

(12) STANDARD PATENT
(19) AUSTRALIAN PATENT OFFICE

(11) Application No. **AU 2012211249 B2**

(54) Title
Encoding of video stream based on scene type

(51) International Patent Classification(s)
H04N 19/124 (2014.01)

(21) Application No: **2012211249**

(22) Date of Filing: **2012.01.26**

(87) WIPO No: **WO12/103332**

(30) Priority Data

(31) Number
61/437,193
61/437,211

(32) Date
2011.01.28
2011.01.28

(33) Country
US
US

(43) Publication Date: **2012.08.02**

(44) Accepted Journal Date: **2016.10.20**

(71) Applicant(s)
Eye IO, LLC

(72) Inventor(s)
Guerrero, Rodolfo Vargas

(74) Agent / Attorney
FPA Patent Attorneys Pty Ltd, Level 43 101 Collins Street, Melbourne, VIC, 3000

(56) Related Art
US 2001/0017887 A1
US 2007/0253699 A1
US 6,026,217 A



- (51) International Patent Classification:
H04N 7/26 (2006.01) *H04N 7/32* (2006.01)
- (21) International Application Number:
PCT/US2012/022720
- (22) International Filing Date:
26 January 2012 (26.01.2012)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
61/437,193 28 January 2011 (28.01.2011) US
61/437,211 28 January 2011 (28.01.2011) US
- (71) Applicant (for all designated States except US): **EYE IO, LLC** [US/US]; 165 University Avenue, Suite #3, Palo Alto, California 94301 (US).
- (72) Inventor; and
(75) Inventor/Applicant (for US only): **GUERRERO, Rodolfo Vargas** [MX/US]; 165 University Avenue, Suite #3, Palo Alto, California 94301 (US).
- (74) Agents: **COLEMAN, Brian R.** et al.; Perkins Coie LLP, P.O. Box 1208, Seattle, Washington 98111-1208 (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, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

[Continued on next page]

(54) Title: ENCODING OF VIDEO STREAM BASED ON SCENE TYPE

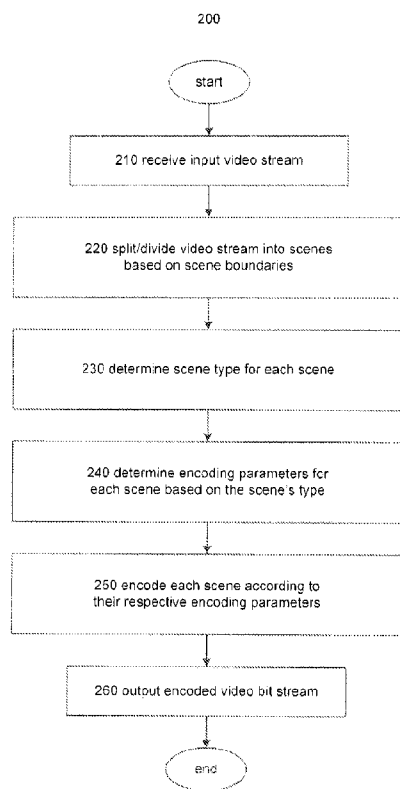


FIG. 2

(57) Abstract: An encoder for encoding a video stream or an image is described herein. The encoder receives an input video stream and outputs an encoded video stream that can be decoded at a decoder to recover, at least approximately, an instance of the input video stream. The encoder encodes a video stream by first identifying scene boundaries and encoding frames between scene boundaries using a set of parameters. For at least two different scene sequences, different sets of parameters are used, providing adaptive, scene-based encoding.



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

Published:

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

(88) Date of publication of the international search report:

1 November 2012

ENCODING OF VIDEO STREAM BASED ON SCENE TYPE

CROSS-REFERENCE TO RELATED APPLICATIONS

5

[0001] This application claims priority to U.S. Provisional Patent Application No. 61/437,193 filed January 28, 2011, and U.S. Provisional Patent Application No. 61/437,211 filed January 28, 2011, the contents of which are expressly incorporated by reference herein.

FIELD OF THE INVENTION

[0002] The present invention relates to video encoding techniques, and more particularly, to automatic selection of video encoding parameters for video encoding.

BACKGROUND

[0003] While video streaming continues to grow in popularity and usage among everyday users, there are several inherent limitations that need to be overcome. For example, users often want to watch a video over the Internet having only a limited bandwidth for obtaining that video stream. In instances, users might want to obtain the video stream over a mobile telephone connection or a home wireless connection. In some scenarios, users compensate for the lack of adequate bandwidth by spooling content (i.e., download content to local storage for eventual viewing). This method is rife with several disadvantages. First, the user is unable to have a real “run-time” experience – that is, the user is unable to view a program when he decides to watch it. Instead, he has to experience significant delays for the content to be spooled prior to viewing the program. Another disadvantage is in the availability of storage – either the provider or the user has to account for storage resources to ensure that the spooled content can be stored, even if for a short period of time, resulting in unnecessary utilization of expensive storage resources.

[0004] A video stream (typically containing an image portion and an audio portion) can require considerable bandwidth, especially at high resolution (e.g., HD videos). Audio typically requires much less bandwidth, but still sometimes needs to be taken into account. One streaming video approach is to heavily compress the video stream enabling rapid video delivery to allow a user to view content in run-time or substantially instantaneously (i.e., without experiencing substantial spooling delays). Typically, lossy compression (i.e.,

compression that is not entirely reversible) provides more compression than lossless compression, but heavy lossy compression provides an undesirable user experience.

[0005] In order to reduce the bandwidth required to transmit digital video signals, it is well known to use efficient digital video encoding where the data rate of a digital video signal may be substantially reduced (for the purpose of video data compression). In order to ensure interoperability, video encoding standards have played a key role in facilitating the adoption of digital video in many professional- and consumer applications. Most influential standards are traditionally developed by either the International Telecommunications Union (ITU-T) or the MPEG (Motion Pictures Experts Group) 15 committee of the ISO/IEC (the International Organization for Standardization/the International Electrotechnical Committee. The ITU-T standards, known as recommendations, are typically aimed at real-time communications (e.g. videoconferencing), while most MPEG standards are optimized for storage (e.g. for Digital Versatile Disc (DVD) and broadcast (e.g. for Digital Video Broadcast (DVB) standard).

[0006] At present, the majority of standardized video encoding algorithms are based on hybrid video encoding. Hybrid video encoding methods typically combine several different lossless and lossy compression schemes in order to achieve desired compression gain. Hybrid video encoding is also the basis for ITU-T standards (H.26x standards such as H.261, H.263) as well as ISO/IEC standards (MPEG-X standards such as MPEG-1, MPEG-2, and MPEG-4). The most recent and advanced video encoding standard is currently the standard denoted as H.264/MPEG-4 advanced video coding (AVC) which is a result of standardization efforts by joint video team (JVT), a joint team of ITU-T and ISO/IEC MPEG groups.

[0007] The H.264 standard employs the same principles of block-based motion compensated hybrid transform coding that are known from the established standards such as MPEG-2. The H.264 syntax is, therefore, organized as the usual hierarchy of headers, such as picture-, slice- and macro-block headers, and data, such as motion-vectors, block-transform coefficients, quantizer scale, etc. However, the H.264 standard separates the Video Coding Layer (VCL), which represents the content of the video data, and the Network Adaptation Layer (NAL), which formats data and provides header information.

[0008] Furthermore, H.264 allows for a much increased choice of encoding parameters. For example, it allows for a more elaborate partitioning and manipulation of 16x16 macro-blocks whereby e.g. motion compensation process can be performed on segmentations of a macro-block as small as 4x4 in size. Also, the selection process for motion compensated prediction of a sample block may involve a number of stored

previously-decoded pictures, instead of only the adjacent pictures. Even with intra coding within a single frame, it is possible to form a prediction of a block using previously-decoded samples from the same frame. Also, the resulting prediction error following motion compensation may be transformed and quantized based on a 4x4 block size, instead of the traditional 8x8 size. Additionally, an in-loop deblocking filter that reduces block artifacts may be used.

[0009] The H.264 standard may be considered a superset of the H.262 / MPEG-2 video encoding syntax in that it uses the same global structuring of video data while extending the number of possible coding decisions and parameters. A consequence of having a variety of coding decisions is that a good trade-off between the bit rate and picture quality may be achieved. However, although it is commonly acknowledged that while the H.264 standard may significantly reduce typical artifacts of block-based coding, it can also accentuate other artifacts. The fact that H.264 allows for an increased number of possible values for various coding parameters thus results in an increased potential for improving the encoding process, but also results in increased sensitivity to the choice of video encoding parameters.

[0010] Similar to other standards, H.264 does not specify a normative procedure for selecting video encoding parameters, but describes through a reference implementation, a number of criteria that may be used to select video encoding parameters such as to achieve a suitable trade-off between coding efficiency, video quality and practicality of implementation. However, the described criteria may not always result in an optimal or suitable selection of coding parameters suitable for all kind of contents and applications. For example, the criteria may not result in selection of video encoding parameters optimal or desirable for the characteristics of the video signal or the criteria may be based on attaining characteristics of the encoded signal which are not appropriate for the current application.

[0011] Accordingly, an improved system for video encoding would be advantageous.

[0012] The foregoing examples of the related art and limitations related therewith are intended to be illustrative and not exclusive. Other limitations of the related art will become apparent upon a reading of the specification and a study of the drawings.

2012211249 09 Sep 2016

SUMMARY

[00013] The present teaching contemplates a variety of methods, systems and encoders for encoding a video stream. Introduced herein is an encoder for encoding a video stream. The encoder receives an input video stream and outputs an encoded video stream that can be decoded at a decoder to recover, at least approximately, an instance of the input video stream, In 5 embodiments of the present invention, an encoder encodes a video stream or stored sequence by first identifying scene boundaries and encoding frames between scene boundaries (i.e., "a scene sequence" comprising one or more image frames) using a set of parameters.

[00013A] In a first aspect of the present invention there is provided a method for encoding a video stream using scene types each having a predefined set of one or more of a plurality of encoder parameters used by a video encoder to encode any given scene type, the method comprising: receiving an input video stream; dividing the input video stream into a plurality of scenes based on scene boundaries, each scene comprising a plurality of temporally contiguous image frames, wherein a given scene boundary is determined according to 15 relatedness of two temporally contiguous image frames in the input video stream by determining the difference between two image frames that are temporarily contiguous to each other on the input video stream's timeline and when the difference exceeds a set threshold that signals a scene change, the two image frames are determined to be part of two different scene sequences and therefore a scene boundary is established between the two image frames; analysing the scenes 20 and assigning a scene type for each of the plurality of scenes; and encoding each of the plurality of scenes using encoding parameters according to the assigned scene type.

[00013B] In a second aspect of the present invention there is provided a video encoding apparatus for encoding a video stream using scene types each having a predefined set of one or more of a plurality of encoder parameters used by the video encoder to encode any given scene 25 type, the apparatus comprising: an input module for receiving an input video stream; a video processing module to divide the video stream into a plurality of scenes based on scene boundaries, each scene comprising a plurality of temporally contiguous image frames, wherein the video processing module determines a given scene boundary according to the relatedness of two temporally contiguous image frames in the input video stream by determining the difference 30 between two image frames that are temporarily contiguous to each other on the input video stream's timeline and when the difference exceeds a set threshold that signals a scene change,

2012211249 09 Sep 2016

the two image frames are determined to be part of two different scene sequences and therefore a scene boundary is established between the two image frames; the video processing module to analyse and assign a scene type for each of the plurality of scenes; and a video encoding module to encode each of the plurality of scenes using encoding parameters according to the assigned scene type.

[00014] In one embodiment of the present invention, a scene change can be identified in the video stream where the camera suddenly changes from one viewing angle to another such that the difference between two frames on each side of a scene change is not as compressible as other frames shot from the same viewing angle. For at least two different scene sequences, different sets of parameters are used, this providing adaptive, scene-based encoding.

[00015] This Summary is provided to introduce a selection of concepts in a simplified form that are further described below in the Detailed Description. This Summary is not intended to identify key features or essential features of the claimed subject matter, not is it intended to be used to limit the scope of the claimed subject matter.

[00015A] As used herein, except where the context requires otherwise the term 'comprise' and variations of the term, such as 'comprising', 'comprises' and 'comprised', are not intended to exclude other additives, components, integers or steps.

[00015B] Reference to any prior art in the specification is not an acknowledgment or suggestion that this prior art forms part of the common general knowledge in any jurisdiction or that this prior art could reasonably be expected to be understood, regarded as relevant, and/or combined with other pieces of prior art by a skilled person in the art.

BRIEF DESCRIPTION OF THE DRAWINGS

[00016] One or more embodiments of the present invention are illustrated by way of example and are not limited by the figures of the accompanying drawings, in which like references indicate similar elements.

[00017] FIG. 1 illustrates an example of an encoder.

[00018] FIG. 2 illustrates steps of a method for encoding an input video stream.

2012211249 09 Sep 2016

[00019] FIG. 3 illustrates steps of a method for dividing an input video stream into a plurality of scene sequences.

[00020] FIG. 4 illustrates steps of a method for determining a scene sequence's scene type.

5 [00021] Fig. 5 is a block diagram explaining motion estimation in a frame.

[00022] FIG. 6 is a block diagram of a processing system that can be used to implement an encoder implementing certain techniques described herein.

DETAILED DESCRIPTION

[00023] Various aspects of the invention will now be described. The following description provides specific details for a thorough understanding and enabling description of these examples. One skilled in the art will understand, however, that the invention may be practiced without many of these details. Additionally, some well-known structures or functions may not be shown or described in detail, so as to avoid unnecessarily obscuring the relevant description. Although the diagrams depict components as functionally separate, such depiction is merely for illustrative purposes. It will be apparent to those skilled in the art that the components portrayed in this figure may be arbitrarily combined or divided into separate components.

[00024] The terminology used in the description presented below is intended to be interpreted in its broadest reasonable manner, even though it is being used in conjunction with a detailed description of certain specific examples of the invention. Certain terms may even be emphasized below; however, any terminology intended to be interpreted in any restricted manner will be overtly and specifically defined as such in this Detailed Description section.

[00025] References in this specification to “an embodiment,” “one embodiment,” or the like mean that the particular feature, structure, or characteristic being described is included in at least one embodiment of the present invention. Occurrences of such phrases in this specification do not necessarily all refer to the same embodiment.

[00026] In one embodiment of the present invention, an encoder is provided to receive an input video stream and output an encoded video stream that can be decoded at a decoder to recover, at least approximately, an instance of the input video stream. The encoder comprises: an input module for receiving an input video stream; a video processing module to divide the video stream into a plurality of scenes based on scene boundaries, wherein the video processing module determines a given scene boundary according to the relatedness of two temporally contiguous image frames in the input video stream; the video processing module to further determine a scene type for each of the plurality of scenes, each scene type being associated with one or more of a plurality of predetermined encoder parameters used by a video encoder to encode the given scene type; and a video encoding module to encoding each of the plurality of scenes according to the scene type associated with each of the plurality of scenes.

[00027] In this manner, the encoder can encode an input video stream at a quality that best suits each of the scenes in the input video stream being encoded.

[00028] FIG. 1 illustrates an example of an encoder 100, according to one embodiment of the present invention. The encoder 100 receives an input video stream 110 and outputs an encoded video stream 120 that can be decoded at a decoder to recover, at least approximately, an instance of the input video stream 110. The encoder 100 comprises an input module 102, a video processing module 104, and a video encoding module 106. The encoder 100 may be implemented in hardware, software, or any suitable combination. The encoder 100 may include other components such as a parameter input module, memory for storing parameters, etc. The encoder 100 may perform other video processing functions not specifically described herein.

[00029] The input module 102 receives the input video stream 110. The input video stream 110 may take any suitable form, and may originate from any of a variety of suitable sources such as memory, or even from a live feed.

[00030] The video processing module 104 analyzes an input video stream 110 and splits the video stream 110 into a plurality of scenes along with their respective video encoding parameters for each of the plurality of scenes. In one embodiment, video processing module 104 divides the video stream into a plurality of scenes based on scene boundaries, wherein the scene boundaries are determined according to the relatedness of two temporally contiguous image frames in the input video stream. The video processing module 104 further determines a scene type for each of the plurality of scenes. Finally, the video processing module 104 determines video encoding parameters used by a video encoder 106 to encode each scene by associating each scene type with one or more of a plurality of predetermined encoder parameters. The parameters may be predefined for each scene type, or may be calculated and/or adapted during the video stream processing. The video encoding module 106 receives a plurality of scenes and their respective video encoding parameters from the video processing module 104 to encode each of the plurality of scenes according to their respective encoding parameters and output an encoded video stream 120.

[00031] FIG. 2 illustrates steps of a method 200 for encoding an input video stream. The method 200 encodes the input video stream to an encoded video bit stream that can be decoded at a decoder to recover, at least approximately, an instance of the input video stream. At step 210, the method receives an input video stream to be encoded. At step 220, the video stream is split into a plurality of scenes based on scene boundaries. Here, as discussed in further detail below with reference to FIG. 3, the method determines scene boundaries according to the relatedness of two temporally contiguous image frames in the input video stream. However, any of a variety of other suitable mechanisms may be utilized to

distinguish between scene types. Then in step 230, the process determines a scene type for each of the plurality of scenes. At step 240, the process determines video encoding parameters to encode each scene by mapping each scene's type with appropriate predetermined encoder parameters, as is also discussed in further detail below. At Step 250, the process encodes the scenes according to each scene's respective video encoding parameters (as, for example, determined in step 240). At step 260, the process outputs the encoded video bit stream.

[00032] The above process is elaborated in more detail in the following sections. The input video stream typically includes multiple image frames. Each image frame can typically be identified based on a distinct "time position" in the input video stream. In embodiments, the input video stream can be a stream that is made available to the encoder in parts or discrete segments. In such instances, the encoder outputs the encoded video bit stream (for example, to a final consumer device such as a HDTV) as a stream on a rolling basis before even receiving the entire input video stream.

[00033] In embodiments, the input video stream and the encoded video bit stream are stored as a sequence of streams. Here, the encoding may be performed ahead of time and the encoded video streams may then be streamed to a consumer device at a later time. Here, the encoding is completely performed on the entire video stream prior to being streamed over to the consumer device. It is understood that other examples of pre, post, or "inline" encoding of video streams, or a combination thereof, as may be contemplated by a person of ordinary skill in the art, are also contemplated in conjunction with the techniques introduced herein.

[00034] In embodiments, scene boundaries in an input video stream are determined by first scaling and removing any high frequency elements present in each image frame. Next, the difference between two image frames that are temporally contiguous to each other on the input video stream's timeline is determined. In some instances, for example, the difference between two images may be discriminated using recursive or adaptive filters. When the computed difference exceeds a set threshold that signals a scene change, the two image frames are determined to be part of two different scene sequences and therefore, a scene boundary is established between the two image frames. By repeating the scene boundary determination process between temporally contiguous image frames, the input video stream can be divided into, for example, an ordered set of scene sequences.

[00035] In embodiments, as illustrated in reference to Figure 2 above, a scene type may be determined for each of the plurality of scene sequence in conjunction with the encoding process. In some instances, a scene sequence type may be determined utilizing one or more

of the following scene sequence parameters: (i) position of the scene sequence in the input stream's timeline; (ii) the scene sequence's length; (iii) its motion vector estimation; (iv) the scene sequence's effective difference from previous frames; (v) the scene sequence's spectral data size; (vi) the scene's textual content determined using optical character recognition; (vii) the scene's screenplay attributes based on screenplay structure information; etc. Additionally, in some instances, facial recognition may be used in the determination of a scene type to determine whether the scene sequence involves faces of individuals.

[00036] A given scene type may include, for example, "fast motion", "static", "talking head", "text", "scroll credits", "mostly black images", "short scene of five frames or less", etc. In some instances, scene sequences may not be assigned a particular scene type. In other instances, assigned scene sequences might include scene types, "miscellaneous", "unknown", "default", etc.

[00037] In embodiments, once the scene types are assigned, the scene sequences are encoded. In some instances, such encoding is performed by running a parameterized encoding process according to a set of software or hardware instructions. Here, in some instances, a set of highly optimized parameters may be utilized to control details of the encoding according to the scene type. The plurality of parameters might be stored in a scene type database or other data structure or machine learning system. In an illustrative example, a database stored in memory and accessible by the encoder might have the structure illustrated in Table 1. Parameters that are used for encoding, but are not specifically set in the scene type database, may utilize a default parameter value determined at the beginning of the encoding process. In some instances, a default parameter value may be determined based on a value recommended by an encoding standard utilized to encode the input video stream.

TABLE 1.

Scene Type	Motion Estimation Range Search	Quantizers	Reference Frame Numbers	Parameters 4 to N...
fast motion	High Motion Estimation Range Search	Low Quantizers with high deblocking	Low Reference Frame Numbers	
Static	Low motion Estimation Range Search	High Quantizers	High Reference Frame Numbers	
talking head	Low motion Estimation Range Search	Medium Range Quantizers with emphasis on facial areas	Low Reference Frame Numbers	
Text	Low motion Estimation Range Search	Low Quantizers, low deblocking	Medium Reference Frame Numbers	
mostly black	Low motion Estimation Range Search	Very Low Quantizers, no deblocking	No Reference Frame Numbers	
short scene	High motion Estimation Range Search	Low Quantizers	Low Reference Frame Numbers	
Default	Medium motion Estimation Range Search	Medium Quantizers	Medium Reference Frame Numbers	
...				

[00038] FIG. 3 illustrates steps of a method or process 300 for determining scene boundaries in an input video stream. At step 310, the process scales high frequency elements from a current frame (i) and a previous frame (i-1) for which scene boundaries need to be determined. In at least some embodiments, at step 320, the process removes high frequency elements from the current frame (i) and the previous frame (i-1). In one embodiment, a transform coder converts pixel data in an image frame into frequency coefficients. In the frequency domain, low frequency data has greater human perceptual importance than high frequency data. Steps 310 and 320 allow the analysis to be based on the perceptually important low frequency elements of the frame.

[00039] At step 330, a luma of the current frame (i) is computed. The luma value, also known as luminescence, represents the brightness of an image (the "black-and-white" or achromatic portion of the image).

[00040] At step 340, a luma value of a projection based on the current (i) and previous (i-1) frame is computed. The projection is that of the current frame (i)'s onto a subspace based on the previous frame (i-1). The subspace is obtained by a singular value decomposition of the previous frame (i-1).

[00041] At step 350, a residual value based on the difference between the luma values obtained in step 330 and 340 is computed. At step 360, the process filters any residual value using, for example, recursive or adaptive filters and maps the residual value on a score range of 0 - 1. The adaptive filter helps filter out any artifacts in the residual value recursively. At step 370, the process signals a scene change and marks a scene boundary in the input video stream when the normalized score is greater than a first threshold value. In one embodiment, an exemplary value of such a first threshold value is 0.65. At step 380, in some instances, steps 310 through 370 are repeated for each frame in the input video stream to divide the input video stream into ordered sequences of scenes.

[00042] FIG. 4 illustrates steps of a process 400 for determining a scene type for a given scene sequence. At step 410, the process determines a scene's position in the input video stream's timeline. Based on the scene's position, a score is assigned on a scale of, for example, 1-5. In an illustrative example, a score of 1 could indicate that the scene is at the start of the input video stream and a score of 5 could indicate that the scene is at the end of the input video stream.

[00043] At step 420, the process determines a play-time length of a scene sequence and assigns a commensurate score (for example, on a scale of 1-5). In an illustrative example, a score of 1 could mean a scene length of less than 10 seconds and a score of 5 could mean a scene of length greater than 50 seconds.

[00044] At step 430, the process performs motion estimation in a scene sequence and assigns a commensurate score (for example, on a scale of 1-5). For example, a score of 1 could mean a scene with little or no motion vectors and a score of 5 could mean a scene with large motion vectors across the scene. Motion Estimation (ME) is generally a technique used to explore temporal redundancy in video sequences during compression. Temporal redundancy arises from the fact that neighboring frames very often share similar pixel regions. Therefore the goal of Motion Estimation is to estimate the shifting of such similar regions (macro-block) across neighbor frames, thus enabling them to be differentially encoded. In block-based ME, the displacement of similar regions is represented by motion vectors, which are computed by a Block-Matching Algorithms.

[00045] At step 440, the process computes an effective difference between the given scene sequence and a previous scene sequence. Based on the effective difference, the process assigns a commensurate score (for example, on a scale of 1-5). In an illustrative example, a score of 1 may indicate little difference between the scenes and a score of 5 may indicate a difference greater than xyz threshold. In exemplary instances, the effective difference may be computed using the same scoring principles described above in reference to steps 310 through 370.

[00046] At step 450, the process determines a spectral data size of the scene sequence. Based on the spectral data size, a score is assigned on a scale of, for example, 1-5. In an illustrative example, a score of 1 may indicate a scene with low spectral data and a score of 5 may indicate a scene with high spectral data. In one embodiment, transform coding techniques convert video data to the frequency (or spectral) domain, where the frequency domain range of an image frame represents the spectral data size. A transform coder converts pixel data in an image frame into frequency coefficients. In the frequency domain, low frequency data has greater human perceptual importance than high frequency data.

[00047] At step 460, the process optionally (or compulsorily in some instances) performs a search for facial structures in a scene sequence using, for example, facial recognition software. Based on the search results, a score may be assigned, for example, on a scale of 1-5. Here, in an illustrative example, a score of 1 may indicate no recognized facial structures and a score of 5 may indicate that a scene has a high number of facial structures.

[00048] At step 470, the process performs optical character recognition (OCR) in the scene sequence to identify any textual information in the scene sequence. OCR helps differentiate between pictorial and textual content in an image file. OCR utilizes pattern recognition, artificial intelligence and computer vision to perform the differentiation. Based on the OCR analysis, the process assigns a commensurate score (for example, on a scale of 1-5). In an illustrative example, a score of 1 may indicate the absence of any textual content in the scene sequence and a score of 5 may indicate textual content constituting at least 30 percent of the scene sequence's content. i.e. film's credits.

[00049] At step 480, the process determines screenplay structural information associated with the scene. In at least some embodiments, the screenplay structural information is a relative attention parameter. A relative attention parameter approximates the relative amount of viewer attention that is to be expected for a given scene sequence. In some instances, the relative attention parameter approximates the relative amount of viewer attention to be expected for a given video segment that a given scene sequence is to be a part

of. Based on the analysis, the process assigns a commensurate score (for example, on a scale of 1-5). In an illustrative example, a score of 1 may indicate a low viewer interest in the scene sequence's content and a score of 5 may indicate a high viewer interest in the scene sequence.

[00050] At step 490, the process determines a scene type for the scene sequence based on scores from the steps 410 through 480. In one embodiment, the determination of scene type using the scores from steps 410 through 480 can be based on a waterfall process. A waterfall process is a linear, beginning-to-ending, sequential decision-making process, where the process generally does not revisit any intermediate conclusions it has reached along its path to a final decision.

[00051] In an illustrative example, a scene involving credits at the end of a movie will generally have text moving in the upward or downward direction. Such a scene sequence typically has small but constant motion vectors, scored at 2 and lower, pointing either upward or downward, depending on the direction of the text. Additionally, the scene sequence normally includes text in the form of movie credits, constituting more than, for example, 30% of the scene content. The optical character recognition process generally scores the scene at 4 or above. Given that movie credits are generally part of every movie and constitute a material part of the end movie timeline, the waterfall process first checks to see whether a scene is of the "scroll credits" type before performing checks for other scene types. In the example, the 2 scores strongly suggest that the scene type involved is of "scroll credits" type. and hence the scene type determination could be ended for the scene once the scene is tagged as such. If the scene type were determined not to be of "scroll credits" type, the waterfall process checks the scene sequence to see if it is of one of the scene types other than "scroll credits". Also, once the waterfall process had made a determination that a given scene is not of a particular type, the process generally never reevaluates the scene against that particular scene type.

[00052] In another illustrative example, a scene capturing the redwood trees generally involves the trees' green foliage and their surroundings. The foliage would typically constitute a major part of the scene's content. Such a scene would have little or random motion vectors, as the trees themselves stay static while their branches and leaves have minimal motions. The motion estimation score would be close to zero. Furthermore, any recognition of text in the scene would generally be a brief description of the scene, resulting in a low textual content score. The spectral analysis, however, would result in a high score, as the green in the scene's foliage will be captured in the high frequency domain of the spectral data. As discussed earlier, low frequency spectral data has greater human perceptual

importance than high frequency data, enabling encoding of images with high spectral data score at lower quality. Based on the scores, the waterfall process will determine that the scene sequence is of a "static scene" type, requiring high encoding quality at the first frame followed by low quality residual encoding and low deblocking filtering.

[00053] The above description illustrated processes for determining scene boundaries and scene types. For purposes of determining scene types, in at least some instances, it is useful to determine and analyze motion estimation in a scene in order to determine the magnitude of motion, represented by motion vectors, in a scene sequence. Figure 5 now illustrates an exemplary process of motion estimation in a scene sequence, as is explained in further detail here. Motion Estimation (ME) is generally a technique used to explore temporal redundancy in video sequences during compression. Temporal redundancy arises from the fact that neighboring frames very often share similar pixel regions. Therefore the goal of Motion Estimation is to estimate the shifting of such similar regions (macro-block) across neighbor frames, thus enabling them to be differentially encoded. In block-based ME, the displacement of similar regions is represented by motion vectors, which are computed by a Block-Matching Algorithms.

[00054] In one embodiment, the Block-Matching Algorithm (BMA) searches for similar blocks in an image frame and generates the motion vectors. BMA uses a fast-search approach, which looks only in specific points of the search window, while a similar block is being searched. In another approach, known as multi-resolution motion estimation, ME is performed hierarchically, computing motion vectors for a specific frame region, and refining them in each level. The ME works with different resolutions of one frame, successively refining the found motion vectors. Other strategies look into finding parallelism in BMAs, in order to run ME stages simultaneously.

[00055] FIG. 5 illustrates an exemplary approach for motion estimation. Here, in embodiments, images of macroblocks of one frame that are found in subsequent frames (for example, frames at different positions) are communicated by use of a motion vector. Figures 5.1 and 5.2 represent the reference frame and the desired frame respectively. The frames are divided into macroblocks, for example, in sizes ranging from 4x4 to 16x16. In embodiments, each macroblock in the reference frame is compared to the macroblocks in the desired frame to detect a match between any of the macroblocks. Figures 5.3 and 5.4 illustrate the reference frame and the desired frame, broken up into their respective macroblocks, which are compared against each other. Figure 5.5 represents a macroblock from the reference frame

that matches a macroblock in the desired frame, although the macroblocks are not in the same grid position in their respective frames. Figure 5.6 represents motion vectors that are generated by an encoder to communicate a position of the macroblock in the desired frame with respect to the macroblock's position in the reference frame. The motion estimation thus helps determine the motion vectors in a scene sequence, allowing the determination of scene type to be influenced by the magnitude of motion vectors in the scene sequence.

[00056] The above description illustrated processes for determining scene type based on motion estimation. For purposes of determining scene types, in addition to motion estimation, in at least some instances, it is useful to determine and analyze screenplay structural information associated with a scene. The screenplay structural information utilizes the general organization of a movie story line to determine the appropriate scene type, allowing for the proper encoding of a given scene.

[00057] A movie is generally based on a screenplay. The screenplay is structured so as to grab the audiences' attention. The first part of a movie's screenplay, called the "bite and switch" segment, is generally when most people decide whether to watch the entire movie or not. Therefore, the image quality here is expected to be very high in order to not compromise audience viewing experience. The next part of a movie's screenplay, called the "character development" segment, generally garners low audience attention and can therefore be of lower image quality than the previous segment. The subsequent segment of the movie generally constitutes the plot of the movie, where the audience attention is higher compared to the previous segment. The image quality has to be higher than the previous quality. The next segment of a movie is the "climax", which is the most important part of the movie and the quality of the images need to be high. The final segment is the "credits" of the movie, which garners very low audience attention. The segment can utilize lower quality images without affecting audience viewing experience.

[00058] In one embodiment, the screenplay structural information used to determine scene types could be based on a movie's time-line. For example, when a given scene sequence is part of the start of the movie, the scene sequence could be classified as a "bite and switch" scene, garnering high audience attention. The scene sequence could be scored as a 5 on the scale, indicating high audience interest. As a further example, when a given scene sequence is thirty minutes into the movie, it could be assumed that the movie segment involves character development. The character development segment gets low audience attention. Therefore, any scene sequence that is part of the character development could be

scored as a 2 or less on the scale. The time-line information, thus, helps determine the scene type.

[00059] In one embodiment, the screenplay structural information used to determine scene types could be a relative attention parameter, where the relative attention parameter approximates an estimated viewer interest to be expected for a given segment of the input video stream. The relative attention parameter could be predetermined by a viewer or could be based on input from the movie's director. The information could be included in the input video stream as part of the input video stream's metadata. By parsing the metadata, the relative attention parameter could be determined. The predetermined relative attention parameter could be defined for each given scene sequence in the input video stream or for a given segment of the input video stream that comprises a plurality of scene sequences. When the relative attention parameter indicates high audience attention, the score could be set at 4 or higher. When the relative attention parameter indicates low audience attention, the score could be set at 2 or lower. The relative attention parameter could thus be utilized to determine the scene type.

[00060] In one embodiment, the screenplay structural information used to determine scene types could be based on the textual content in the scene sequence or could be based on the closed captioning associated with the scene sequence. In both cases, the textual information is used to determine the movie's screenplay sequence. The screenplay sequence can then be utilized to determine the audience attention for the given scene, with a score of 1 for a scene of low interest and a score of 5 for a scene of high interest. The textual content information could thus be utilized to determine the scene type.

[00061] In another embodiment, the screen play structural information used to determine scene types could be based on the audio content associated with the scene sequence. The audio content could be, for example, loudness (amplitude) of the audio content, human speech, silence, language recognition, language differentiation, musical score, sound effects, surround sounds, etc. In an illustrative example, the loudness of the audio content could be used to determine the screenplay segment the scene sequence is part of. Action segments in a movie generally have loud audio content associated with them. The loud audio content is needed to get the audiences' full attention. Furthermore, action scenes typically involve special effects, such as explosions, that generate loud audio content. On the other hand, movie segments associated with character development generally involve dialogues at the normal range of human audible amplitude and little special effects such as explosions. The audience attention is typically low in the character development phase of the

movie. The audio content's loudness can thus be utilized to determine the audiences' attention for a given scene, with a score of 1 for a scene of low amplitude and a score of 5 for a scene of high amplitude. The amplitude (loudness) of the audio content could thus be utilized to determine the scene type based on the audiences' attention.

5 [00062] In another illustrative example, the sound effects associated with a scene sequence could be used to determine the screenplay segment the scene sequence is part of. Special sound effects such as an increasing tempo in the audio content are generally used to indicate a build up to an interesting twist in the movie, an exhilarating action sequence, etc. which garner high audience attention. On the other hand, little sound effects are associated
10 with movie segments involving conversations. The segments generally lack sound effects as conversations typically lack dramatic emotion shifts that can be further emphasized with the sound effects. The audio content's special effects can thus be utilized to determine the audiences' attention for a given scene, with a score of 1 for a scene of low sound effects and a score of 5 for a scene of rich sound effects. The sound effects of the audio content could thus
15 be utilized to determine the scene type based on the audiences' attention.

[00063] FIG. 6 is a block diagram of a processing system that can be used to implement any of the techniques described above, such as an encoder. Note that in certain embodiments, at least some of the components illustrated in FIG. 6 may be distributed between two or more physically separate but connected computing platforms or boxes. The
20 processing can represent a conventional server-class computer, PC, mobile communication device (e.g., smartphone), or any other known or conventional processing/communication device.

[00064] The processing system 601 shown in FIG. 6 includes one or more processors 610, i.e. a central processing unit (CPU), memory 620, at least one communication device
25 640 such as an Ethernet adapter and/or wireless communication subsystem (e.g., cellular, WiFi, Bluetooth or the like), and one or more I/O devices 670, 680, all coupled to each other through an interconnect 690.

[00065] The processor(s) 610 control(s) the operation of the computer system 601 and may be or include one or more programmable general-purpose or special-purpose
30 microprocessors, microcontrollers, application specific integrated circuits (ASICs), programmable logic devices (PLDs), or a combination of such devices. The interconnect 690 can include one or more buses, direct connections and/or other types of physical connections, and may include various bridges, controllers and/or adapters such as are well-known in the

art. The interconnect 690 further may include a "system bus", which may be connected through one or more adapters to one or more expansion buses, such as a form of Peripheral Component Interconnect (PCI) bus, HyperTransport or industry standard architecture (ISA) bus, small computer system interface (SCSI) bus, universal serial bus (USB), or Institute of
5 Electrical and Electronics Engineers (IEEE) standard 1394 bus (sometimes referred to as "Firewire").

[00066] The memory 620 may be or include one or more memory devices of one or more types, such as read-only memory (ROM), random access memory (RAM), flash memory, disk drives, etc. The network adapter 640 is a device suitable for enabling the
10 processing system 601 to communicate data with a remote processing system over a communication link, and may be, for example, a conventional telephone modem, a wireless modem, a Digital Subscriber Line (DSL) modem, a cable modem, a radio transceiver, a satellite transceiver, an Ethernet adapter, or the like. The I/O devices 670, 680 may include, for example, one or more devices such as: a pointing device such as a mouse, trackball,
15 joystick, touchpad, or the like; a keyboard; a microphone with speech recognition interface; audio speakers; a display device; etc. Note, however, that such I/O devices may be unnecessary in a system that operates exclusively as a server and provides no direct user interface, as is the case with the server in at least some embodiments. Other variations upon the illustrated set of components can be implemented in a manner consistent with the
20 invention.

[00067] Software and/or firmware 630 to program the processor(s) 610 to carry out actions described above may be stored in memory 620. In certain embodiments, such software or firmware may be initially provided to the computer system 601 by downloading it from a remote system through the computer system 601 (e.g., via network adapter 640).

[00068] The techniques introduced above can be implemented by, for example, programmable circuitry (e.g., one or more microprocessors) programmed with software and/or firmware, or entirely in special-purpose hardwired circuitry, or in a combination of such forms. Special-purpose hardwired circuitry may be in the form of, for example, one or more application-specific integrated circuits (ASICs), programmable logic devices (PLDs),
25 field-programmable gate arrays (FPGAs), etc.

[00069] Software or firmware for use in implementing the techniques introduced here may be stored on a machine-readable storage medium and may be executed by one or more general-purpose or special-purpose programmable microprocessors. A "machine-readable storage medium", as the term is used herein, includes any mechanism that can store

information in a form accessible by a machine (a machine may be, for example, a computer, network device, cellular phone, personal digital assistant (PDA), manufacturing tool, any device with one or more processors, etc.). For example, a machine-accessible storage medium includes recordable/non-recordable media (e.g., read-only memory (ROM); random access
5 memory (RAM); magnetic disk storage media; optical storage media; flash memory devices; etc.), etc.

[00070] The term "logic", as used herein, can include, for example, programmable circuitry programmed with specific software and/or firmware, special-purpose hardwired circuitry, or a combination thereof.

[00071] The foregoing description of various embodiments of the claimed subject matter has been provided for the purposes of illustration and description. It is not intended to be exhaustive or to limit the claimed subject matter to the precise forms disclosed. Many modifications and variations will be apparent to the practitioner skilled in the art. Embodiments were chosen and described in order to best describe the principles of the invention and its practical application, thereby enabling others skilled in the relevant art to understand the claimed subject matter, the various embodiments and with various modifications that are suited to the particular use contemplated.

[00072] The teachings of the invention provided herein can be applied to other systems, not necessarily the system described above. The elements and acts of the various embodiments described above can be combined to provide further embodiments.

[00073] While the above description describes certain embodiments of the invention, and describes the best mode contemplated, no matter how detailed the above appears in text, the invention can be practiced in many ways. Details of the system may vary considerably in its implementation details, while still being encompassed by the invention disclosed herein. As noted above, particular terminology used when describing certain features or aspects of the invention should not be taken to imply that the terminology is being redefined herein to be restricted to any specific characteristics, features, or aspects of the invention with which that terminology is associated. In general, the terms used in the following claims should not be construed to limit the invention to the specific embodiments disclosed in the specification, unless the above Detailed Description section explicitly defines such terms. Accordingly, the actual scope of the invention encompasses not only the disclosed embodiments, but also all equivalent ways of practicing or implementing the invention under the claims.

2012211249 09 Sep 2016

CLAIMS

What is claimed is:

1. A method for encoding a video stream using scene types each having a predefined set of one or more of a plurality of encoder parameters used by a video encoder to encode any given scene type, the method comprising:

receiving an input video stream;

dividing the input video stream into a plurality of scenes based on scene boundaries, each scene comprising a plurality of temporally contiguous image frames, wherein a given scene boundary is determined according to relatedness of two temporally contiguous image frames in the input video stream by determining the difference between two image frames that are temporarily contiguous to each other on the input video stream's timeline and when the difference exceeds a set threshold that signals a scene change, the two image frames are determined to be part of two different scene sequences and therefore a scene boundary is established between the two image frames;

analysing the scenes and assigning a scene type for each of the plurality of scenes; and encoding each of the plurality of scenes using encoding parameters according to the assigned scene type.

2. The method for encoding a video stream as recited in claim 1, wherein each scene type is assigned based on one or more of the following scene sequence parameters:

a given scene's position on the input video stream's timeline;

a length of the given scene;

a motion estimation in the given scene;

a effective difference in the given scene from a previous scene;

a spectral data size of the given scene;

a scene's textual content determined by optical character recognition in the given scene; or

a screenplay structure information of the given scene.

3. The method for encoding a video stream as recited in either of claims 1 or 2, wherein the determination of a scene type further comprises utilizing facial recognition.

4. The method of either of claims 2 or 3, wherein the screenplay structure information includes a relative attention parameter, wherein the relative attention parameter approximates a predetermined estimation of a relative amount of viewer attention to be expected for a segment of the input video stream that comprises the given scene.

2012211249 09 Sep 2016

5. The method of claim 2, wherein the screenplay structure information further includes one or more of:

- a time range definition;
- a textual information from the given scene;
- a audio content associated with the given scene;
- a close captioning information associated with the given scene; or
- a meta data associated with the given scene.

6. The method for encoding a video stream as recited any one of the preceding claims, wherein a given scene type includes one or more of:

- a fast motion;
- a static;
- a talking head;
- a text;
- a mostly black images;
- a short scenes;
- a scroll credits;
- a title scene;
- a miscellaneous; or
- a default.

7. The method for encoding a video stream as recited in any one of the preceding claims, further comprising:

determining that a first image frame is temporally contiguous to a second image frame when the first image frame has one adjacent position to the second image frame in the input video stream's timeline.

8. The method for encoding a video stream as recited in claim 1, wherein determining relatedness of two temporally contiguous image frames in the input video stream comprises:

- scaling one or more high frequency elements of each image frame;
- removing the one or more high frequency elements of each image frame;
- analyzing the image frames to determine a difference between temporally contiguous image frames, wherein a score is computed based on the difference; and

identifying a degree of unrelatedness between the image frames when the score exceeds a preset limit, wherein the preset limit score is at a threshold where a scene change occurs.

2012211249 09 Sep 2016

9. The method of claim 8, wherein the difference is tracked by one of a recursive filter or an adaptive filter.

10. The method for encoding a video stream as recited in claim 1, wherein the predetermined encoder parameters includes one or more of:

- a motion estimation range search;
- a deblocking amount factor;
- a quantizer; or
- a reference frame numbers.

11. A video encoding apparatus for encoding a video stream using scene types each having a predefined set of one or more of a plurality of encoder parameters used by the video encoder to encode any given scene type, the apparatus comprising:

an input module for receiving an input video stream;

a video processing module to divide the video stream into a plurality of scenes based on scene boundaries, each scene comprising a plurality of temporally contiguous image frames, wherein the video processing module determines a given scene boundary according to the relatedness of two temporally contiguous image frames in the input video stream by determining the difference between two image frames that are temporarily contiguous to each other on the input video stream's timeline and when the difference exceeds a set threshold that signals a scene change, the two image frames are determined to be part of two different scene sequences and therefore a scene boundary is established between the two image frames;

the video processing module to analyse and assign a scene type for each of the plurality of scenes; and

a video encoding module to encode each of the plurality of scenes using encoding parameters according to the assigned scene type.

12. The video encoding apparatus as recited in claim 11, wherein the video processing module assigns each scene type using one or more of the following scene sequencing parameters:

- a given scene's position on the input video stream's timeline;
- a length of the given scene;
- a motion estimation in the given scene;
- a effective difference in the given scene from a previous scene;
- a spectral data size of the given scene;
- a optical character recognition in the given scene; or
- a screenplay structure information of the given scene.

2012211249 09 Sep 2016

13. The video encoding apparatus as recited in claim 12, wherein the screenplay structure information utilized by the video encoding apparatus includes a relative attention parameter, wherein the relative attention parameter approximates a predetermined estimation of a relative amount of viewer attention to be expected for a segment of the input video stream that comprises the given scene.

14. The video encoding apparatus as recited in either of claims 12 or 13, wherein the screenplay structure information utilized by the video encoding apparatus further includes one or more of:

- a time range definition;
- a textual information from the given scene;
- a audio content associated with the given scene;
- a close captioning information associated with the given scene; or
- a meta data associated with the given scene.

15. The video encoding apparatus as recited in claim 11, wherein the video processing module utilizes facial recognition to determine scene type.

16. The video encoding apparatus as recited in any one of claims 11 to 15, wherein the scene type assigned by the video processing module includes one or more of:

- a fast motion;
- a static;
- a talking head;
- a text;
- a mostly black images;
- a short scenes;
- a scroll credits;
- a title scene;
- a miscellaneous; or
- a default.

17. The video encoding apparatus as recited in any one of claims 11 to 16, wherein the video processing module further comprises:

determining that a first image frame is temporally contiguous to a second image frame when the first image frame has at least one adjacent position to the second image frame in the input video stream's timeline.

2012211249 09 Sep 2016

18. The video encoding apparatus as recited in any one of claims 11 to 17, wherein the video processing module's determination of relatedness of two temporally contiguous image frames in the input video stream comprises:

scaling one or more high frequency elements of each image frame;

removing the one or more high frequency elements of each image frame;

analyzing the image frames to determine a difference between temporally contiguous image frames, wherein a score is computed based on the difference; and

identifying a degree of unrelatedness between the image frames when the score exceeds a preset limit, wherein the preset limit score is at a threshold where a scene change occurs.

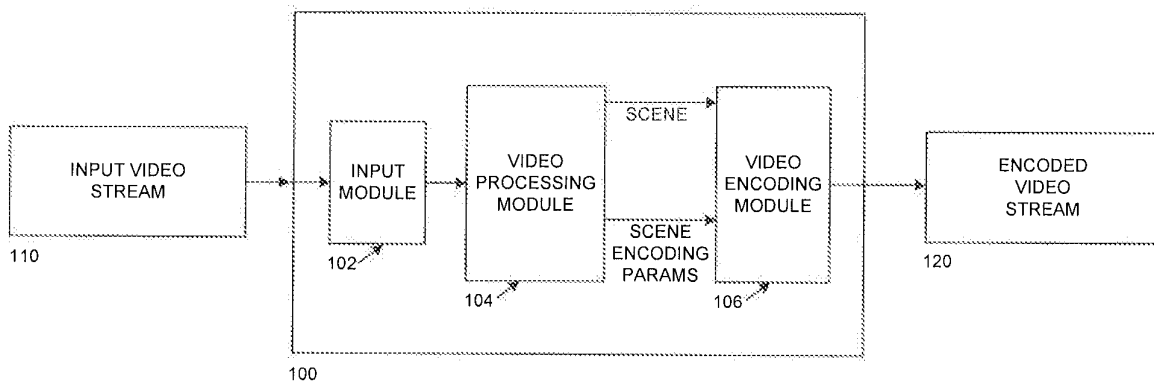
19. The video encoding apparatus as recited in any one of claims 11 to 18, wherein the video processing module utilizes one of a recursive filter or an adaptive filter to track the differences.

20. The video encoding apparatus as recited in any one of claims 11 to 18, wherein the predetermined encoder parameters utilized by video encoding module includes one or more of:

a motion estimation range search;

a quantizer; or

a reference frame numbers.



BLOCK DIAGRAM OF VIDEO ENCODING APPARATUS

FIG. 1

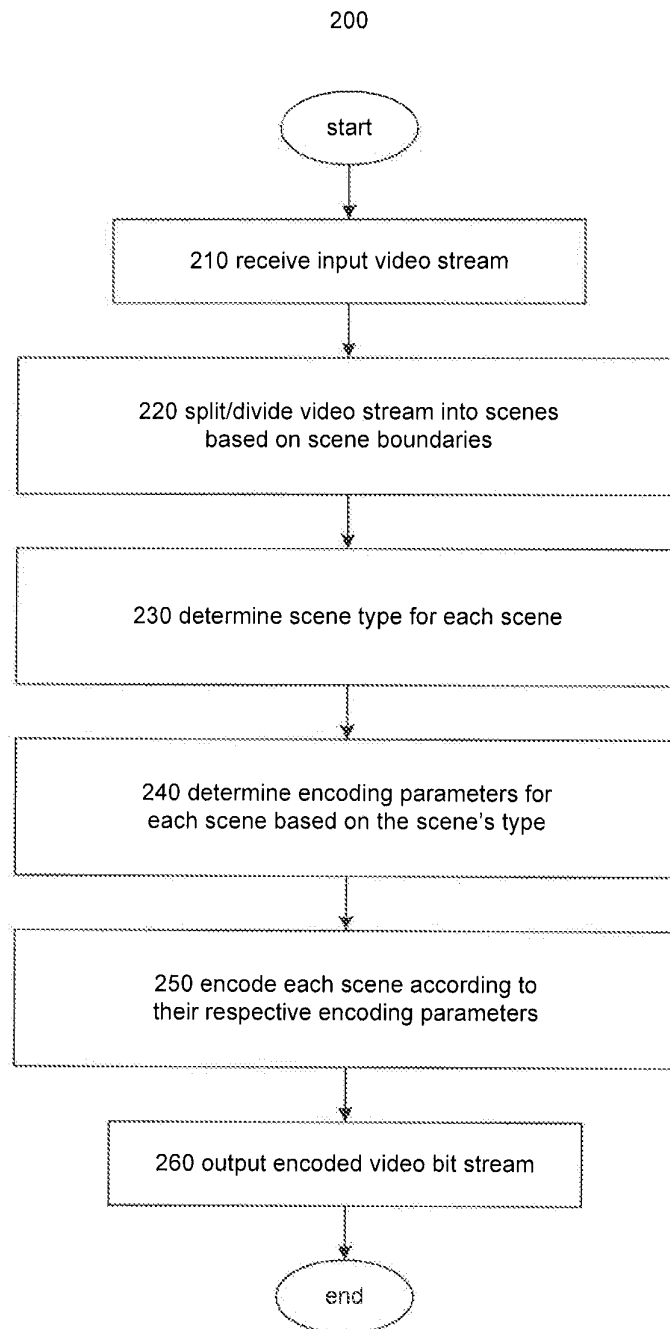
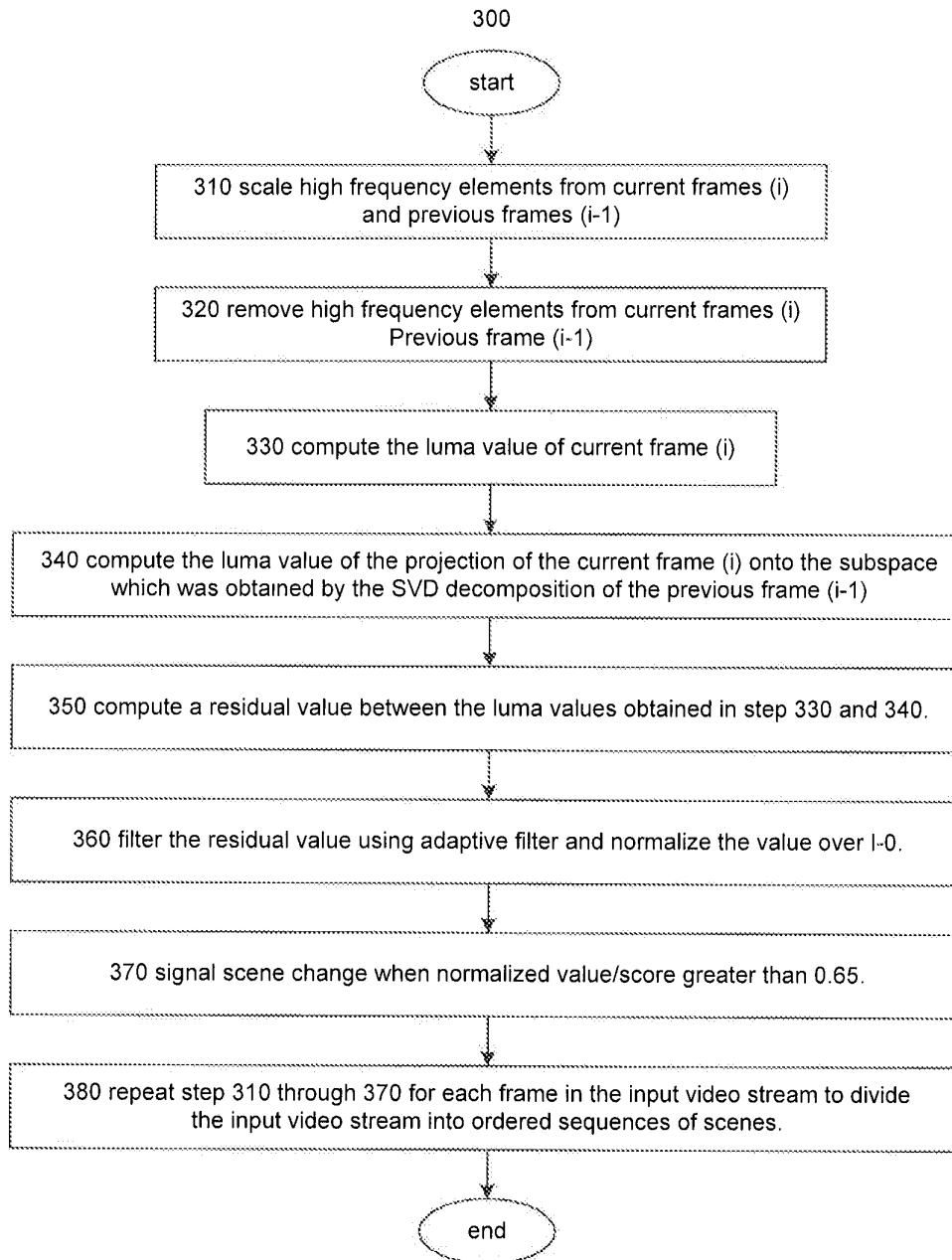
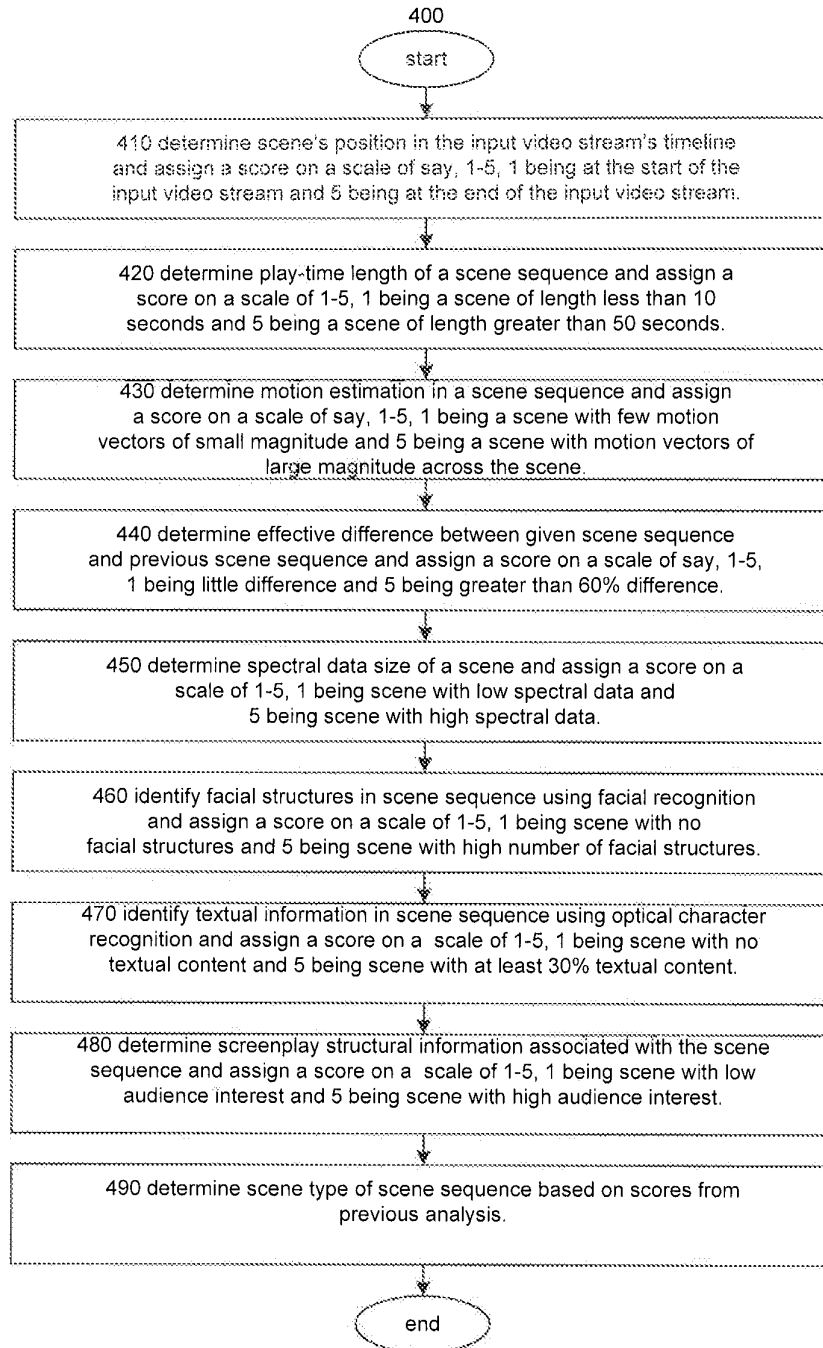
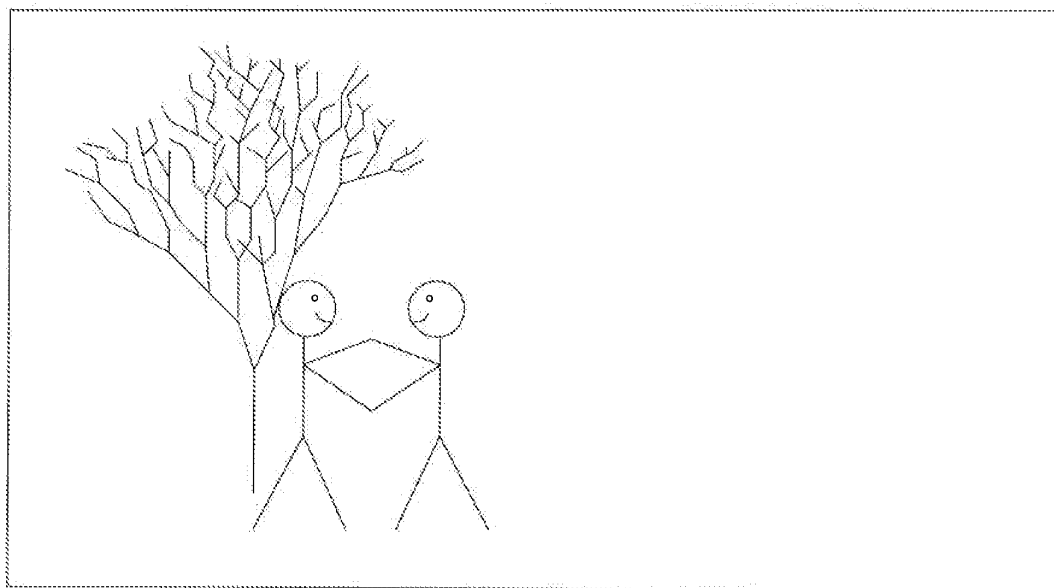


FIG. 2

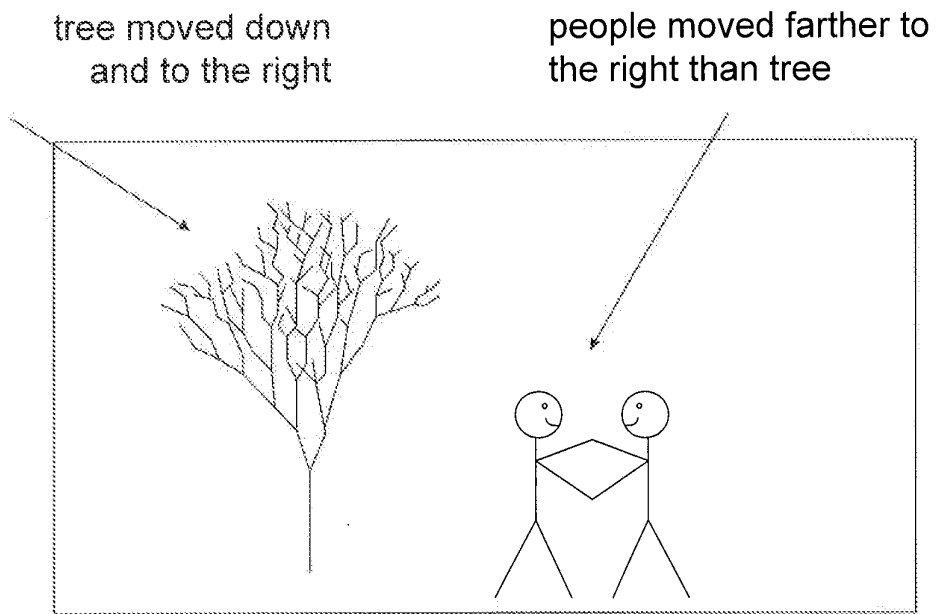
**FIG. 3**

**FIG. 4**



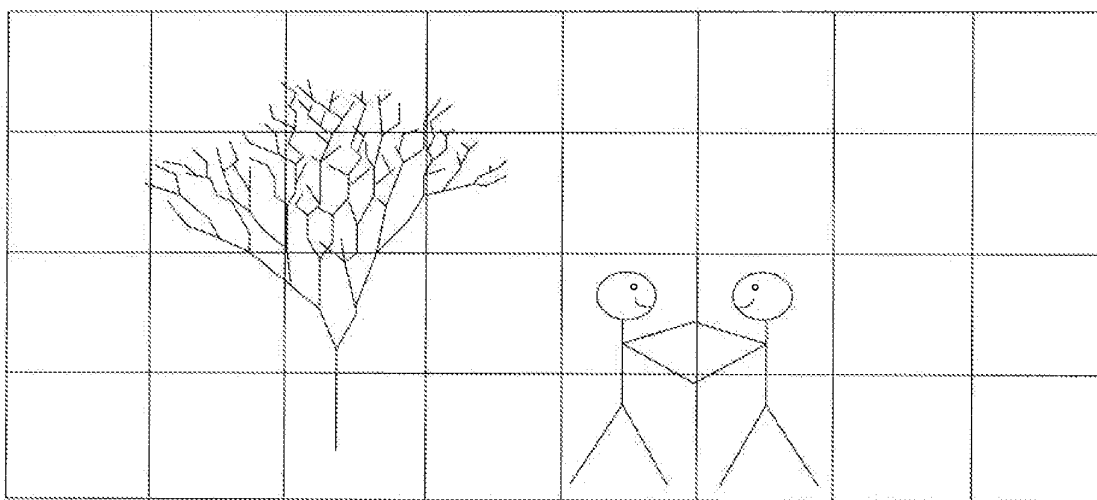
FRAME 1

FIG. 5.1



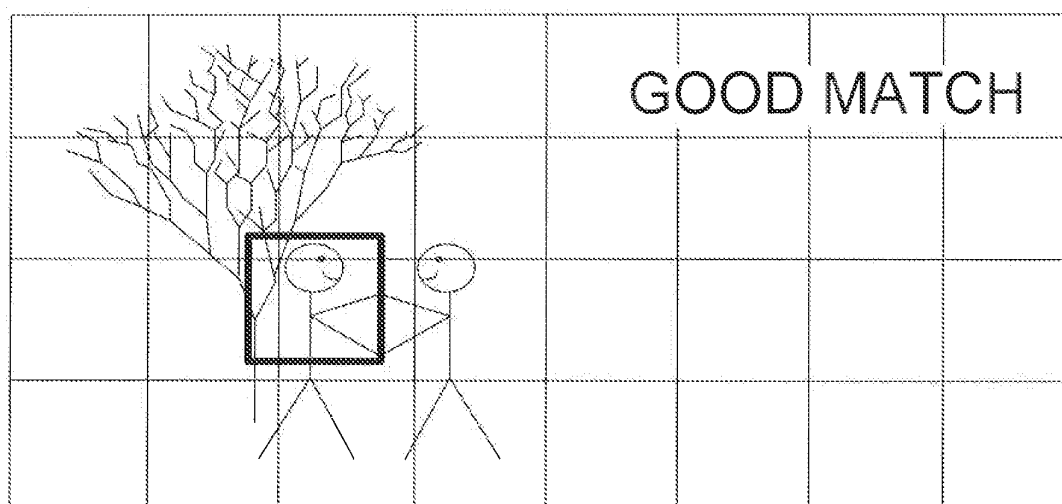
FRAME 2

FIG. 5.2



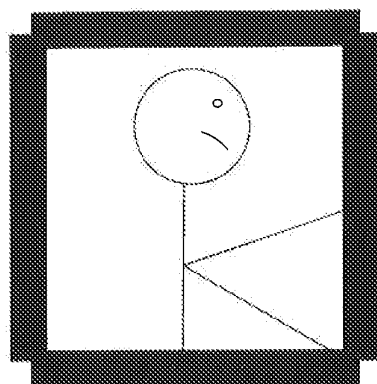
Desired Picture

FIG. 5.3



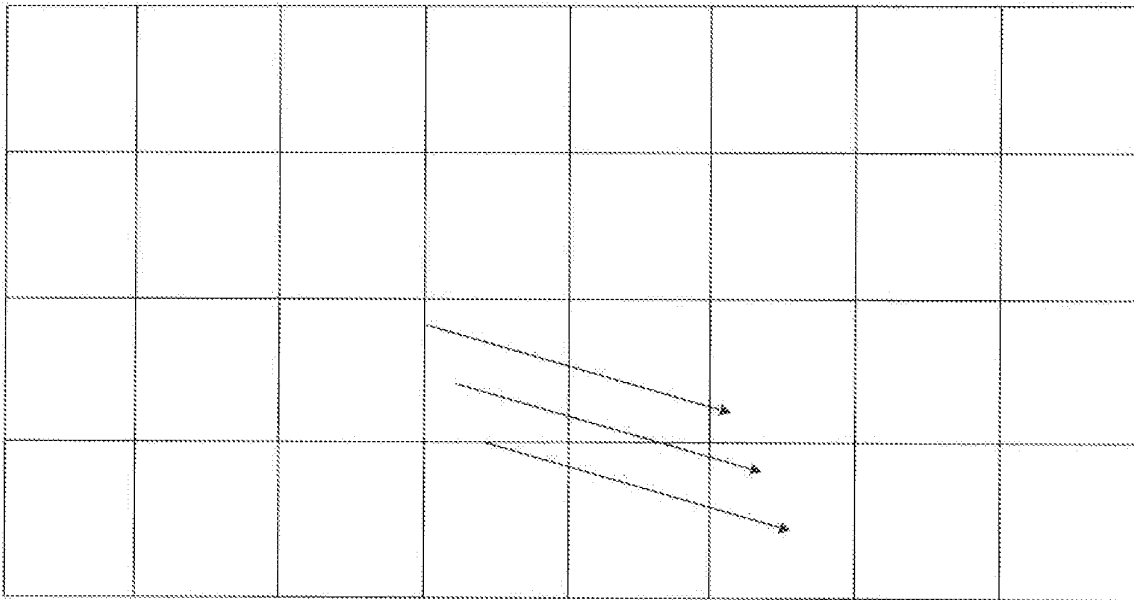
Motion Estimation Macroblock Example

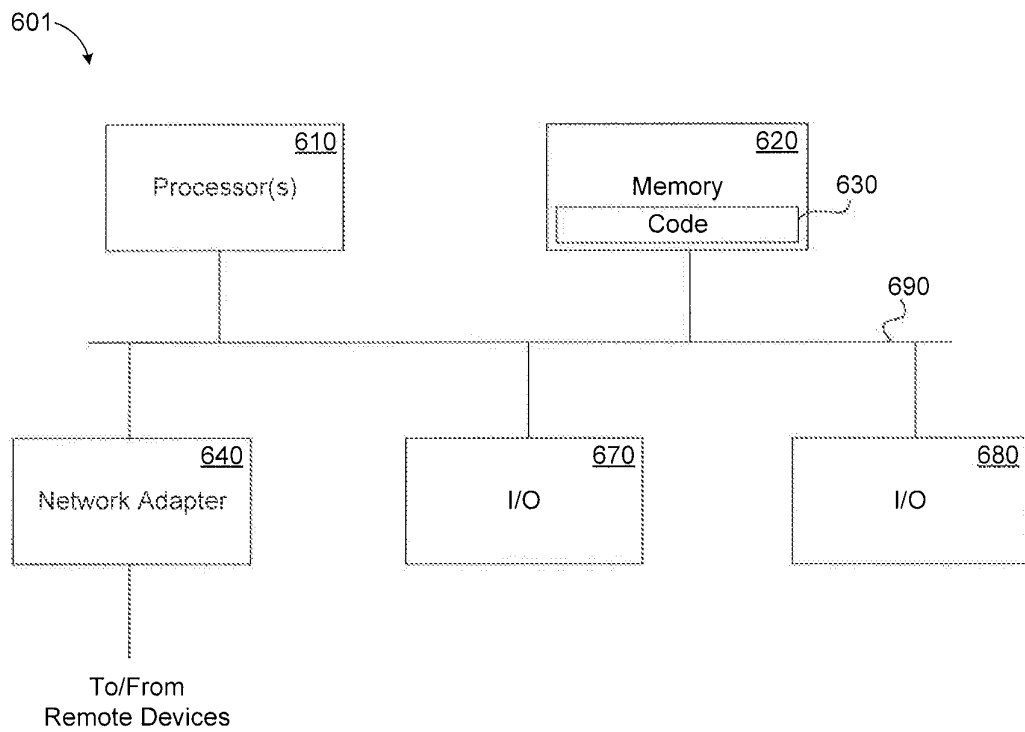
FIG. 5.4



Macroblock to be coded

FIG. 5.5

**FIG. 5.6**

**FIG. 6**