



(51) International Patent Classification:

H04N 21/44 (2011.01) *H04N 7/15* (2006.01)
H04N 21/43 (2011.01)

(21) International Application Number:

PCT/US2011/035494

(22) International Filing Date:

6 May 2011 (06.05.2011)

(25) Filing Language:

English

(26) Publication Language:

English

(71) Applicant (for all designated States except US):
GOOGLE INC. [US/US]; 1600 Amphitheatre Parkway,
Mountain View, CA 94043 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **FLODMAN, Mag-
nus** [SE/SE]; c/o Google Inc., 1600 Amphitheatre Park-
way, Mountain View, CA 94043 (US). **WESTIN, Patrik**
[SE/SE]; c/o Google Inc., 1600 Amphitheatre Parkway,
Mountain View, CA 94043 (US).

(74) Agents: **BASILE, JR., Andrew R.** et al.; Young Basile
Hanlon & Macfarlane, P.c., 3001 West Big Beaver Road,
Suite 624, Troy, MI 48084 (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, RO, RS, RU, 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, 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))

(54) Title: APPARATUS AND METHOD FOR RENDERING VIDEO USING POST-DECODING BUFFER

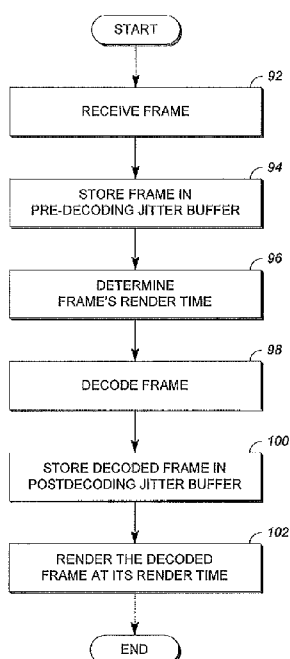


FIG. 5

(57) Abstract: A system, apparatus, and method for rendering a video stream having a plurality of frames is disclosed. The embodiments include receiving at least some of the plurality of frames, determining a render time for at least some of the received frames before decoding based on an estimated arrival time and a delay, decoding at least some of the received frames in advance of the render time, storing the decoded frames in a post-decoding jitter buffer located in a memory, and rendering at least some of the stored decoded frames from the post-decoding jitter buffer at their render times.

APPARATUS AND METHOD FOR RENDERING VIDEO USING POST-DECODING BUFFER

TECHNICAL FIELD

[0001] The present invention relates in general to video encoding, decoding, transmission and rendering.

BACKGROUND

[0002] An increasing number of applications today make use of digital video for various purposes including, for example, remote business meetings via video conferencing, high definition video entertainment, video advertisements, and sharing of user-generated videos. As technology is evolving, users have higher expectations for video quality and expect high resolution video even when transmitted over communications channels having limited bandwidth. One type of video transmission includes real-time encoding and transmission, in which the receiver of a video stream decodes and renders frames as they are received.

[0003] To permit transmission of digital video streams while limiting bandwidth consumption, a number of video compression schemes have been devised, including formats such as VPx, promulgated by Google, Inc. of Mountain View, California, and H.264, a standard promulgated by ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG), including present and future versions thereof. H.264 is also known as MPEG-4 Part 10 or MPEG-4 AVC (formally, ISO/IEC 14496-10).

SUMMARY

[0004] Disclosed herein are embodiments of methods and apparatuses for encoding a video signal.

[0005] One aspect of the disclosed embodiments is a method for rendering a video stream having a plurality of frames. The method includes receiving at least some of the plurality of frames, determining a render time for at least some of the received frames before decoding based on an estimated arrival time and a delay, decoding at least some of the received frames in advance of the render time, storing the decoded frames in a post-decoding jitter buffer located in

a memory, and rendering at least some of the stored decoded frames from the post-decoding jitter buffer at their render times.

[0006] The method can also include storing the received frames in a pre-decoding jitter buffer until they are decoded.

[0007] The method can also include selecting a frame from the post-decoding jitter buffer that has a render time that is closest to and comes before a current time plus a render delay, and rendering the selected frame.

[0008] The method can also include removing the selected frame and frames having a render time before the selected frame from the post-decoding jitter buffer, or removing frames from the post-decoding jitter buffer having a render time before the selected frame after rendering the selected frame.

[0009] The method can also include determining a jitter delay, and determining the render time as the estimated arrival time plus the jitter delay.

[0010] The method can also include a plurality of received video streams that includes the video stream and creating a conference mix by rendering at least some frames of at least some of the plurality of received video streams into the conference mix.

[0011] The method can also include video streams having at least two different frame rates.

[0012] In the method, each video stream can be received from one of a plurality of stations, and the created conference mix can be sent to at least one of the plurality of stations.

[0013] The method can also include creating a different conference mix for at least two stations of the plurality of stations.

[0014] Another aspect of the disclosed embodiments is an apparatus for rendering a current frame from a video stream's plurality of frames. The apparatus comprises a memory and at least one processor configured to execute instructions stored in the memory to receive at least some of the plurality of frames, determine a render time for at least some of the received frames based on an estimated arrival time, decode at least some of the received frames, store the decoded frames in a post-decoding jitter buffer, and render at least some of the stored decoded frames from the post-decoding jitter buffer at their render times.

[0015] The apparatus can include instructions to store the received frames in a pre-decoding jitter buffer, and decode a stored received frame and remove it from the pre-decoding jitter buffer when its render time is determined

[0016] The apparatus can include instructions to select a frame from the post-decoding jitter buffer that has a render time that is closest to and comes before a current time plus a render delay, and render the selected frame.

[0017] The apparatus can include instructions to remove the selected frame and frames having a render time before the selected frame from the post-decoding jitter buffer, or remove frames from the post-decoding jitter buffer having a render time before the selected frame after rendering the selected frame.

[0018] The apparatus can include instructions to create a conference mix by rendering at least some frames of at least some of the plurality of received video streams into the conference mix.

[0019] These and other embodiments, including combinations and variations of these embodiments, will be described in additional detail hereafter.

BRIEF DESCRIPTION OF THE DRAWINGS

[0020] The description herein makes reference to the accompanying drawings wherein like reference numerals refer to like parts throughout the several views, and wherein:

[0021] FIG. 1 is a schematic of a video encoding and decoding system;

[0022] FIG. 2 is a diagram of a video stream;

[0023] FIG. 3 is a timeline of transmitting and rendering a frame of a video stream in the video encoding and decoding system of FIG. 1;

[0024] FIG. 4 is a diagram of rendering a video stream in the video encoding and decoding system of FIG. 1;

[0025] FIG. 5 is a flowchart of a method of rendering a frame in a video stream in the video encoding and decoding system of FIG. 1;

[0026] FIG. 6 is a timeline illustrating the selection of a frame for rendering from a post-decoding jitter buffer of FIG. 4;

[0027] FIG. 7 is a diagram of a conference mix; and

[0028] FIG. 8 is a timeline illustrating the selection of frames for rendering into a conference mix from one or more post-decoding jitter buffers.

DETAILED DESCRIPTION

[0029] FIG. 1 is a diagram of an encoder and decoder system 10 for still or dynamic video images. An exemplary transmitting station 12 may be, for example, a computer having an internal configuration of hardware including a processor such as a central processing unit (CPU) 14 and a memory 16. CPU 14 is a controller for controlling the operations of transmitting station 12. CPU 14 is connected to memory 16 by, for example, a memory bus. Memory 16 may be random access memory (RAM) or any other suitable memory device. Memory 16 stores data and program instructions that are used by CPU 14. Other suitable implementations of transmitting station 12 are possible.

[0030] A network 28 connects transmitting station 12 and a receiving station 30 for transmission of an encoded video stream. Specifically, the video stream can be encoded by an encoder in transmitting station 12 and the encoded video stream can be decoded by a decoder in receiving station 30. Network 28 may, for example, be the Internet, which is a packet-switched network. Network 28 may also be a local area network (LAN), wide area network (WAN), virtual private network (VPN), or any other means of transferring the video stream from transmitting station 12.

[0031] The transmission of the encoded video stream can be accomplished using a real-time protocol, such as the real-time transport protocol (RTP) standard as promulgated by the Internet Engineering Task Force (IETF). Control of the transmission can be accomplished using the real-time transport control protocol (RTCP). For example, the RTCP can allow a receiving station to determine information about sent frames, including the time those frames were sent by a transmitting station. This information can be used to determine an estimated arrival time for frames.

[0032] In some circumstances, the rendered video stream can include small gaps in the video stream ("jitter") if there are portions of the video stream that are delayed in transmission. Jitter can cause the rendered video stream to freeze momentarily, reducing the quality of rendering.

[0033] Receiving station 30, in one example, may be a computer having an internal configuration of hardware including a processor such as a central processing unit (CPU) 32 and a memory 34. CPU 32 is a controller for controlling the operations of receiving station 30. CPU 32 can be connected to memory 34 by, for example, a memory bus. Memory 34 may be RAM or any other suitable memory device. Memory 34 stores data and program instructions that are used by CPU 32. Other suitable implementations of receiving station 30 are possible.

[0034] A display 36 configured to display a video stream can be connected to receiving station 30. Display 36 may be implemented in various ways, including by a liquid crystal display (LCD) or a cathode-ray tube (CRT). The display 36 can be configured to display a rendering of the video stream decoded by the decoder in receiving station 30.

[0035] Other implementations of the encoder and decoder system 10 are possible. In one implementation, additional components may be added to the encoder and decoder system 10. For example, a display or a video camera may be attached to transmitting station 12 to capture the video stream to be encoded. In another implementation, a transport protocol other than RTP may be used. Real-time encoding, transmission, decoding and rendering can result in a rendered video stream (i.e. on display 36) that includes gaps in the video stream if there are portions of the original video stream that are lost or delayed in transmission. At least some gaps can be avoided by rendering at a render time calculated by adding a delay to the actual receive time of video stream frames. However, variations in the arrival time of frames can be caused by, for example, changes in network transmission time (network jitter), changes in the amount of time to encode the frames, and clock drift between the transmitting station and the receiving station. These variations can cause a video stream to be rendered at frame rates different than at which the video stream was encoded or captured.

[0036] To render the video stream while minimizing such variations, the render time can be based on an estimated arrival time instead of the actual arrival time. The estimated arrival time is determined using a filter designed to filter out the variations. The estimated arrival time can determine an actual render time at which frames are rendered, instead of rendering frames out of a buffer at a constant frame rate and adjusting for variations such as clock drift by manipulating the buffer (i.e. by dropping frames).

[0037] FIG. 2 is a diagram a typical video stream 50 to be encoded and decoded. Video coding formats, for example, VP8 or H.264, provide a defined hierarchy of layers for video

stream 50. Video stream 50 includes a video sequence 52. At the next level, video sequence 52 consists of a number of adjacent frames 54, which can then be further subdivided into a single frame 56. At the next level, frame 56 can be divided into a series of blocks 58, which can contain data corresponding to, for example, a 16x16 block of displayed pixels in frame 56. Each block can contain luminance and chrominance data for the corresponding pixels. Blocks 58 can also be of any other suitable size such as 16x8 pixel groups or 8x16 pixel groups.

[0038] FIG. 3 is a timeline 60 of transmitting and rendering a frame n (current frame) of a video stream in the video encoding and decoding system of FIG. 1. The frame has a timestamp $T_S(n)$ 62 assigned by a transmitting station. $T_S(n)$ 62 is generated using the transmitting station's clock. The timestamp can be a send time of when the frame is sent to the receiving station by the transmitting station or a capture time of when the frame is captured by a capture device, such as a video camera. The frame can be transmitted with the timestamp. The frame has an estimated arrival time $\hat{T}_A(n)$ 64 of when the frame is expected to arrive at the receiving station. The estimated arrival time $\hat{T}_A(n)$ 64 is determined using the receiving station's clock and is described further later.

[0039] The frame will be rendered at the render time $T_R(n)$ 66 on the receiving station. The frame may be rendered at the render time $T_R(n)$ 66 even if the frame is not actually rendered at that time. The render time $T_R(n)$ 66 is a target time for rendering and the time that the rendering actually occurs at may vary from the render time. The difference between these values can be, for example, due to high resource utilization of the receiving station's resources. Other factors may also contribute to the difference. The render time $T_R(n)$ 66 for frame n can be defined using this formula:

$$T_R(n) = \hat{T}_A(n) + D_J(n) ; \text{ wherein} \quad (1)$$

$\hat{T}_A(n)$ is the estimated arrival time for frame n ; and

$D_J(n)$ is a jitter delay.

[0040] The time interval between the timestamp $T_S(n)$ 62 and the estimated arrival time $\hat{T}_A(n)$ 64 is the estimated one-way time offset $\Delta(n)$ 68. The estimated one-way time offset $\Delta(n)$ 68 includes a one-way transmission time for the frame to transit the network 28 and a clock offset, which is the difference between the transmitting station and receiving station clocks. The time interval between the estimated arrival time 64 and the render time $T_R(n)$ 66 is the delay

$D_J(n)$ 69. The delay $D_J(n)$ 69 is determined and used by the decoder to account for at least some expected variations in the actual arrival time of the frame. Delay $D_J(n)$ 69 is, for example, a jitter delay. Jitter is the variation of actual one-way time offset across multiple frames in a video stream. A jitter delay can be determined such that the actual arrival time of a frame generally comes before the determined render time of that frame. In other words, a jitter delay can be used to filter out jitter when rendering the video stream. Delay $D_J(n)$ 69 may alternatively include other delays in other implementations.

[0041] The actual arrival time of the frame can be any time after timestamp $T_S(n)$ 62. Generally, the delay 69 will be of a time interval where the actual arrival time of frames will be earlier than the render time $T_R(n)$ 66. In this case, the frame will be rendered at its render time, and gaps in the rendered video stream will be avoided. However, in certain circumstances, it may be advantageous for the actual arrival time to be after the render time. For example, in the case of severe network congestion where frames are delayed for a significant period of time, allowing video jitter or gaps may be advantageous as compared to introducing a significant delay resulting in a significantly later render time.

[0042] FIG. 4 is a diagram 70 of rendering a video stream in the video encoding and decoding system of FIG. 1. When a frame is received by a receiving station, it is placed in a pre-decoding jitter buffer 72. Due to the nature of packet-based network transmission a frame may be received in pieces. The pieces may be stored in the pre-decoding jitter buffer 72 until the entire frame is received. Once a render time is determined for the received frame, it is decoded using decoder 74. The decoded frame is then stored in the post-decoding jitter buffer 76. The decoded frame can then be selected from the post-decoding jitter buffer 76 to be rendered using a graphics card 78. The graphics card 78 can have a render delay attributed to the time necessary to render a frame from memory to a display.

[0043] The post-decoding buffer 76 above is not merely filled and emptied in a first-in-first-out (FIFO) fashion. Rather, frames are taken from the buffer based on the determined render time for each frame. The determined render time is based on an estimated arrival time of each frame. Further, frames are decoded in advance of the render time, meaning that frames are decoded without waiting until their render times. The decoding process may take different amounts of time depending on the resource utilization of the receiving station. Decoding in advance of the render time (as opposed to decoding as a part of the rendering process) provides

for more consistent rendering by eliminating the variability of decoder delay from the rendering process and instead accounting for network jitter and decoder delay at the post-decoding buffer 76.

[0044] The stages of FIG. 4 are illustrative in nature, and stages may be added, omitted, or changed in various implementations. For example, the pre-decoding jitter buffer 72 may be omitted. In another example, there may be multiple pre-decoding buffer 72 or multiple post-decoding buffer 76 in an implementation.

[0045] FIG. 5 is a flowchart of a method 90 of rendering a frame in a video stream in the video encoding and decoding system of FIG. 1. At stage 92, a frame is received by the receiving station. As described previously, the frame may be received all at once, or in pieces. At stage 94, the frame or portions thereof are stored in a pre-decoding jitter buffer until the reception of the frame is complete. Once the frame is complete, a render time for the frame is determined at stage 96. The render time can be determined based on an estimated arrival time for the frame and a determined delay, for example, as described above with respect to FIG. 3. However, the render time may be determined in other ways in alternate implementations.

[0046] Once the render time is determined for the frame, the frame is decoded at stage 98. The decoded frame is stored in a post-decoding jitter buffer at stage 100. At stage 102, the frame can be rendered at the render time or be discarded. Whether the frame is rendered is dependant on the other decoded frames included in the post-decoding jitter buffer, the render time, the current time, and a render delay. The determination of whether to render a frame is described in more detail with respect to FIG. 6. Once the frame is rendered or discarded, the method 90 ends.

[0047] FIG. 6 is a timeline 110 illustrating the selection of a frame for rendering from the post-decoding jitter buffer. The timeline shows a current time 112 and a projected rendering time 114 based on a render delay 113. The projected rendering time 114 indicates when a frame will be rendered to a display if the rendering of a frame is initiated at the current time 112. The timeline 110 includes frames 116, including individual frames 116a-116e spaced at intervals.

[0048] If a frame is required for rendering at the current time 112, the frame in the post-decoding buffer that is both before the projected rendering time 114 and the closest to the projected rendering time 114 (as compared to other frames in the post-decoding jitter buffer) will be selected. Once a frame is rendered, the rendered frame along with all frames in the post-

decoding jitter buffer having a render time before the rendered frame will be removed from the post-decoding jitter buffer. However, in an alternate implementation, the rendered frame may be maintained in the post-decoding jitter buffer after rendering. If no frames are available in the post-decoding jitter buffer for rendering, the receiving station will wait until a frame is available for rendering. During this time, the previously rendered frame can be maintained on the display.

[0049] In a first example, each of frames 116 are located in the post-decoding jitter buffer. Frames 116d and 116e each have a render time that is after the projected rendering time 114, and will not be selected for rendering. Of the frames having render times before the projected rendering time 114, frame 116c is the closest to the projected rendering time 114 and will be selected for rendering. After frame 116c is rendered, frames 116a and 116b will be removed from the post-decoding jitter buffer. Depending on the implementation, frame 116c may be removed as well.

[0050] In a second example, only frame 116d is in the post-decoding jitter buffer. Frame 116c is in the process of decoding, and frames 116a and 116b have already been rendered and removed from the post-decoding jitter buffer. Since no frames are available in the post-decoding jitter buffer with a render time before the projected rendering time 114, no frame will be selected for rendering, and the last rendered frame (116b) will be maintained on the display. The next frame selected for rendering will either be frame 116c or 116d, depending on whether frame 116c is decoded and stored in the post-decoding jitter buffer before the render time of frame 116d comes before the projected rendering time 114.

[0051] In a third example, the render delay 113 is increased such that projected rendering time 114 comes after frame 116d but before frame 116e. In this example, the render delay 113 is greater than the time interval between frames of the video stream. The methodology of selecting frames for rendering will, in effect, result in the selection of every other frame for rendering (and the dropping of other frames) because of the long render delay 113. In this example, the last frame having a render time within the render delay 113 (frame 116d) would be selected for rendering.

[0052] FIG. 7 is a diagram of a conference mix 120. The conference mix is a collection of one or more video streams, and can be used in multi-party video conferencing applications. The conference mix enables viewing of video streams from multiple parties (stations) at once. The conference mix can be created on a receiving station using a video stream from each of

several transmitting stations. Once the conference mix is created, it can be used by the receiving station and also transmitted back to the receiving stations for use by the receiving stations. The conference mix 120 can include a video stream from each transmitting and receiving station, or may alternatively include a subset of video streams. For example, the conference mix for a station may omit the video stream transmitted by that station.

[0053] The conference mix 120 can include a matrix of frames from the video streams, such as the one shown. The conference mix 120 shown can include up to four video streams, although other combinations are possible. Shown are a first video stream 122, a second video stream 124, and a third video stream 126.

[0054] FIG. 8 is a timeline 130 illustrating the selection of frames for rendering into a conference mix from one or more post-decoding jitter buffers. The timeline 130 illustrates frames from first video stream 122, second video stream 124, and third video stream 126. The timeline 130 includes a current time 132 and a projected rendering time 134 based on a render delay 133

[0055] Each video stream is shown having different frame rates (i.e. the time interval between successive frames or the number of frames per period). However, two or more of the video streams in a conference mix may alternatively have the same frame rates. The video streams may each have their own buffer(s), or in an alternative implementation, one or more video streams may share one or more buffers.

[0056] In a first example, all of the frames are located in one or more post-decoding jitter buffers. As with the single video stream application, a frame will be selected from each video stream that is both before the projected rendering time 134 and closest to the projected rendering time 134 of all of the frames from that video stream in the one or more post-decoding jitter buffers. In this example, the selected frames will include frame 136c, 138d, and 140e.

[0057] In a second example, all of the frames before the current time have either been rendered or discarded. The remaining frames are in the post-decoding jitter buffer. In this case, two frames meet the criteria for selection, frame 138d and 140e. However, no frames from video stream 122 meet the criteria. In this case, the existing rendered frame from video stream 122 will be maintained in the conference mix. Alternatively, the last previously rendered frame from each video stream can be maintained in the post-decoding jitter buffer. In this alternate

implementation, frame 136c would still be in the post-decoding jitter buffer and would be selected and rendered.

[0058] The operations of encoding and decoding can be performed in many different ways and can produce a variety of encoded data formats. The above-described embodiments of encoding or decoding may illustrate some exemplary encoding techniques. However, in general, encoding and decoding are understood to include any transformation or any other change of data whatsoever.

[0059] The embodiments of transmitting station 12 and/or receiving station 30 (and the algorithms, methods, instructions etc. stored thereon and/or executed thereby) can be realized in hardware, software, or any combination thereof including, for example, IP cores, ASICS, programmable logic arrays, optical processors, programmable logic controllers, microcode, firmware, microcontrollers, servers, microprocessors, digital signal processors or any other suitable circuit. In the claims, the term “processor” should be understood as encompassing any the foregoing, either singly or in combination. The terms “signal” and “data” are used interchangeably. Further, portions of transmitting station 12 and receiving station 30 do not necessarily have to be implemented in the same manner.

[0060] Further, in one embodiment, for example, transmitting station 12 or receiving station 30 can be implemented using a general purpose computer/processor with a computer program that, when executed, carries out any of the respective methods, algorithms and/or instructions described herein. In addition or alternatively, for example, a special purpose computer/processor can be utilized which can contain specialized hardware for carrying out any of the methods, algorithms, or instructions described herein.

[0061] Transmitting station 12 and receiving station 30 can, for example, be implemented on computers in a screencasting or a videoconferencing system. Alternatively, transmitting station 12 can be implemented on a server and receiving station 30 can be implemented on a device separate from the server, such as a hand-held communications device (i.e. a cell phone). In this instance, transmitting station 12 can encode content using an encoder into an encoded video signal and transmit the encoded video signal to the communications device. In turn, the communications device can then decode the encoded video signal using a decoder. Alternatively, the communications device can decode content stored locally on the communications device (i.e. no transmission is necessary). Other suitable transmitting station 12

and receiving station 30 implementation schemes are available. For example, receiving station 30 can be a personal computer rather than a portable communications device.

[0062] Further, all or a portion of embodiments of the present invention can take the form of a computer program product accessible from, for example, a computer-usable or computer-readable medium. A computer-usable or computer-readable medium can be any device that can, for example, tangibly contain, store, communicate, or transport the program for use by or in connection with any processor. The medium can be, for example, an electronic, magnetic, optical, electromagnetic, or a semiconductor device. Other suitable mediums are also available.

[0063] The above-described embodiments have been described in order to allow easy understanding of the present invention and do not limit the present invention. On the contrary, the invention is intended to cover various modifications and equivalent arrangements included within the scope of the appended claims, which scope is to be accorded the broadest interpretation so as to encompass all such modifications and equivalent structure as is permitted under the law.

What is claimed is:

1. A method for rendering a video stream, the video stream having a plurality of frames, the method comprising:
 - receiving at least some of the plurality of frames;
 - determining a render time for at least some of the received frames before decoding based on an estimated arrival time and a delay;
 - decoding at least some of the received frames in advance of the render time;
 - storing the decoded frames in a post-decoding jitter buffer located in a memory;and
 - rendering at least some of the stored decoded frames from the post-decoding jitter buffer at their render times.
2. The method of claim 1, further comprising:
 - storing the received frames in a pre-decoding jitter buffer until they are decoded.
3. The method of claim 1 or claim 2, wherein rendering a decoded frame from the post-decoding jitter buffer comprises:
 - selecting a frame from the post-decoding jitter buffer that has a render time that is closest to and comes before a current time plus a render delay; and
 - rendering the selected frame.
4. The method of claim 3, wherein rendering frames from the post-decoding jitter buffer further comprises:
 - removing the selected frame and frames having a render time before the selected frame from the post-decoding jitter buffer; or
 - removing frames from the post-decoding jitter buffer having a render time before the selected frame after rendering the selected frame.
5. The method of claim 1, wherein determining the render time for a frame comprises:

determining a jitter delay; and

determining the render time as the estimated arrival time plus the jitter delay.

6. The method of claim 1 or claim 2, wherein the video stream is one of a plurality of received video streams, and rendering includes:

creating a conference mix by rendering at least some frames of at least some of the plurality of received video streams into the conference mix.

7. The method of claim 6, wherein the plurality of video streams include video streams having at least two different frame rates.

8. The method of claim 7, wherein each video stream of the plurality of video streams is received from one of a plurality of stations, and the created conference mix is sent to at least one of the plurality of stations.

9. The method of claim 8, wherein a different conference mix is created for at least two stations of the plurality of stations.

10. An apparatus for rendering a video stream, the video stream having a plurality of frames, the apparatus comprising:

a memory; and

a processor configured to execute instructions stored in the memory to:

receive at least some of the plurality of frames,

determine a render time for at least some of the received frames based on an estimated arrival time,

decode at least some of the received frames,

store the decoded frames in a post-decoding jitter buffer, and

render at least some of the stored decoded frames from the post-decoding jitter buffer at their render times.

11. The apparatus of claim 10, further comprising instructions to:

store the received frames in a pre-decoding jitter buffer; and
decode a stored received frame and remove it from the pre-decoding jitter buffer
when its render time is determined

12. The apparatus of claim 10, wherein the instructions to render one of the
decoded frames from the post-decoding jitter buffer include instructions to:

select a frame from the post-decoding jitter buffer that has a render time that is
closest to and comes before a current time plus a render delay; and
render the selected frame.

13. The apparatus of claim 12, wherein the instructions to render one of the
decoded frames from the post-decoding jitter buffer further include instructions to:

remove the selected frame and frames having a render time before the selected
frame from the post-decoding jitter buffer; or

remove frames from the post-decoding jitter buffer having a render time before
the selected frame after rendering the selected frame.

14. The apparatus of claim 10, wherein the video stream is one of a plurality
of received video streams, and the memory further includes instructions to:

create a conference mix by rendering at least some frames of at least some of the
plurality of received video streams into the conference mix.

1/6

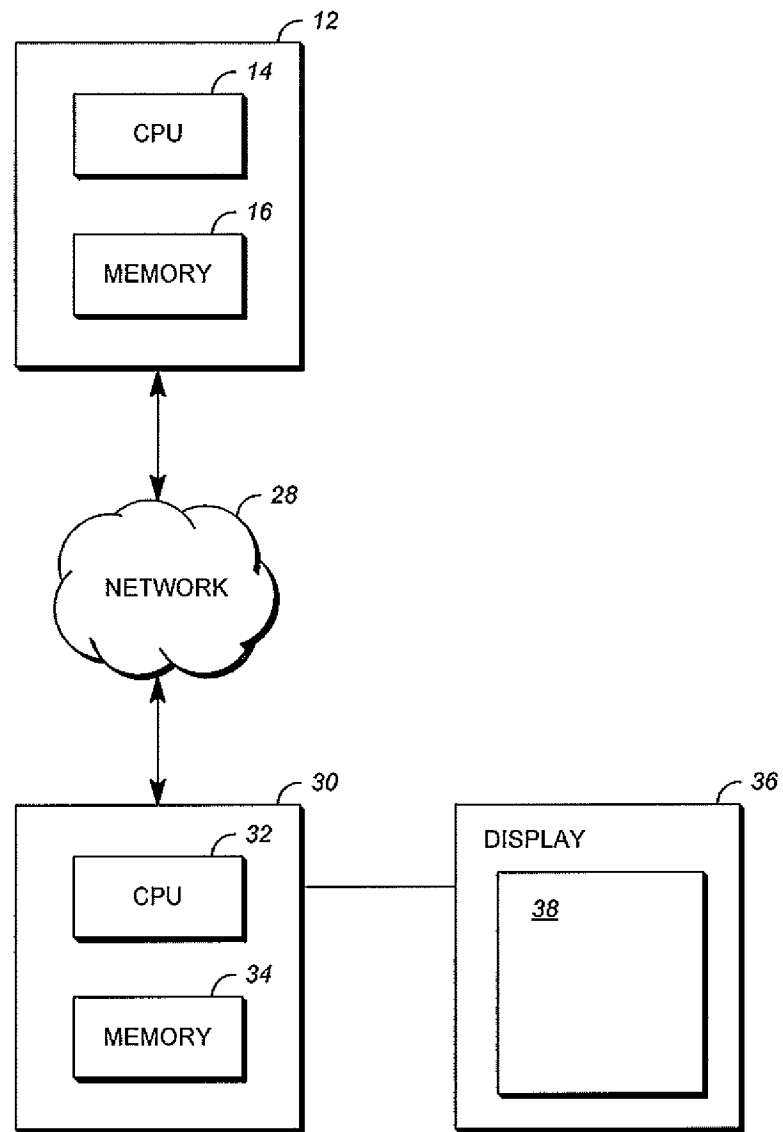
10

FIG. 1

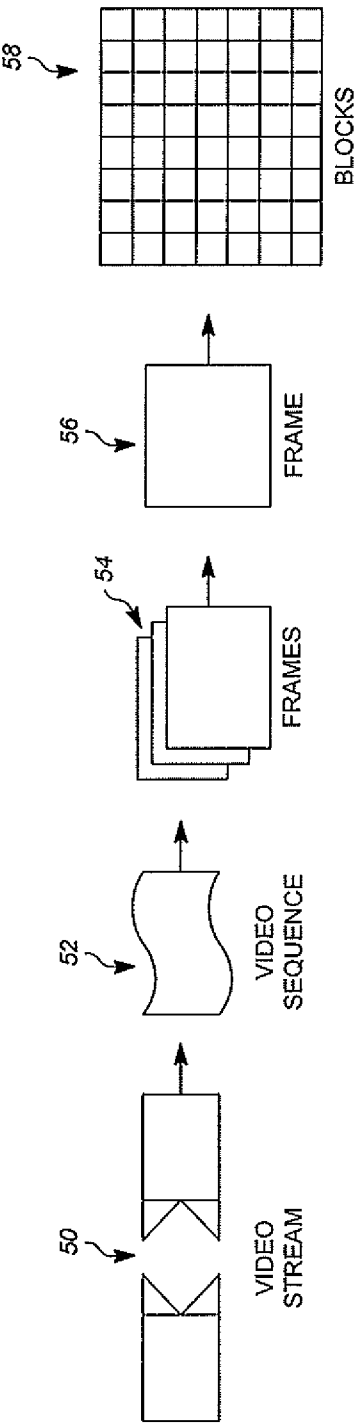


FIG. 2

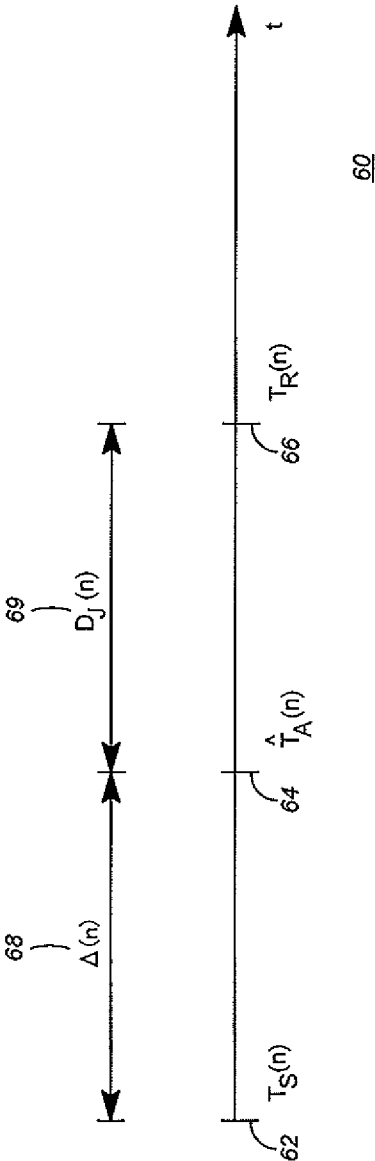


FIG. 3

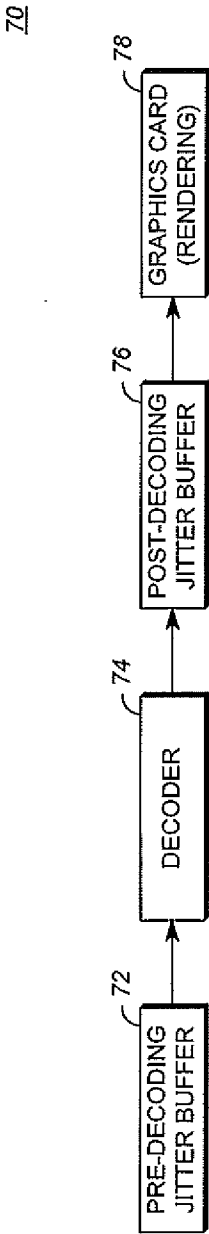


FIG. 4

4/6

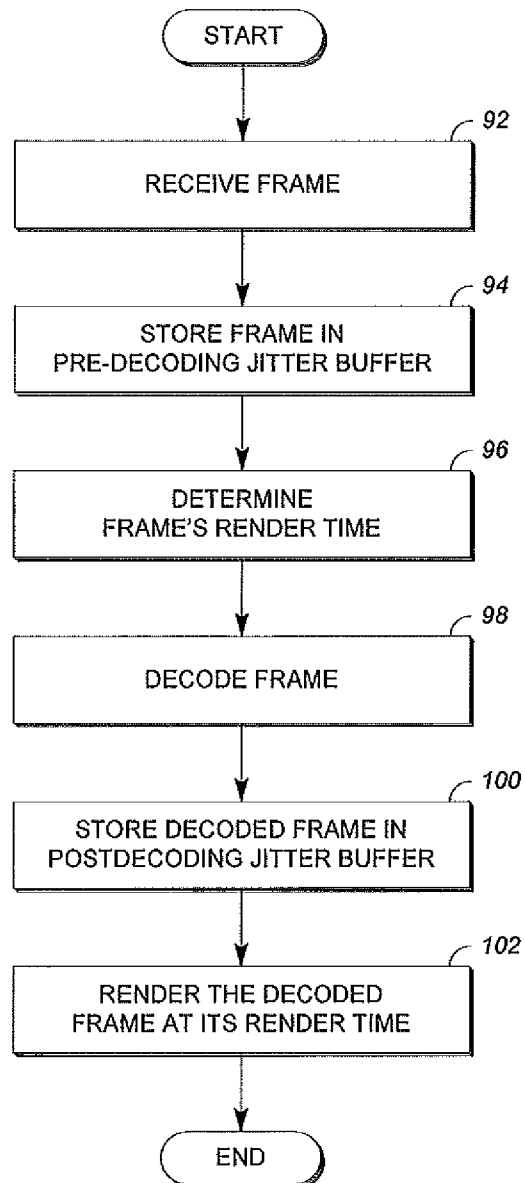


FIG. 5

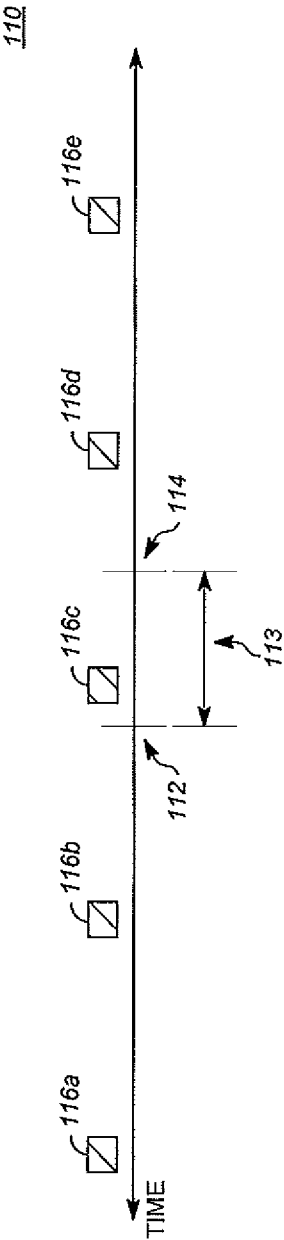


FIG. 6

120	
122	124
126	

FIG. 7

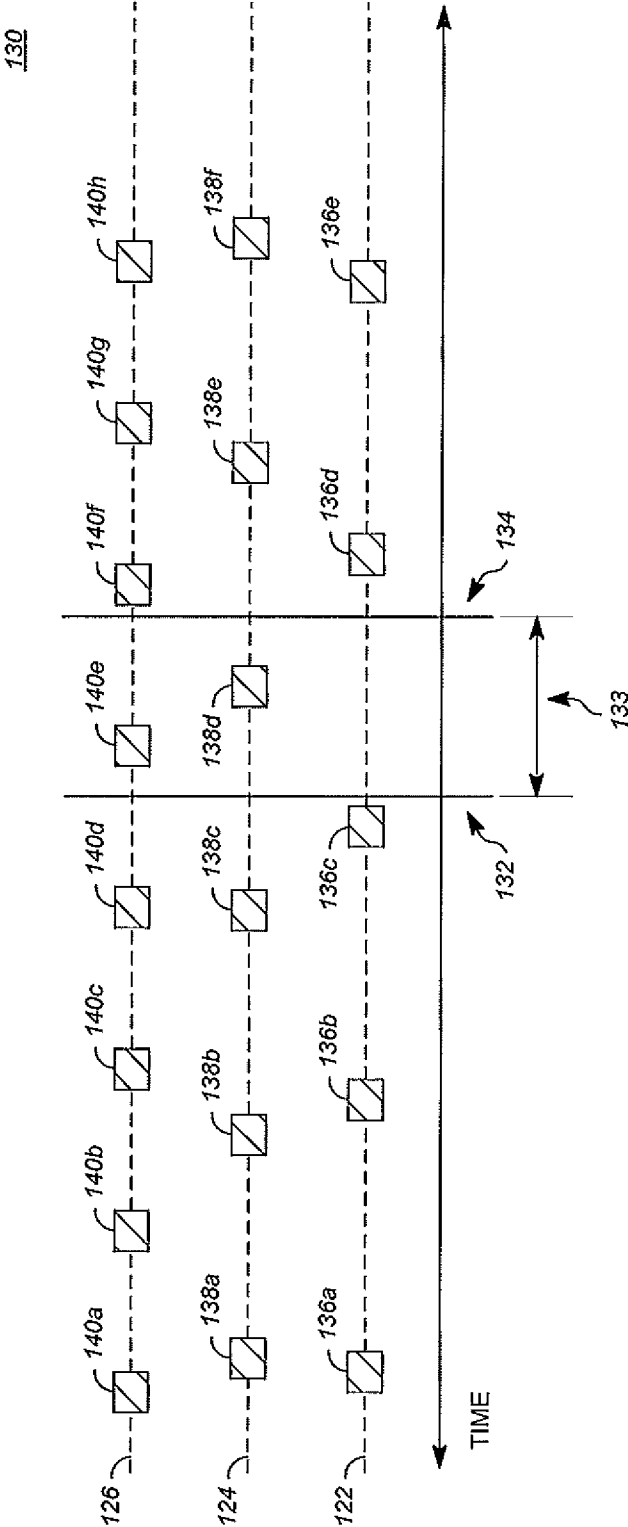


FIG. 8

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2011/035494

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N21/44 H04N21/43 H04N7/15
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 practical, 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.
Y	<p>US 2009/262252 A1 (WAN WADE [US]) 22 October 2009 (2009-10-22) figure 1B paragraph [0020] paragraph [0024] paragraph [0026] paragraph [0052]</p> <p>----- -/--</p>	1-14



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 document 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 February 2012

Date of mailing of the international search report

06/03/2012

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

Boubal, Olivier

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2011/035494

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	<p>TASAKA S ET AL: "LIVE MEDIA SYNCHRONIZATION QUALITY OF A RETRANSMISSION-BASED ERROR RECOVERY SCHEME", ICC 2000. 2000 IEEE INTERNATIONAL CONFERENCE ON COMMUNICATIONS. CONFERENCE RECORD. NEW ORLEANS, LA, JUNE 18-21, 2000; [IEEE INTERNATIONAL CONFERENCE ON COMMUNICATIONS], NEW YORK, NY : IEEE, US, vol. 3, 18 June 2000 (2000-06-18), pages 1535-1541, XP001208669, ISBN: 978-0-7803-6284-0 figure 1 paragraph [0001] paragraph [2.2.1]</p>	1-14
A	<p>-----</p> <p>AU 2003 271 320 A1 (CANON KK) 2 September 2004 (2004-09-02) page 8, line 14 - line 17 page 13, line 16 - line 17 page 10, line 11 - line 12 page 10, line 22 - line 24 page 13, line 18 - line 22 page 16 - page 17 page 21</p>	1-14
A	<p>-----</p> <p>EP 1 096 804 A2 (MATSUSHITA ELECTRIC IND CO LTD [JP]) 2 May 2001 (2001-05-02) paragraph [0016] - paragraph [0017] paragraph [0108] paragraph [0112] - paragraph [0116] paragraph [0122] paragraph [0134]</p>	1-14
A	<p>-----</p> <p>EP 2 124 447 A1 (ERICSSON TELEFON AB L M [SE]) 25 November 2009 (2009-11-25) paragraph [0027] - paragraph [0029] paragraph [0034] - paragraph [0039] paragraph [0043]</p> <p>-----</p>	1-14

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2011/035494

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2009262252	A1	22-10-2009	NONE

AU 2003271320	A1	02-09-2004	NONE

EP 1096804	A2	02-05-2001	CN 1309505 A 22-08-2001
		DE 60032315 T2	28-06-2007
		EP 1096804 A2	02-05-2001
		KR 20010051249 A	25-06-2001
		US 6757332 B1	29-06-2004

EP 2124447	A1	25-11-2009	NONE
