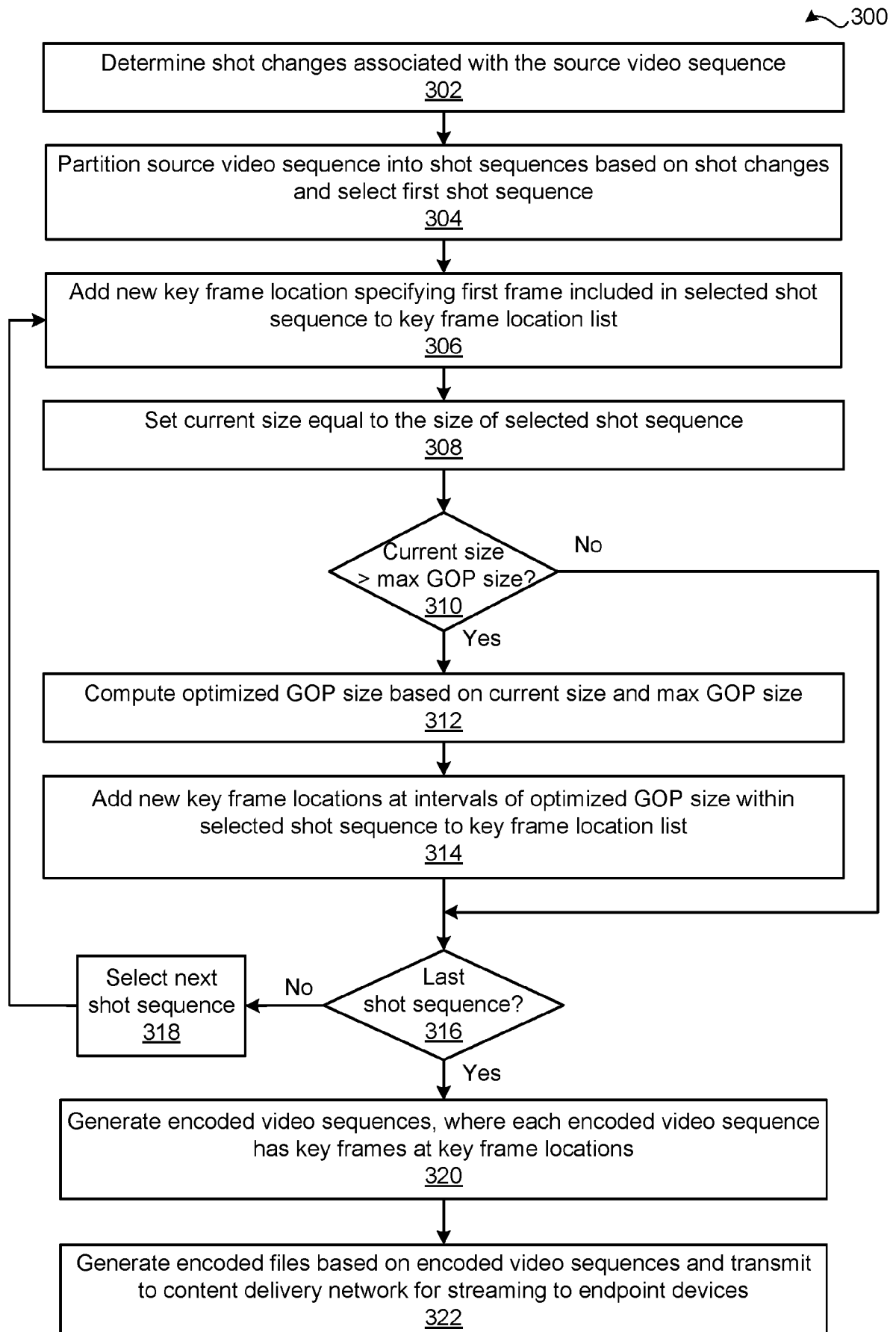
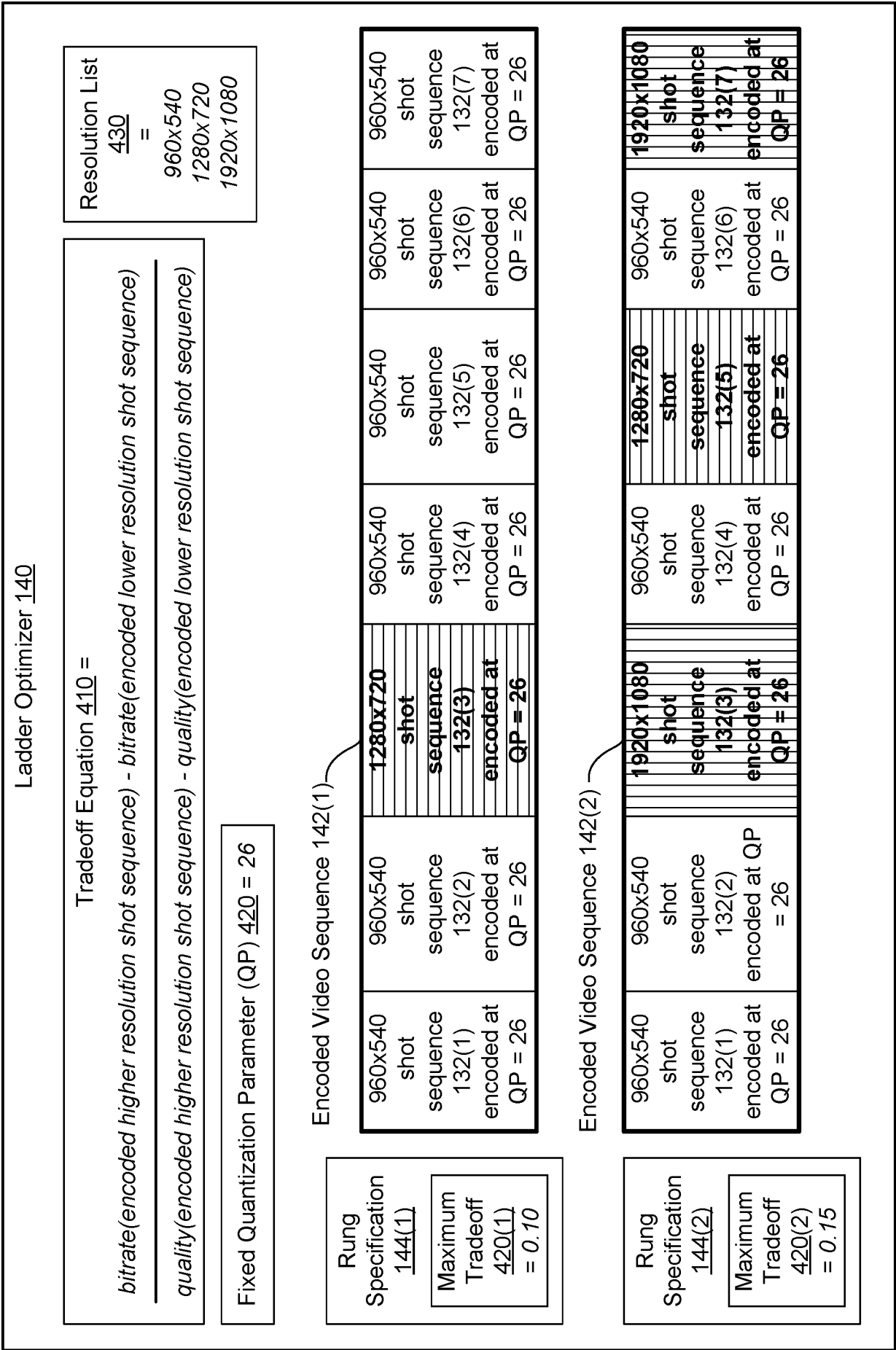


FIGURE 3



500

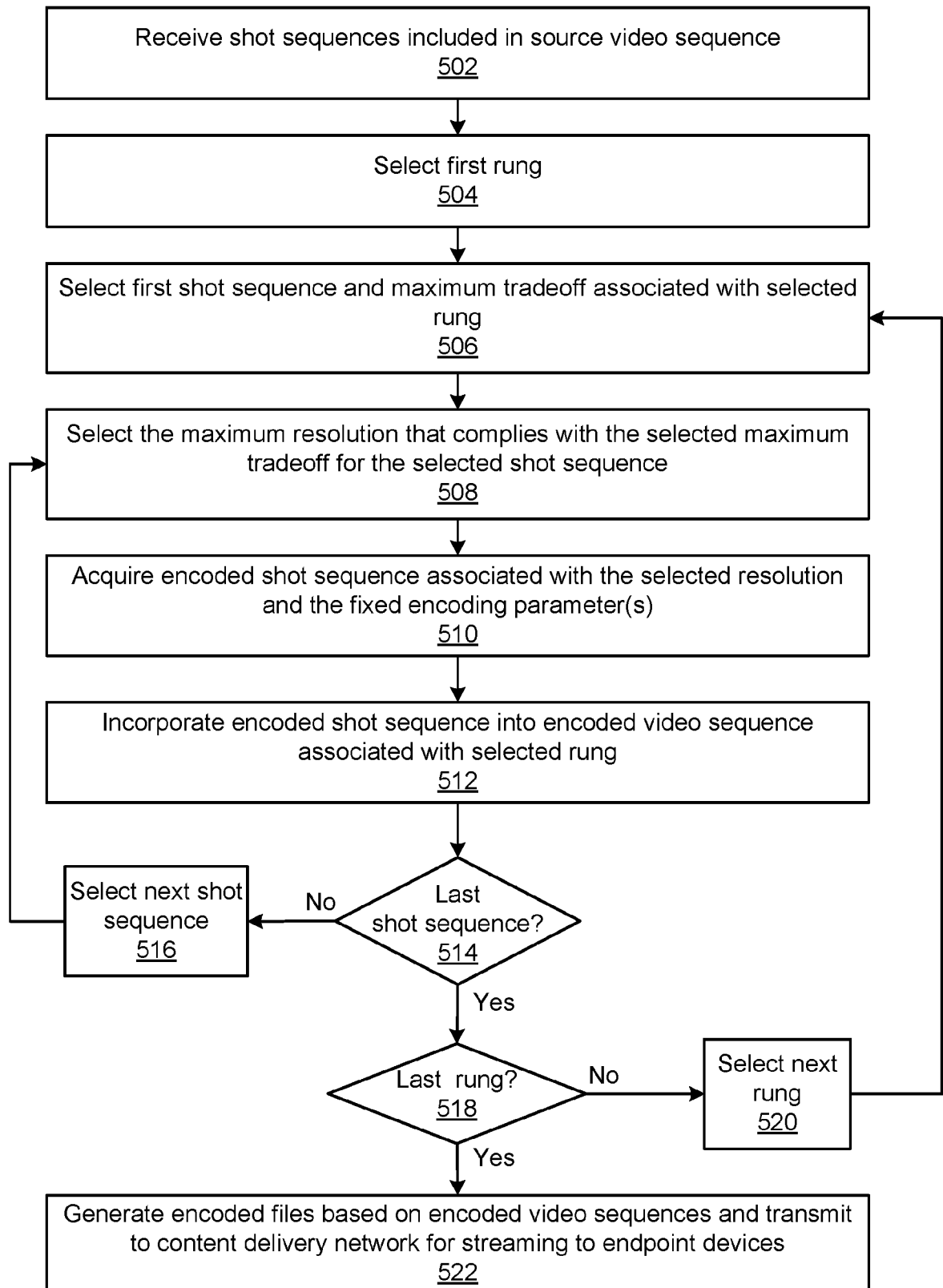


FIGURE 5

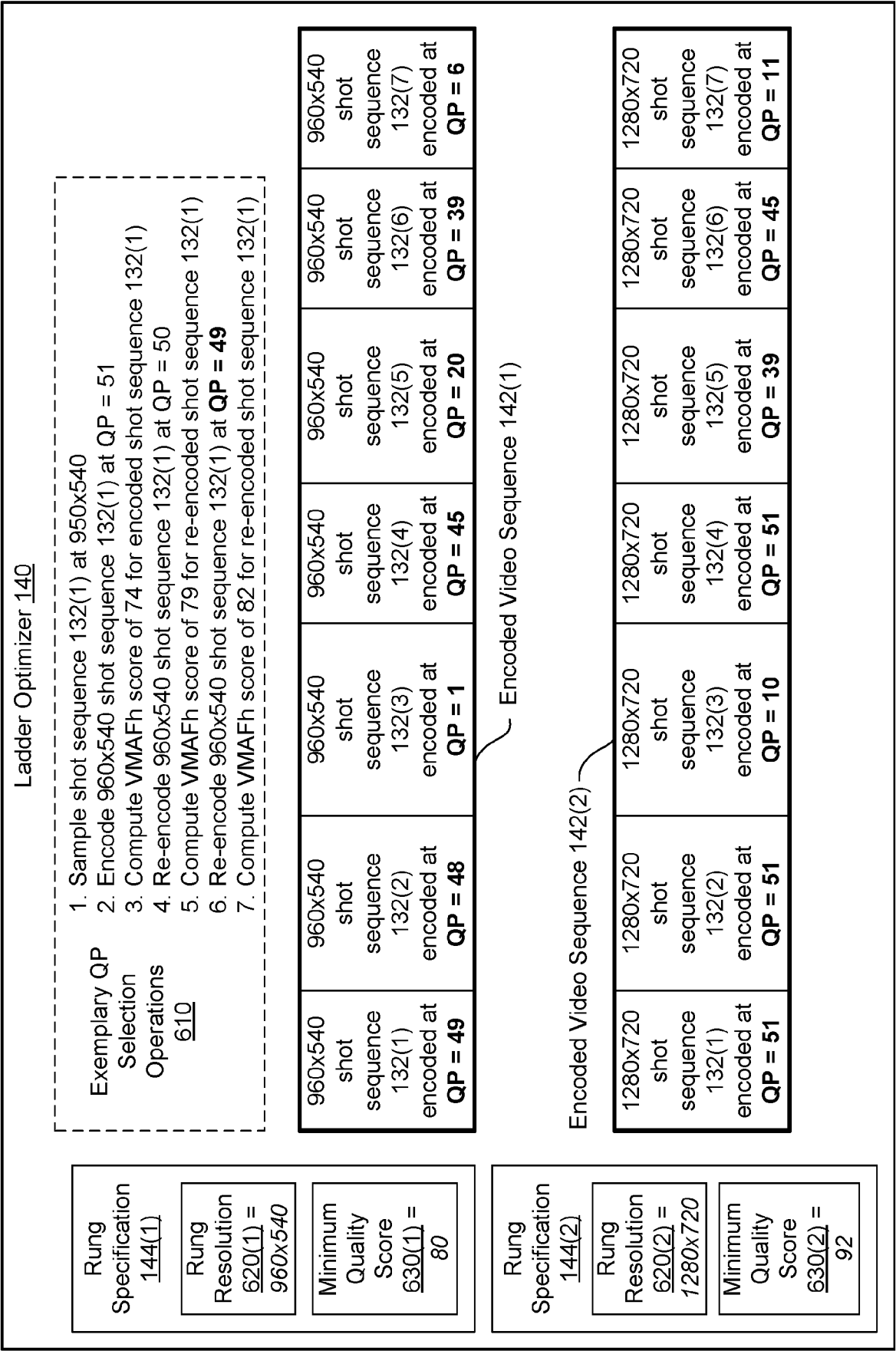


FIGURE 6

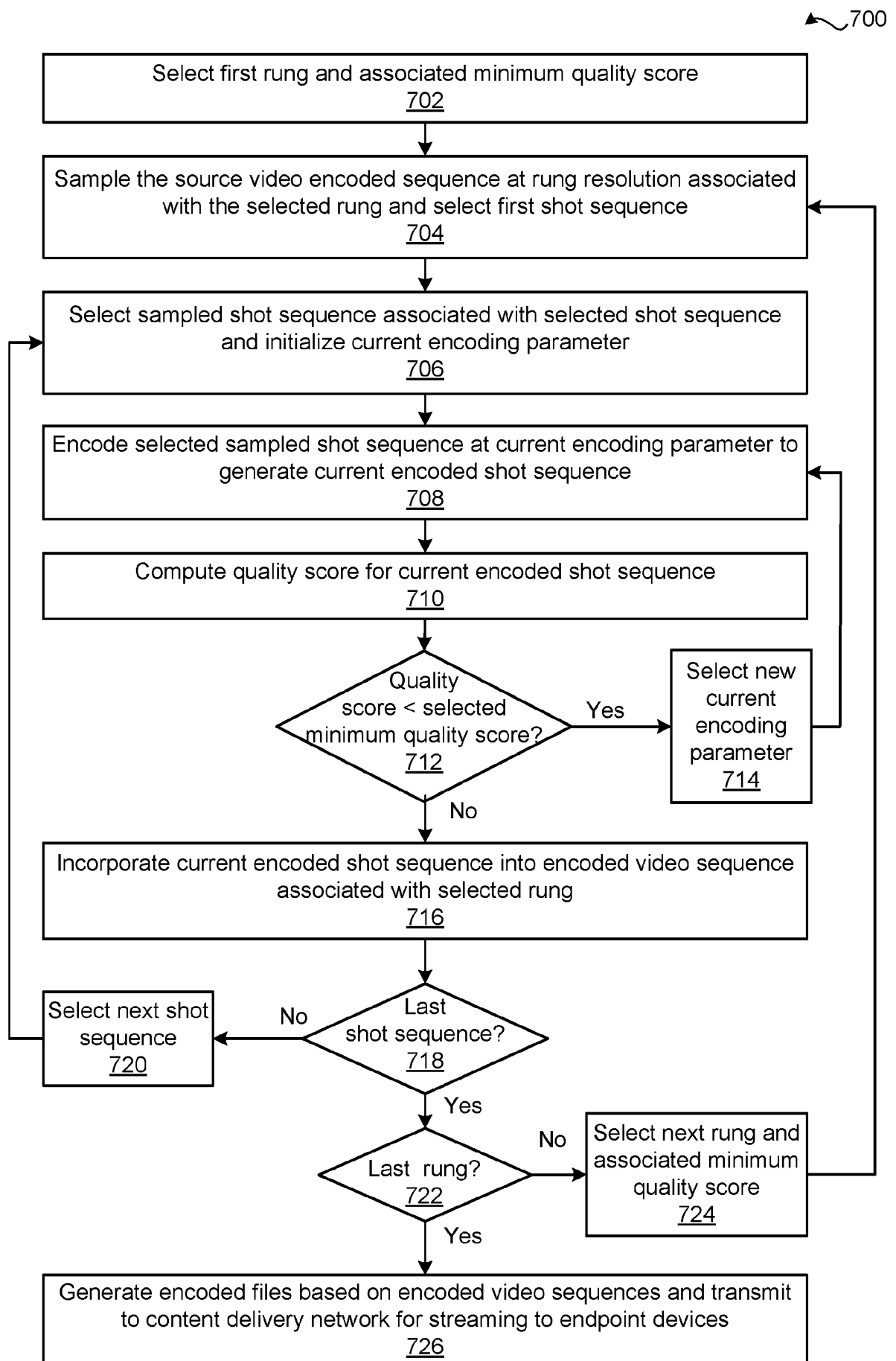


FIGURE 7

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2018/019575

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N21/2343 H04N21/238 H04N21/845
ADD.

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

B. FIELDS SEARCHED

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

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

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

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2010/189183 A1 (GU CHUANG [US] ET AL) 29 July 2010 (2010-07-29)	1,2,4-8, 10, 13-16, 19,20
Y	abstract; figures 1,3-5 paragraph [0019] - paragraph [0024] paragraph [0040] - paragraph [0063] -----	3,9,11, 12,17,18
X	US 2013/329781 A1 (SU YEPING [US] ET AL) 12 December 2013 (2013-12-12)	1,2,4-8, 10, 13-16, 19,20
Y	figures 1-3, 6(a), 10 paragraph [0038] - paragraph [0047] paragraph [0059] paragraph [0075] ----- -/-	3,9,11, 12,17,18



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search

28 March 2018

Date of mailing of the international search report

09/04/2018

Name and mailing address of the ISA/

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

Authorized officer

Kopilovic, Ivan

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2018/019575

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2012/195369 A1 (GUERRERO RODOLFO VARGAS [US]) 2 August 2012 (2012-08-02)	1,2,4-8, 10, 13-16, 19,20
Y	abstract; figure 2 paragraph [0026] - paragraph [0029] -----	3,9,11, 12,17,18
X	EP 2 410 749 A1 (FREEBOX [FR]) 25 January 2012 (2012-01-25)	1,2,4-8, 10, 13-16, 19,20
Y	abstract; figure 1 paragraph [0030] - paragraph [0051] -----	3,9,11, 12,17,18
Y	US 2016/379057 A1 (KATSAVOUNIDIS IOANNIS [US]) 29 December 2016 (2016-12-29) abstract paragraph [0009] -----	3,11,12
Y	US 2004/161158 A1 (KONDO SATOSHI [JP] ET AL) 19 August 2004 (2004-08-19) abstract; figures 1-4 paragraph [0125] - paragraph [0157]; figures 25-29 -----	3,11,12
Y	US 2014/201324 A1 (ZHANG SHAOBO [CN] ET AL) 17 July 2014 (2014-07-17) abstract; figures 4,5 -----	9,17,18

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2018/019575

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2010189183 A1	29-07-2010	AU 2010208597 A1 BR PI1006824 A2 CA 2748373 A1 CN 102301710 A DK 2382784 T3 EP 2382784 A2 HK 1163405 A1 IL 213566 A JP 5490146 B2 JP 2012516644 A KR 20110106419 A RU 2011131864 A US 2010189183 A1 WO 2010088030 A2	14-07-2011 12-04-2016 05-08-2010 28-12-2011 06-07-2015 02-11-2011 15-01-2016 26-02-2015 14-05-2014 19-07-2012 28-09-2011 10-02-2013 29-07-2010 05-08-2010
US 2013329781 A1	12-12-2013	TW 201404170 A US 2013329781 A1 WO 2013184374 A1	16-01-2014 12-12-2013 12-12-2013
US 2012195369 A1	02-08-2012	AU 2012211243 A1 AU 2016250476 A1 CA 2825929 A1 CN 103493481 A EP 2668779 A2 JP 6134650 B2 JP 2014511137 A KR 20140034149 A TW 201238356 A US 2012195369 A1 WO 2012103326 A2	22-08-2013 17-11-2016 02-08-2012 01-01-2014 04-12-2013 24-05-2017 08-05-2014 19-03-2014 16-09-2012 02-08-2012 02-08-2012
EP 2410749 A1	25-01-2012	EP 2410749 A1 FR 2963189 A1	25-01-2012 27-01-2012
US 2016379057 A1	29-12-2016	AU 2016284583 A1 CA 2989430 A1 KR 20180020238 A US 2016379057 A1 US 2016381401 A1 US 2018012076 A1	18-01-2018 29-12-2016 27-02-2018 29-12-2016 29-12-2016 11-01-2018
US 2004161158 A1	19-08-2004	US 6625322 B1 US 2004161158 A1 US 2005238243 A1	23-09-2003 19-08-2004 27-10-2005
US 2014201324 A1	17-07-2014	CN 105052107 A EP 2939420 A1 JP 6270873 B2 JP 2016508348 A JP 2018023133 A KR 20150106961 A US 2014201324 A1 US 2017111426 A1 WO 2014113486 A1	11-11-2015 04-11-2015 31-01-2018 17-03-2016 08-02-2018 22-09-2015 17-07-2014 20-04-2017 24-07-2014