

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
24 August 2006 (24.08.2006)

PCT

(10) International Publication Number
WO 2006/089052 A2

- (51) International Patent Classification:
H04N 7/173 (2006.01)
- (21) International Application Number:
PCT/US2006/005520
- (22) International Filing Date:
16 February 2006 (16.02.2006)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
60/653,569 16 February 2005 (16.02.2005) US
- (71) Applicant (for all designated States except US): NMS COMMUNICATIONS CORPORATION [US/US]; 100 Crossing Boulevard, Framingham, Massachusetts 01702 (US).
- (72) Inventors; and
- (75) Inventors/Applicants (for US only): KATZ, Michael [US/US]; 29 Almeria Circle, Westford, Massachusetts 01886 (US). BISHOP, Wendell E. [CA/US]; 110 Antlers Shore Drive, East Falmouth, Massachusetts 02536 (US). BASSO, Andrea [IT/US]; 32 Provident Court, Ocean, New Jersey 07712 (US).
- (74) Agents: LUTZ, Christopher J. et al.; Chapin Intellectual Property Law, LLC, Westborough Office Park 1700 West Park Drive, Westborough, Massachusetts 01581 (US).

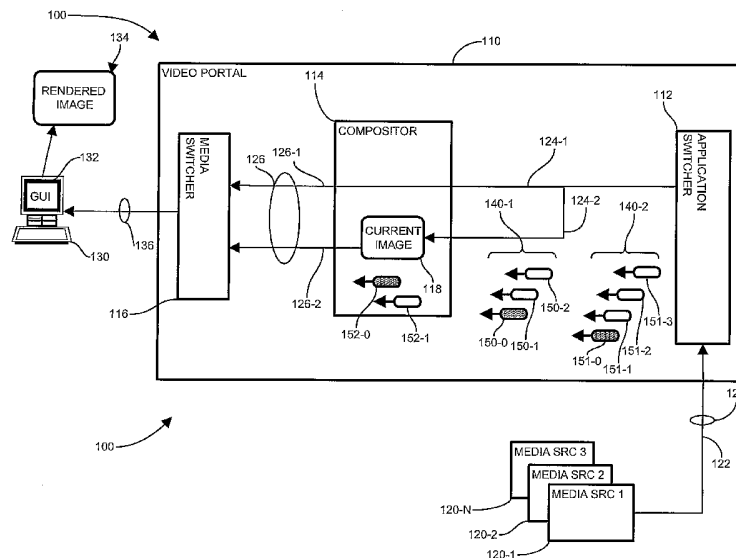
- (81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.
- (84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))
- as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))
- of inventorship (Rule 4.17(iv))

[Continued on next page]

(54) Title: METHODS AND APPARATUS FOR DELIVERING SERVICES



(57) Abstract: A video portal delivers video services as disclosed herein by identifying a media stream having a plurality of segments, in which each of the segments is defined by an ordered sequence of frames, and receiving a request to direct the identified media stream to a requesting user, such that the request is received during a current segment in the stream. The video portal operable according to configurations discussed herein redirects the media stream to the user in a realtime manner by immediately sending a decoded, computed current frame to the user in a seamless manner for rendering on a user display device, the seamless manner operable to preserve synchronization of the display image with the computed current frame, providing frames remaining in the current segment, and concatenating the provided frames with a successive next segment of the media stream.

WO 2006/089052 A2



Published:

— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

METHODS AND APPARATUS FOR DELIVERING SERVICES

BACKGROUND

In a media services environment, users receive a multitude of video services via a user display device. Often, the user display device is a personal communications device such as a cellular phone, palm pilot or personal digital assistant (PDA). The user device is operable to render a video signal from a service provider for display to a user, and may be responsive to a variety of transmission mediums, including wireless, broadband, digital subscriber line (DSL) and others.

The video services often emanate from different service provider applications, and are switched to the active user stream by a switching mechanism at the service provider. For example, a typical user may have available options includes video mail, stock tickers, interactive video games, current motion picture trailers, traffic, news, and others. An active user may wish to switch rapidly between the different streams provided by the various service provider applications, such as applications transmitting so-called "3G24M video," as is known in the art. Accordingly, many user devices offer a user interface operable for "point-and-click" type selection of the various video streams available from network application services.

SUMMARY

Users often receive video services from a video portal in communication with various video service applications. In a conventional video portal providing video services to users, a plurality of video service applications (media sources) provide a media stream to a user display device and allow on-demand switching from the different media sources to a user feed rendered on the user device. The typical user device includes cell phones, laptops, PDAs, desktop PCs, or other computing device enabled for personal communications. Such media sources are typically video and may include, for example, video mail, interactive video games, movie trailers, or other video or multimedia applications.

Unfortunately, conventional media stream delivery mechanisms suffer from the shortcoming that media stream switching introduces periods of undefined or unintelligible frames into the user delivery path. While it may not affect the functional delivery of the conventional video stream, such a delay, typically observable as a "blip" of blank, undefined, or unintelligible screens, at the very least interrupts the user experience and introduces a delay

that prevents a seamless switching of video streams. In a user delivery device, such as a handheld mobile phone or other personal communications device, users expect an ability to perform frequent, immediate switching between incoming media streams, typically video. Delays of even a brief interval tend to become annoying and objectionable when a user
5 desires to flip between a plurality of video streams in succession.

Accordingly, configurations of the invention are based, in part, on the observation that conventional video service users tend to experience an objectionable delay or “blip” when switching between video services. In particular, users of frame based services, such as conventional video on demand, observe a delay or period of unintelligible displayed images
10 upon switching from a particular video stream to another video stream. Configurations discussed herein substantially overcome the above described shortcomings by maintaining a decoded display frame of the requested video stream and delivering the display frame for rendering on a user device immediately upon a request from the user. Conventional end user video switching devices do not maintain a display image of incoming streams but rather pass
15 through the video stream directly to the user.

In a conventional video stream, successive frames are often similar to a previous frame, and need not include information about unchanged portions. In other words, such frames only need include information pertaining to changed portions, or pixels, of the corresponding display screen. Conventional video streams avoid replicating duplicative
20 information between frames by transmitting a base, or complete, frame representing every pixel in the display image, followed by a relative frame including only the changed portions of the display image. Such a sequence including a base, or I-frame, followed by a series of relative, or P-frames, is called a segment and recurs periodically as the displayed image changes. However, the relative frames assume the context established by previous relative
25 and base frames. Conventional devices, therefore, are unable to deliver a coherent image until the next base frame (I-Frame). Unless the user request arrives at the exact moment as a new base frame, for example, the requested stream cannot be merged (a process called concatenation) with the user delivery stream until the next base frame. It is computationally expensive to maintain a renderable (encoded) frame of each potential stream, and merely
30 switching the relative frames renders them unintelligible without the corresponding base frame.

Therefore, configurations herein accumulate a display frame, or current image, representing the most recent base frame and successive relative frames for each of a plurality

of anticipated video streams. Upon receipt of a switching request from a user, the display frame corresponding to the requested stream is encoded and transmitted to a user for rendering on the display device. Each of the anticipated video streams is decoded as successive relative frames are received, thus maintaining a current display image
5 corresponding to each of the anticipated streams. Upon the user request, the particular stream is deliverable in a seamless manner because only the requested video stream is encoded. The system need not encode each of the anticipated media streams pending a request from a user. Since decoding is computationally less expensive than encoding, a video compositor accumulating the anticipated streams decodes each stream upon receipt, and only encodes in
10 response to a user request.

In a typical video media stream therefore, a complete base frame (I-frame) is periodically received, every 5-10 seconds for example, followed by interim relative frames (P-frame) for the remaining frames in a segment of frames until the next complete I-frame. Conventional mechanisms cannot interpret the intervening P-frames intelligibly without the
15 corresponding I frames. Therefore, simply switching the user feed to the requested video stream results in garbled or unintelligible P-frames until the next I frame.

In contrast, configurations herein perform decoding on each anticipated stream to maintain a display frame indicative of each I-frame and aggregated P-frames. The current display frame is encoded as a base or I-frame and transmitted to a user in response to a video
20 stream request. Successive relative frames (P-frames) are applied (decoded) to the current display image, and an interim stream of new P-frames (referenced to the newly created I-frame) is immediately (real time) encoded and sent to the user, until the next I-frame is received, at which point the user feed is switched to the incoming stream for receipt of successive (already encoded) frames that are simply fed through, thus concatenating the
25 interim stream with the requested stream. The end user device (i.e. cell phone) is then operable to decode and display the I-frame and successive P-frames. In this manner, the video switching and compositing device need only assign an encoder to requested streams for the remaining frames in a segment until the next I- frame is received. Since the switching and compositing device need only perform encoding for incomplete I-frame segments, encoders
30 may be switched as needed among requesting users for providing a realtime viewing experience.

Similarly, other media forms, such as audio, may also experience an unsatisfactory “blip” or interruption when switching between services. In contrast, the method disclosed

herein provides a way to switch between services, or between internal media (e.g. menus) and services, that is relatively instantaneous from the user's perspective, but also results in minimal degradation of video quality, and places minimal load on portal system resources.

The method involves having a video portal keep active sessions (continuous video) from various 'live feed' services available at all times, continuously decoding the video stream from each of those services, and encoding the video stream from each service for only a brief period at the beginning of a session a user requests connection to the service.

Without the portal as disclosed herein, the user must establish separate sessions (phone calls) with different services. With such a portal, only a single session is established between the user and the portal in order to make available multiple services. The portal, as an agent of the user, is operable to establish a separate conventional session per user with each service application only at the time of a user request to connect. This session establishment (using telephony calling protocols and H.245) takes considerable time, and injects an objectionable delay in the interaction with the user. The disclosed method avoids this delay by having sessions pre-established with each of the services.

In further detail, the method of delivering video services as disclosed herein includes identifying a media stream having a plurality of segments, in which each of the segments is defined by an ordered sequence of frames, and receiving a request to direct the identified media stream to a requesting user, such that the request is received during a current segment in the stream. A video portal operable according to configurations discussed herein redirects the media stream to the user in a realtime manner by providing frames remaining in the current segment, and concatenating the provided frames with a successive next segment of the media stream.

The video portal switches the media stream in a manner that preserves synchronization between an existing display image at a user display device and the redirected media stream. In the exemplary configuration, the portal generates relative frames corresponding to the switched media stream for the remainder of a current segment, in which the relative frames complement a complete frame in the current segment.

Configurations discussed herein are operable with a user display device operable to render a visual image from an encoded frame, in which encoding incurs greater resources than decoding. Such user display devices include, for example, wireless phones, PDAs, WiFi enabled laptops, wired PCs, or any suitable video enabled personal communication device. Rendering further includes computing a displayable image from a complete frame and at least

zero relative frames of the media stream, since a switch may occur at an initial I-frame (complete frame, as discussed herein) or any successive P-frame (relative frames, as discussed herein). Thus, the received frames include complete frames and relative frames, the relative frames indicative of changes from a previous frame, wherein a display image includes a complete frame and a sequence of at least zero relative frames.

In the exemplary configuration, the video portal redirects the requested video stream by aggregating received frames in the media stream to maintain a current frame, and decodes each of the aggregated frames to generate a renderable frame. A compositor receives successive relative frames from the media stream, and computes the difference between the relative frame and the renderable frame. The compositor generates interim frames (P-frames) indicative of the difference between the relative frame and the renderable frame for rendering an accurate depiction of the switched media stream on the user device. Computing the difference further includes decoding the successive relative frame, and applying the decoded relative frame to a renderable image indicative of a current display state of the media stream. The compositor invokes an encoder to encode the renderable image to generate a relative frame indicative of changes between the renderable image and the current display state.

In the exemplary configuration, redirecting the media stream further includes receiving each of the frames in a segment, and aggregating the frames to maintain a display frame corresponding to a display image. The video portal generates an interim image from the display frame, and receives successive relative frames indicative of incremental changes to the display image.

Redirecting, in the exemplary arrangement, further includes receiving a request to redirect the media stream to a user, and computing a current frame indicative of each of the frames in the sequence received for the current segment. The video portal immediately sends the computed current frame to the user in a seamless manner for rendering on a user display device, such that the seamless manner is operable to preserve synchronization of the display image with the computed current frame. The compositor receives the remaining frames in the current segment, and computes, for each received frame, changes contained therein. The compositor then computes the difference between the computed current frame and the computed changes in a relative frame (P-frame), and transmits the computed difference to the user display device for rendering on the user display device.

To perform the switchover, in the exemplary arrangement, the compositor accumulates successive relative frames to maintain the current frame, such that generating

interim frames occurs upon receiving a request for delivery of the media stream, in which the request arrives in the middle of a segment.

The exemplary compositor receives a plurality of media streams, and maintaining a current image of each of the plurality of media streams, in which maintaining including
5 decoding a current frame of each of the received media stream in anticipation of a switch request. Upon receiving such a request, the media portal switches a selected one of the plurality of media streams in response to the received request, such that switching is performed by rendering the current frame corresponding to the selected media stream, and concatenates the switched media stream to an existing user media stream without loss of
10 synchronization on the user display device.

In the exemplary arrangement, switching further includes computing a current display frame based on the successive relative frames applied to the display image, and computing changes between the computed current display frame and the generated interim image. The compositor encodes a relative frame indicative of the computed changes, such that the
15 relative frame is operable to render the interim images similar to the display image from the media source. A media switcher then transmits the encoded relative frame to a user device for rendering, such that rendering is operable to display the display frame including the aggregated frames in the segment.

Alternate configurations of the invention include a multiprogramming or
20 multiprocessing computerized device such as a workstation, handheld or laptop computer or dedicated computing device or the like configured with software and/or circuitry (e.g., a processor as summarized above) to process any or all of the method operations disclosed herein as embodiments of the invention. Still other embodiments of the invention include software programs such as a Java Virtual Machine and/or an operating system that can
25 operate alone or in conjunction with each other with a multiprocessing computerized device to perform the method embodiment steps and operations summarized above and disclosed in detail below. One such embodiment comprises a computer program product that has a computer-readable medium including computer program logic encoded thereon that, when performed in a multiprocessing computerized device having a coupling of a memory and a
30 processor, programs the processor to perform the operations disclosed herein as embodiments of the invention to carry out data access requests. Such arrangements of the invention are typically provided as software, code and/or other data (e.g., data structures) arranged or encoded on a computer readable medium such as an optical medium (e.g., CD-ROM), floppy

or hard disk or other medium such as firmware or microcode in one or more ROM or RAM or PROM chips, field programmable gate arrays (FPGAs) or as an Application Specific Integrated Circuit (ASIC). The software or firmware or other such configurations can be installed onto the computerized device (e.g., during operating system or execution environment installation) to cause the computerized device to perform the techniques explained herein as embodiments of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing and other objects, features and advantages of the invention will be apparent from the following description of particular embodiments of the invention, as illustrated in the accompanying drawings in which like reference characters refer to the same parts throughout the different views. The drawings are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the invention.

Fig. 1 is a context diagram of an exemplary managed information environment 100 having media streams suitable for use with the present invention;

Fig. 2 is a flowchart of media stream operations the environment of Fig. 1;

Fig. 3 is a block diagram of media stream processing as disclosed in the flowchart of Fig. 2;

Fig. 4 is a data flow diagram depicting information flow in the media stream in Fig. 3; and

Figs. 5-9 are a flow chart of media streams from multiple sources as depicted in Figs. 3 and 4.

DETAILED DESCRIPTION

Configurations herein accumulate a display frame including the base frame and successive relative frames, indicative of a current image, for each of a plurality of anticipated video streams. Upon a request from a user, the display frame corresponding to the requested stream is encoded and transmitted to a user for rendering on the display device. Each of the anticipated video streams is decoded as successive relative frames are received from the corresponding video stream, thus maintaining a display image corresponding to each of the anticipated streams. Upon receipt of a user request, the particular stream is deliverable in a seamless manner because only the requested video stream is encoded. The system need not encode each of the anticipated media streams pending a request from a user. Since decoding

is computationally less expensive than encoding, a video compositor accumulating the anticipated streams decodes each stream upon receipt, and only encodes in response to a user request.

5 A video feed from a service application to a user display device, or terminal, is typically in the form of an H.263, MPEG4, or H.264 video stream. Configurations herein typically employ a live feed in the same format as the active session from the user to the portal. However, multiple live feeds with different standards from the same service may be used, if necessary, to handle users with different terminal types. Such streams typically have periodic I-frames (every 5-10 seconds) with many P-frames between any two I-frames. An I-
10 frame is where the stream 'starts from scratch', and a P-frame uses knowledge of what came before in the stream (back to the previous I-frame). A user switched into an already active service at an arbitrary time would receive P-frames for a period of possibly several seconds until the next I-frame arrives. During that time, the video image would be highly degraded. This provides an unacceptable user experience. Configurations disclosed herein avoid this
15 degradation by starting the stream from a new service application with an initial I-frame.

Alternatively, in conventional mechanisms, the connection of a user to a new service can be delayed until the next I-frame from that service arrives. This does not inject degraded video into the stream to the user, but does cause a delay of possibly many seconds before the connection can be made. This delay is objectionable and to be avoided as well. The
20 exemplary arrangement herein avoids this delay by starting with the most recent frame from the corresponding video service application, whether I-frame or P-frame.

Another conventional alternative is to buffer all frames from each service application between one I-frame and the next, and to start video to the user with the 'old' I-frame from the service. The entire successive stream must be delayed by the period between the arrival
25 of the old I-frame and the current frame. This results in an immediate connection to the user, but the video that the user sees is delayed from that which is sent by the service, by up to several seconds. For some services, this may be an acceptable user experience. For other services, especially those for which a single user may interact with the service application itself to cause a change of the video feed, the delay in the stream appears as a delay in
30 response to user input, which is also objectionable and to be avoided. The disclosed video portal avoids this delay by not buffering any data from the service applications, but only working with the most recent frame from each.

Yet another conventional alternative is to decode the video feed from each service at all times, and as a new user requests service, to assign a conventional encoder to that user in order to start encoding the stream to that user on demand. The stream would then continue to be encoded for that user for the duration of the connection. The typical portal may connect to only a few service applications, but may service many users. Video decoding consumes significantly fewer processing resources than does video encoding. Running a decoder continuously for each service application is thus acceptable. However, running an encoder continuously for each user is very costly. Furthermore, there is some video quality degradation in decoding and re-encoding a video stream, versus simply passing through the stream from the service application to the user as is. The configuration discussed below avoids the extra cost of processing resources, and the video degradation, by only assigning an encoder to a user for a brief period at the beginning of a stream from a new service application. A system with many users may use statistical sharing of a few encoding resources across many users if each user consumes an encoding resource for only a small fraction of the total session.

According to configurations discussed further below, the video portal (portal) will pre-establish connections with a set of available service applications. In some cases, where a user needs a private session with a service application (video mail, for instance) the portal may pre-establish one or more sessions, and as users consume those sessions, pre-establish additional sessions such that there is always a session ready to go without requiring session setup time when a user decides to connect. For each pre-established session, a video decoder will be assigned and active. For each decoder, each input frame from a service application will be decoded as soon as possible (in real time within a frame period, as portal resources allow), and only the most recently decoded frame (in bit map or pixel form) will be retained. In the idle state, before any users ask for connection to service applications, no encoding resources will be assigned - no service application streams will be re-encoded.

For each user that chooses to connect to a particular service application, the following applies. (For multiple users to either the same or different service applications, the following applies to each user. For a user who accesses multiple service applications successively, the following applies to each access.) An encoder is assigned and immediately encodes the most recently stored pixel (decoded) frame for the service application into an I-frame that is immediately sent to the user. Then for each subsequent frame received as input from the service application, there is first a check to see whether it is an I-frame or a P-frame. If it is

an I-frame, the encoder is shut down and released, the I-frame is forwarded along as is to the user, and all subsequent frames on that channel, whether P-frames or I-frames, are forwarded through as is to the user. If instead it is a P-frame, the frame is decoded as usual, and the output of the decoder is fed to the input of the encoder, which is requested to produce a P-frame that is then sent on to the user.

In this fashion, an encoder will only be assigned for a brief period (few seconds) at the beginning of a connection to a new service application, until the next input I-frame arrives. During that period, there will be some minimal video degradation from re-coding, and there will be extra consumption of resources due to use of the encoder. But for the major portion of the connection, there will be no video degradation and no extra use of resources.

Note that while an encoder is active for a given user, the decoding of frames for the service application continues undisturbed. If the service application has a live feed that may be shared (broadcast), a second or more subsequent users may be attached using the same set of decoded frames. Only a new encoder needs to be assigned per user.

Fig. 1 is a context diagram of an exemplary managed information environment 100 having media streams suitable for use with the present invention. Referring to Fig. 1, the managed information environment 100 includes a video portal 110 connected to a plurality of media sources 120-1..120-N (120 generally) and operable to deliver the media sources 120 to a user display device 130 (user device), such as a cellular phone, personal computer (PC), laptop, personal digital assistant (PDA) or other wired or wireless personal communication device. The user device 130 has a graphical user interface 132 for presenting a rendered image 134 from a user feed 136 supplied by the video portal 110. Note that the user feed 136 may represent an intra-device interface, as the video portal 110 may be co-located in the user device 130. Alternatively, the user feed may represent a local wired connection, local wireless (WiFi) connection, or any suitable multimedia interface. Similarly, the media sources 120 connect to a media interface 122 such as a wired or wireless internet connection, typically from a video service provider (not specifically shown) or other suitable interconnection to a video service, discussed further below.

The video portal 110, therefore, interconnects with a plurality of incoming video sources 120-N for selectively directing the video sources to the user device 130 as a rendered image. The video portal 110 includes an application switcher 112, for selecting media streams 124 from the various media sources 122, a compositor 114 for performing seamless concatenation of the media streams, discussed further below, and a media switcher 116, for

assigning a renderable media stream 126 to the user feed 136 for rendering (display) on the user device 130.

The media streams 124 each include a plurality of segments 140-1, 140-2 (140 generally) each having a sequence of frames 150-0..152-N. Each segment 140 includes a first complete frame 150-0, 151-0 and 152-0 optionally followed by one or more relative frames 150-1..150-N, 151-1..151-N, 152-1..152-N. In the discussion that follows herein, the 0th frame is the complete, or I-frame, and successive (1+N) frames are relative, or P-frames, N being 0 or greater, for the frame sequence 150, 151 and 152. Each of the complete frames 150-0 represents a complete pixel definition of a visible display image, and the successive relative frames 150-1..150-N represent changes to the previous complete frame 150-0. In other words, each relative frame, say 150-5 for example, represents changes to the aggregation of frames 150-0 through 150-4.

In configurations discussed herein, the compositor 114 maintains a current image 118 representing the current state of the image encompassing each received frame 150 in a particular media stream 124-2, typically a video stream. This video stream 124-2 represents a seamless video stream operable to be immediately rendered to the user according to principles discussed further below. The compositor 114 also receives and passes through a direct feed 124-1. As indicated above, the media switcher 116 switches various video feeds 126 to the user feed 136. However, a switchover in the middle of a segment 140 presents seamlessness issues discussed further below. Accordingly, the compositor 114 maintains the current image 118 of a media stream 124-2 indicative of complete 150-0 and relative frames 150-(1+N) received up until the point a switchover is requested. In this manner, the current image 118 is encoded as a complete frame 152-0, renderable as the rendered image 134, immediately upon receipt of a switchover command 166 (Fig. 3) from the user device. The compositor 114 is further operable to deliver relative frames 152-(1+N) from the maintained media stream 124-2 for the remainder of the segment 140-1, at which point the complete image 151-0 of the successive segment may be switched, or fed through as the unprocessed media stream 124-1.

Fig. 2 is a flowchart of media stream operations the environment of Fig. 1. Referring to Figs. 1 and 2, the method of delivering video services as disclosed herein includes, at step 200, identifying a media stream 124 having a plurality of segments 140, in which each of the segments 140 is defined by an ordered sequence of frames 150-N. Typically a user receives a new segment every 5-10 seconds, such as a new scene in the video image. Each segment 140 begins with a complete frame 150-0, or I-frame, followed by a series of relative frames 150-

(1+N), or P-frames, defining changes to the image 118, for the remainder of the segment. At step 201, the video portal 110 receives a request 166 to direct the identified media stream 124 to a requesting user, such that the request is received during a current segment 140 in the stream 124. The request is typically a user request from the user device 130, such as a icon or button to select the stream 124. In response, the media portal redirects the media stream 124 to the user device 130 in a realtime manner by providing frames 150-N remaining in the current segment 140. The frames 150-N are interim frames provided for the remainder of the segment because they relate not to the initial I frame 150-0 in the segment but to the I frame 152-0 representative of the current image 118 delivered as the first frame of the redirected stream 124-2. At the conclusion of the current segment 140, the video portal 110 concatenates the provided frames 152 with a successive next segment 140 of the media stream 124-1. The media switcher 116 delivers the next complete frame 151-0 (I-frame) from the selected media stream 124-1 to the user feed 136, thus concatenating or merging the interim video stream 126-2 with the main feed of the media stream 124-1. Therefore, the compositor 114 need only deliver the interim stream 126-2 for the remainder of the current segment 140-1. As will be discussed further below, this switching frees up encoding resources after the switchover is complete, thus enabling the encoder (164, Fig. 3) to attend to other switching requests.

Fig. 3 is a block diagram of media stream 124 processing as disclosed in the flowchart of Fig. 2. Referring to Figs. 1 and 3, the compositor 114 in the video portal 110 is shown in greater detail. The compositor 114 includes a plurality of current images 118-1..118-N, each corresponding to an incoming media stream 122-1..122-N. The compositor also includes one or more decoders 162-1..162-N (162 generally), one or more encoders 164-1..164-N, operable to decode and encode frames 150 in the media streams 124. The current images 118 are decoded pixel based forms operable for rendering on a user device 130. Therefore, the current images 118 are operable to represent (store) an image including a complete frame 150-0 and a series of successive relative frames 150-(1+N). In other words, the current image 118 stores aggregate information from a particular video stream 124 sufficient to render the image corresponding to the most recently received frame, whether it is a complete or relative frame.

The decoders 162 are operable to receive an incoming media stream 124 from a media source 120, and decode, or interpret, the information therein into a renderable frame in the current image 118. The incoming media streams 124 emanate from a variety of media

sources 120, typical video service applications, and may arrive over the network 128 via a variety of transport mechanism, such as IP, GSM, CDMA, TDMA, broadband, DSL or any suitable packet switched transport mechanism. In the media stream 124, the encoded relative frames 150-(1+N) are not renderable without the information contained in the previous complete frame 150-0 and intervening relative frames in the corresponding segment. The encoder 164 is operable to encode the current image 118 and transmit it via the user feed 136 to the user for rendering on the user device 130. Therefore, when a video switch request 166 arrives from the user device, the encoder 164 is operable to seamlessly encode the current image 118-N for the requested media stream 124-2 and transmit it to the user device 130. A corresponding relative frame in the media stream 124-1 emanating from the corresponding media source 120 would be unrenderable on its own, and the user device 130 would be forced to wait for the next complete frame or perform alternate interim processing to accommodate the request, typically resulting in an unacceptable user experience.

Fig. 4 is a data flow diagram depicting information flow in the media stream in Fig. 3. Referring to Figs. 3 and 4, a rendered image 134-1 and 134-2 corresponding to media streams 124-1 and 124-2, respectively, switched by the compositor 114 are shown. Fig. 4 shows the relationship between segments, complete frames (I-frames) and relative frames (P-frames) as employed in the exemplary configuration. A direct feed video stream 124-1 has a segment 140-3 of frames including complete frame I-A0 and relative frames P-A1..P-A4. A seamless incoming video stream 124-2 switched according to principles herein defines the same video stream fed to the decoder 162. The decoded frames 150 are maintained as a current image 118 in complete frame I-B0 at a time of a user request 170, having been previously decoded from frames I-A0, P-A1 and P-A2, shown by bracket 172. At the time of the user request 170, the unswitched video stream 124-1 has received frames I-A0, representing a complete image, and relative frames P-A1 and P-A2. Accordingly, the rendered image 134-1 displays an image indicative of information in frames I-A1 + P-A1 + P-A2. Stream 124-2, being processed for seamless switchover according to principles herein, had decoded frames I-A1, P-A1 and P-A2 as current image I-B1. Accordingly, frame I-B1 represents the aggregated information contained in frames I-A1, P-A1 and P-A2, shown by bracket 172. Accordingly, at switchover time 170 the encoder 164 receives complete frame I-B1 and encodes a complete frame as a renderable image 134-2. Successive relative frames in the segment 140-3 are received and decoded as relative frames P-B1 and P-B2, containing the same information as frames P-A3 and P-A4, applicable to rendered image 134-1. Therefore,

at time 172, the rendered images 134-1 and 134-2 are equivalent, and include aggregated information from frames I-A0 + P-A1 + P-A2 + P-A3 + P-A4 and frames I-B0 + P-B1 + P-B2, respectively.

5 Figs. 5-9 are a flow chart of media streams 122-N from multiple sources 120-N as depicted in Figs. 2, 3 and 4. Referring to Figs. 2-9, at step 300, the application switcher 112 receives a plurality of media streams 122. The media streams are typically video streams, and emanate from a plurality of video service applications 120. The application switcher 112 identifies one or more media streams 124 having a plurality of segments, each of the segments defined by an ordered sequence of frames, as depicted at step 301. The selected
10 streams 124 are those operable for fast (realtime, or near realtime) switchover with concatenation, as defined further below. Receiving the frames 150 further includes receiving complete frames (I-frames) and relative frames (P-frames), such that the relative frames are indicative of changes from a previous frame, in which a display image 118 includes a complete frame 150-0 and a sequence of at least zero relative frames 150-(1+N), as shown at
15 step 302. The display image 118 (current image) is an aggregation of frames renderable on the user device 130 as a rendered image 134.

The compositor 114 maintains a current image 118-N of each of the plurality of media streams 124-N, such that maintaining including decoding a current frame 150 of each of the received media stream 124-N, as disclosed at step 303. The current frame is the most recently
20 received relative or complete frame 150-0, 150-(2+N) in each media stream 124. Therefore, the compositor 114 aggregates received frames in the media stream to maintain the current frame as the current image 118, as shown at step 304. The compositor 114 accumulates successive relative frames 150 to maintain the current image (frame) 118, as shown at step 305, and aggregates the received frames to maintain a display frame corresponding to a
25 display image, as depicted at step 306. In other words, the current image 118 is operable to be immediately rendered on the user device 130.

Upon receiving each relative frame, the compositor 114 computes a current display frame based on the successive relative frames 150 applied to the current image 118, as depicted at step 307. Therefore, the compositor 114 computes a current frame indicative of
30 each of the frames 150-N in the sequence, or media stream 124-N, received for the current segment 140, as shown at step 140. The compositor 114 applies the frames 150 by decoding the aggregated frames to generate a renderable frame as the current image 118, as disclosed at step 309. The current image 118 is therefore operable to be immediately re-encoded into an I-

frame for display (rendering) on the user device 130.

At a particular point in the transmission of the media stream 124, a user elects to switch media streams. In other words, the user issues a command to direct the video portal 110 to present a media stream 124 from a different video service application (media source) 120 to the user feed.136. Typically, as indicated above, the compositor 114 is maintaining the current image 118 for several media streams 124-N, such as video mail, interactive games, movie trailers, stock tickers, or any suitable video source. Upon the user activating a button or icon, the video portal 110 receives a request 166 to redirect a particular media stream 124-2 to a user. A check is performed, as depicted at step 310, and if no request 166 has been received, control reverts to step 303 to maintain the current streams 124. Otherwise, the compositor 114 receives a video switch request 166 during a current segment 140 following a previous I frame 150-0 on the media stream 124-2 providing the requested content, as disclosed at step 311. Although possible, it is unlikely that the request coincides with a complete frame 150-0, therefore, at step 312, the compositor 114 receives a request to direct the identified media stream 124-2 to a requesting user, in which the request is received during a current segment 140 in the stream, rather than at the beginning (first, or I-frame). Accordingly, generating interim images (the remainder of the current segment 140) occurs upon receiving a request for delivery of the media stream 124-2, in which the request arrives in the middle of a segment 140, as depicted at step 313.

In response, at step 314, the video portal 110 immediately sends the already computed current frame 152-0 to the user in a seamless manner for rendering on the user display device 130, such that the seamless manner is operable to preserve synchronization of the display image with the computed current frame (image) 118. Therefore, switching involves switching a selected one of the plurality of media streams 124-N in response to the received request 166, in which switching is performed by first rendering the current frame 118 corresponding to the selected media stream 124-2, as shown at step 315. Switching the media stream 124-2 is performed in a manner that preserves synchronization between an existing rendered display image 134 at the user display device 130 and the redirected media stream 124-2, as depicted at step 316. The redirected media stream 124-2 arrives first as a complete I-frame 152-0 encoded by the encoder 164, in which the user display device 130 is operable to render a visual image from an encoded frame, and in which encoding incurs greater resources than decoding, as depicted at step 317. Therefore rendering includes computing a displayable image 134 from a complete frame 150-0 and at least zero relative frames 150-

(1+N) of the media stream 124-2, as shown at step 318. Since encoding incurs additional resources, the compositor expects to commit encoding resources to the switchover only for the balance of the current frame, first with the interim I-frame 152-0 followed by the corresponding relative frames 152-(1+N), now discussed further below. In other words, the interim frame is an I frame inclusive of the I frame and all intervening P frames received on the switched media stream 124-1 up until the switchover point 170. The compositor 114 continues to receive successive relative (P-frames) from the media stream 124-2 of the corresponding video service application 122, as disclosed at step 319. Therefore, the compositor 114 provides an interim I-frame I-B0 corresponding to the previously generated I-frame I-A0 and successive series of P frames P-A1, P-A2 received on the incoming stream 124-1, such that providing frames further comprising generating P-frames for the remainder of the current segment 140, as shown at step 320.

After generating an interim image from the display frame for rendering the initial complete frame, the compositor 114 continues redirecting the media stream to the user in a realtime manner by providing frames remaining in the current segment 140.

At step 321, the compositor generates relative frames P-B1, P-B2 corresponding to the switched media stream 124-2 for the remainder of the current segment 140, such that the generated relative frames 150-(2+N) complement the complete frame 118 in the current segment 140. For each frame 150-(2+N), the compositor 114 computes the difference between the relative frame and the renderable frame (I-frame) representing the current image 118, as depicted at step 322. Note that the differences accrue in the rendered image 134, such that each successive P-frame includes differences from the previous P-frame. Computing the difference further includes, at step 323, decoding the successive relative frame, and applying the decoded relative frame to the renderable image 134 indicative of a current display state 134-1 of the media stream 124-1, as shown at step 324. Thus, the relative frame displays a rendered image 134-2 similar to what the unswitched stream would have displayed at the same P-frame.

From the differences, the compositor generates interim frames indicative of the difference between the relative frame 150-(1+N) and the renderable frame 134-2, as depicted at step 325. The generated interim frame results from computing changes between the computed current display frame and the generated interim image, as shown at step 326, thus computing, for each received frame, changes contained therein, as depicted at step 327. Therefore, the interim P-frame includes the difference between the computed current frame

118 and the computed changes in the received P-frame on the application stream 124-1.

The compositor 114 invokes a decoder 162 for encoding the renderable image (current image) 118 to generate a relative frame indicative of changes between the renderable image and the current display state (the image on the user display 132), as shown at step 328. This includes, at step 329, encoding a relative frame (P-frame) indicative of the computed changes, such that the relative frame operable to render the interim images similar to the display image (the image displayed from the direct feed 124-1). The encoder 164 thus encodes a change frame operable to apply the differences to the user display device 132, as shown at step 330. These encoded frames P-B1, P-B2 differ from the P-frames on the media stream 124-1 emanating from the switched media source 122 because they are relative to the interim I-frame 134-2, rather than the unprocessed media source stream 124-1 I-frame depicted at frame 134-1.

The compositor 114 continues receiving successive relative frames indicative of incremental changes to the display image 134-1 (the media source P-frames), as depicted at step 331. For the duration of the current segment 140, the compositor receives successive relative frames from the media stream 124-1, and continues providing interim P frames generated from successively received P frames on the incoming stream 124-1, as disclosed at step 332. For each received P-frame, this involves transmitting the computed difference to the user display device 130 for rendering on the user display device 130, as shown at step 333. Therefore, the compositor transmits the encoded relative frame 152-(2+N) to the user device 130 for rendering, such that the rendering is operable to display the display frame 134-2 including the aggregated frames I-B0, P-B1, P-B2 in the segment 140, as depicted at step 334. The compositor 114 therefore computes the difference by decoding the received remaining frames, (i.e. each interim frame generated from the decoder switchover stream 124-2), applying the changes to the maintained current image 118, and reencoding a display frame as an interim P-frame 152-(1+N) containing the changes.

A check is performed at step 335, to determine if there are more frames in the current segment 140. If so, then control reverts to step 320 for each remaining relative frame. Otherwise, the end of the current segment 140-1 has been reached and the media switcher 116 concatenates the switched media stream 124-2 to an existing user media stream 124-1 without loss of synchronization on the user display device 130, as shown at step 336. Therefore, the media switcher 116 concatenates the provided frames from the interim stream 126-2 with a successive next segment 140-2 of the media stream 126-1 as depicted at step 337. Upon the

receipt of the I-frame of the next segment 140, the interim stream 126-2 may be replaced with the application media stream 126-1 since successive frames may be sent directly through to the user from the application, without reencoding to provide interim relative frames as above.

Accordingly, the compositor 114 merges the generated steam 126-2 with the incoming
5 stream 126-1 upon receiving a successive I frame on the incoming stream 126-1, thus marking the beginning of the next segment 140, as shown at step 338.

Those skilled in the art should readily appreciate that the programs and methods for delivering media services as defined herein are deliverable to a user processing and rendering device in many forms, including but not limited to a) information permanently stored on non-
10 writeable storage media such as ROM devices, b) information alterably stored on writeable storage media such as floppy disks, magnetic tapes, CDs, RAM devices, and other magnetic and optical media, or c) information conveyed to a computer through communication media, for example using baseband signaling or broadband signaling techniques, as in an electronic network such as the Internet or telephone modem lines. The operations and methods may be
15 implemented in a software executable object or as a set of instructions embedded in a carrier wave. Alternatively, the operations and methods disclosed herein may be embodied in whole or in part using hardware components, such as Application Specific Integrated Circuits (ASICs), Field Programmable Gate Arrays (FPGAs), state machines, controllers or other hardware components or devices, or a combination of hardware, software, and firmware
20 components.

While the system and method for delivering media services has been particularly shown and described with references to embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the scope of the invention encompassed by the appended claims.

CLAIMS

What is claimed is:

1. A method of delivering video services comprising:
5 identifying a media stream having a plurality of segments, each of the segments defined by an ordered sequence of frames;
receiving a request to direct the identified media stream to a requesting user, the request received during a current segment in the stream;
redirecting the media stream to the user in a realtime manner by providing frames
10 remaining in the current segment; and
concatenating the provided frames with a successive next segment of the media stream.
2. The method of claim 1 wherein redirecting includes:
15 switching the media stream in a manner that preserves synchronization between an existing display image at a user display device and the redirected media stream; and
generating relative frames corresponding to the switched media stream for the remainder of a current segment, relative frames complementing a complete frame in the current segment.
20
3. The method of claim 2 wherein the user display device is operable to render a visual image from an encoded frame, encoding incurring greater resources than decoding.
4. The method of claim 3 wherein rendering further comprises computing a displayable
25 image from a complete frame and at least zero relative frames of the media stream.
5. The method of claim 1 wherein redirecting further comprises:
aggregating received frames in the media stream to maintain a current frame;
decoding the aggregated frames to generate a renderable frame;
30 receiving successive relative frames from the media stream;
computing the difference between the relative frame and the renderable frame; and
generating interim frames indicative of the difference between the relative frame and the renderable frame.

6. The method of claim 5 wherein receiving frames comprises receiving complete frames and relative frames, the relative frames indicative of changes from a previous frame, wherein a display image includes a complete frame and a sequence of at least zero relative frames.

5 7. The method of claim 6 wherein computing the difference further comprises:
decoding the successive relative frame;
applying the decoded relative frame to a renderable image indicative of a current
display state of the media stream; and
encoding the renderable image to generate a relative frame indicative of changes
10 between the renderable image and the current display state.

8. The method of claim 7 further comprising accumulating successive relative frames to maintain the current frame, wherein generating occurs upon receiving a request for delivery of the media stream, the request arriving in the middle of a segment.

15

9. The method of claim 1 further comprising
receiving a plurality of media streams;
maintaining a current image of each of the plurality of media streams, maintaining
including decoding a current frame of each of the received media stream;
20 switching a selected one of the plurality of media streams in response to a received
request, switching performed by rendering the current frame corresponding to the selected
media stream; and
concatenating the switched media stream to an existing user media stream without
loss of synchronization on the user display device.

25

10. The method of claim 2 wherein redirecting further comprises:
receiving each of the frames in a segment;
aggregating the frames to maintain a display frame corresponding to a display image;
generating an interim image from the display frame; and
30 receiving successive relative frames indicative of incremental changes to the display
image.

11. The method of claim 10 wherein switching further comprises:
computing a current display frame based on the successive relative frames applied to
the display image;
computing changes between the computed current display frame and the generated
5 interim image;
encoding a relative frame indicative of the computed changes, the relative frame
operable to render the interim images similar to the display image;
transmitting the encoded relative frame to a user device for rendering, rendering
operable to display the display frame including the aggregated frames in the segment.

10

12. The method of claim 2 wherein redirecting further comprises:
receiving a request to redirect the media stream to a user;
computing a current frame indicative of each of the frames in the sequence received
for the current segment;
15 immediately sending the computed current frame to the user in a seamless manner for
rendering on a user display device, the seamless manner operable to preserve synchronization
of the display image with the computed current frame;
receiving the remaining frames in the current segment;
computing, for each received frame, changes contained therein;
20 computing the difference between the computed current frame and the computed
changes; and
transmitting the computed difference to the user display device for rendering on the
user display device.

25

13. A method of switching a media stream to a requesting user in response to a request
comprising:

receiving a video switch request during a current segment following a previous I-
frame on the stream providing the requested content;

30

providing an interim I-frame corresponding to a previous I-frame and successive
series of P frames received on an incoming stream, providing frames further comprising
generating P-frames for the remainder of the current segment;

continuing providing interim P frames generated from successively received P-frames
on the incoming stream;

merging the generated steam with the incoming stream upon receiving a successive I-frame on the incoming stream

computing the difference further comprises decoding the received remaining P-frames, and

5 encoding a change frame operable to apply the differences to the user display device.

14. A video portal comprising:

an application switcher operable to identify a media stream having a plurality of segments, each of the segments defined by an ordered sequence of frames;

10 an interface to a user device operable to receive a request to direct the identified media stream to a requesting user, the request received during a current segment in the stream;

a compositor responsive to the interface operable to redirecting the media stream to the user device in a realtime manner by providing frames remaining in the current segment; and

15 a media switcher operable to concatenate the provided frames with a successive next segment of the media stream.

15. The video portal of claim 14 wherein the compositor is further operable to:

20 switch the media stream in a manner that preserves synchronization between an existing display image at a user display device and the redirected media stream; and

generate relative frames corresponding to the switched media stream for the remainder of a current segment, relative frames complementing a complete frame in the current segment.

25 16. The video portal of claim 15 wherein the interface is coupled to a user display device operable to render a visual image from an encoded frame, encoding incurring greater resources than decoding.

17. The video portal of claim 14 wherein the compositor is further operable to:

30 aggregate received frames in the media stream to maintain a current frame

decode the aggregated frames to generate a renderable frame;

receive successive relative frames from the media stream;

compute the difference between the relative frame and the renderable frame; and

generate interim frames indicative of the difference between the relative frame and the renderable frame.

18. The video portal of claim 17 wherein the application switcher is further operable to receive complete frames and relative frames, the relative frames indicative of changes from a previous frame, wherein a display image includes a complete frame and a sequence of at least zero relative frames.

19. The video portal of claim 18 wherein the compositor further comprises:
a decoder operable to decode the successive relative frame;
a current image adapted to apply the decoded relative frame to a renderable image indicative of a current display state of the media stream; and
an encoder operable to encode the renderable image to generate a relative frame indicative of changes between the renderable image and the current display state.

20. The video portal of claim 14 wherein the application switcher is operable to receive a plurality of media streams from media service applications, wherein the compositor is further operable to:

maintain a current image of each of the plurality of media streams, maintaining including decoding a current frame of each of the received media stream;
switch a selected one of the plurality of media streams in response to a received request, switching performed by rendering the current frame corresponding to the selected media stream; and
invoke the media switcher to concatenate the switched media stream to an existing user media stream without loss of synchronization on the user display device.

21. The video portal of claim 20 wherein the compositor is further operable to, for each of the media frames:

receive each of the frames in a segment;
aggregate the frames to maintain a display frame corresponding to a display image;
generate an interim image from the display frame; and
receive successive relative frames indicative of incremental changes to the display image.

22. A computer program product having a computer readable medium operable to store computer program logic embodied in computer program code encoded thereon for delivering video services comprising:

computer program code for receiving a video switch request during a current segment following a previous I-frame on the stream providing the requested content;

computer program code for switching the media stream in a manner that preserves synchronization between an existing display image at a user display device and the redirected media stream;

computer program code for immediately sending the computed current frame to the user in a seamless manner for rendering on a user display device;

computer program code for generating interim I-frames corresponding to the switched media stream for the remainder of a current segment, relative frames complementing a complete frame in the current segment;

computer program code for transmitting each of the interim I frames corresponding to a previous I-frame and successive series of P-frames received on an incoming stream, providing frames further comprising generating P-frames for the remainder of the current segment;

computer program code for continuing providing interim P-frames generated from successively received P-frames on the incoming stream;

computer program code for merging the generated steam with the incoming stream upon receiving a successive I frame on the incoming stream;

computer program code for computing the difference further comprises decoding the received remaining frame, and

computer program code for encoding a change frame operable to apply the differences to the user display device.

23. A video enabled personal communications device comprising responsive to a user request for switching a media stream comprising:

means for identifying a media stream having a plurality of segments, each of the segments defined by an ordered sequence of frames

means for receiving a request to direct the identified media stream from a content processor to a requesting user, the request received during a current segment;

means for computing a current frame indicative of each of the frames in the sequence

received for the current segment;

means for immediately sending the computed current frame to the user in a seamless manner for rendering on a user display device, the seamless manner operable to preserve synchronization of the display image with the computed current frame;

5 means for redirecting the media stream to the user in a realtime manner by providing frames remaining in the current segment; and

means for concatenating the provided frames with the next segment of the media stream.

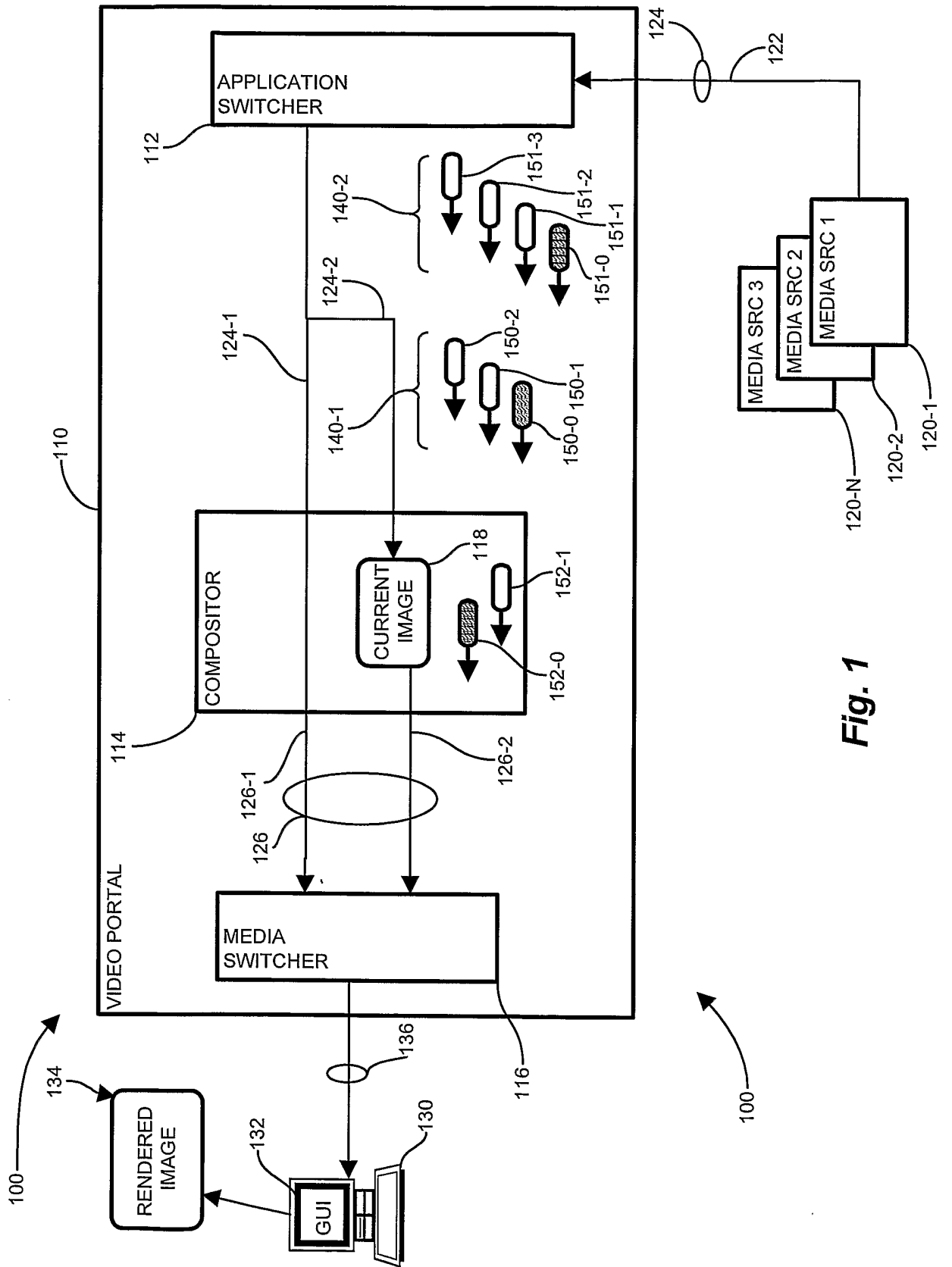
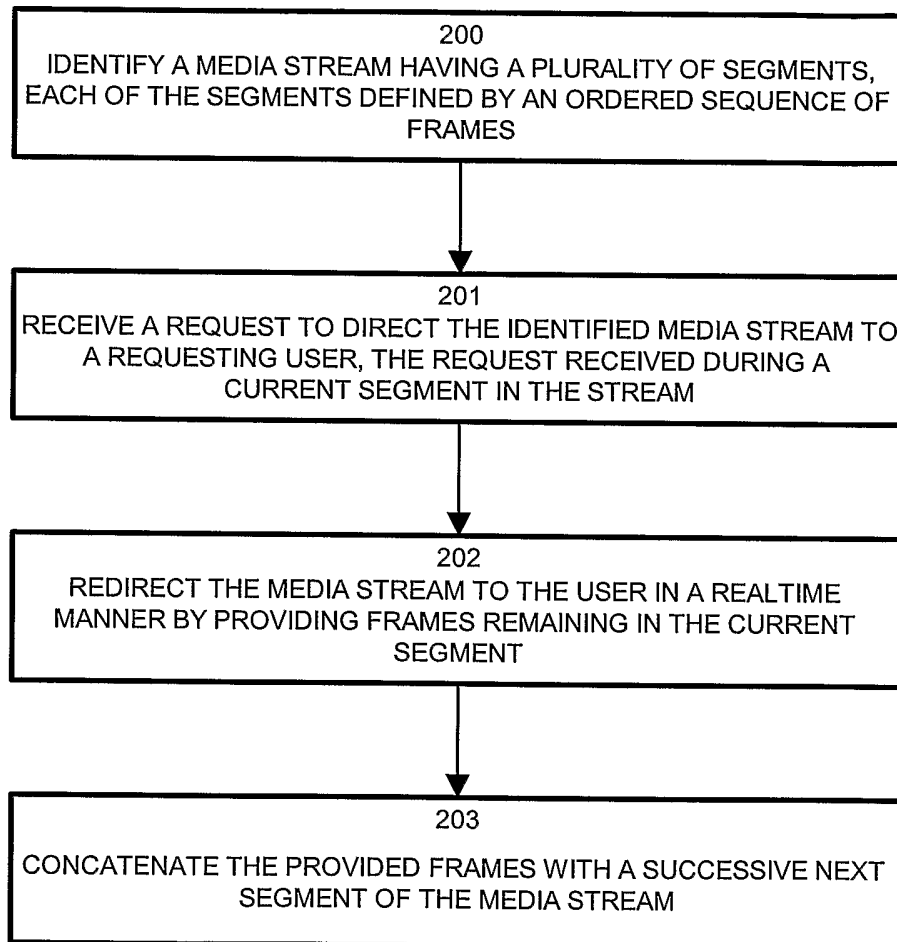


Fig. 1

**Fig. 2**

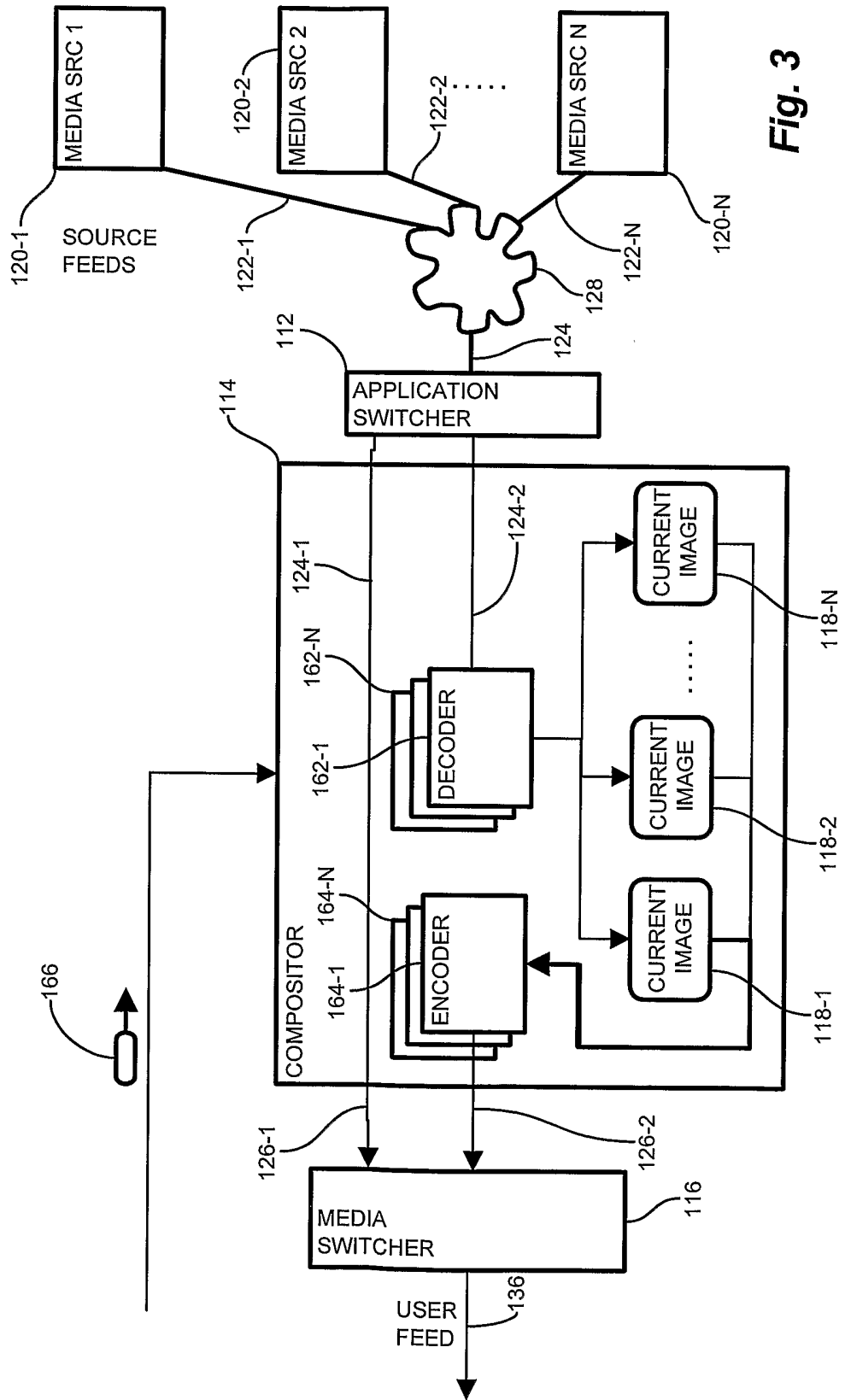


Fig. 3

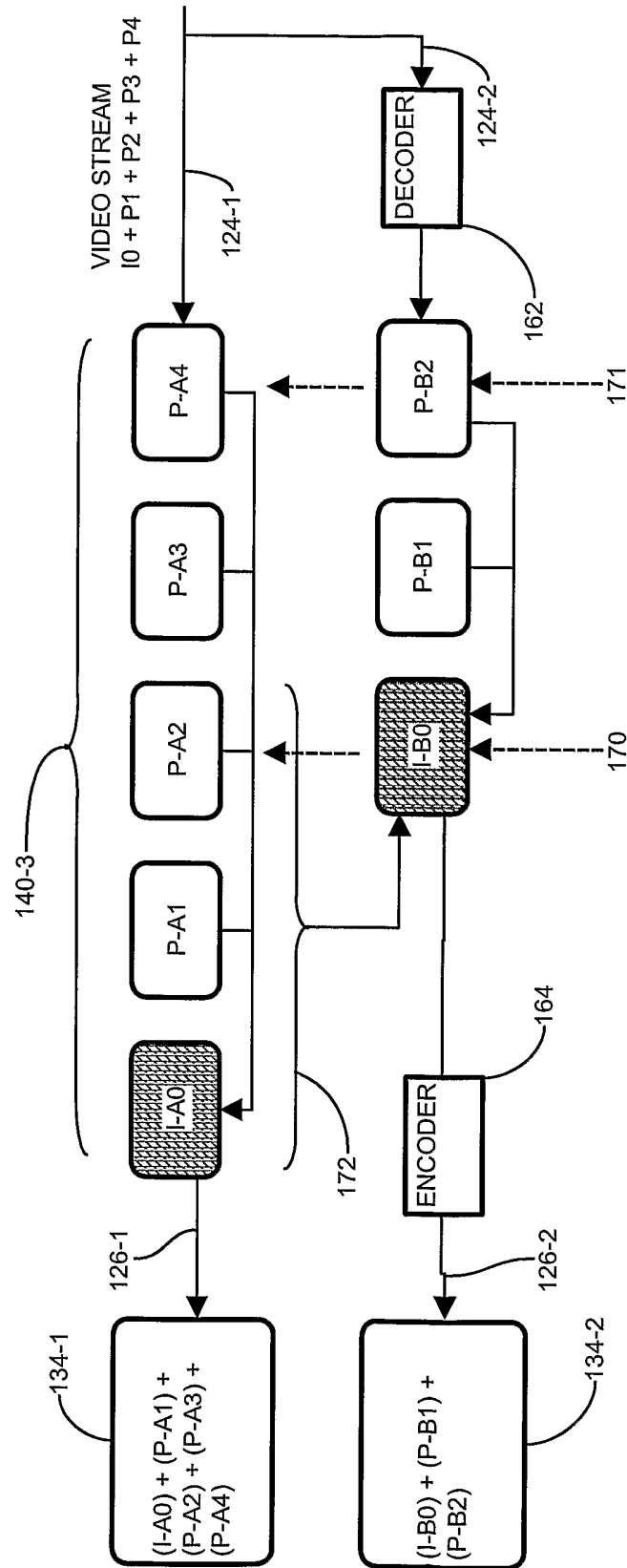


Fig. 4

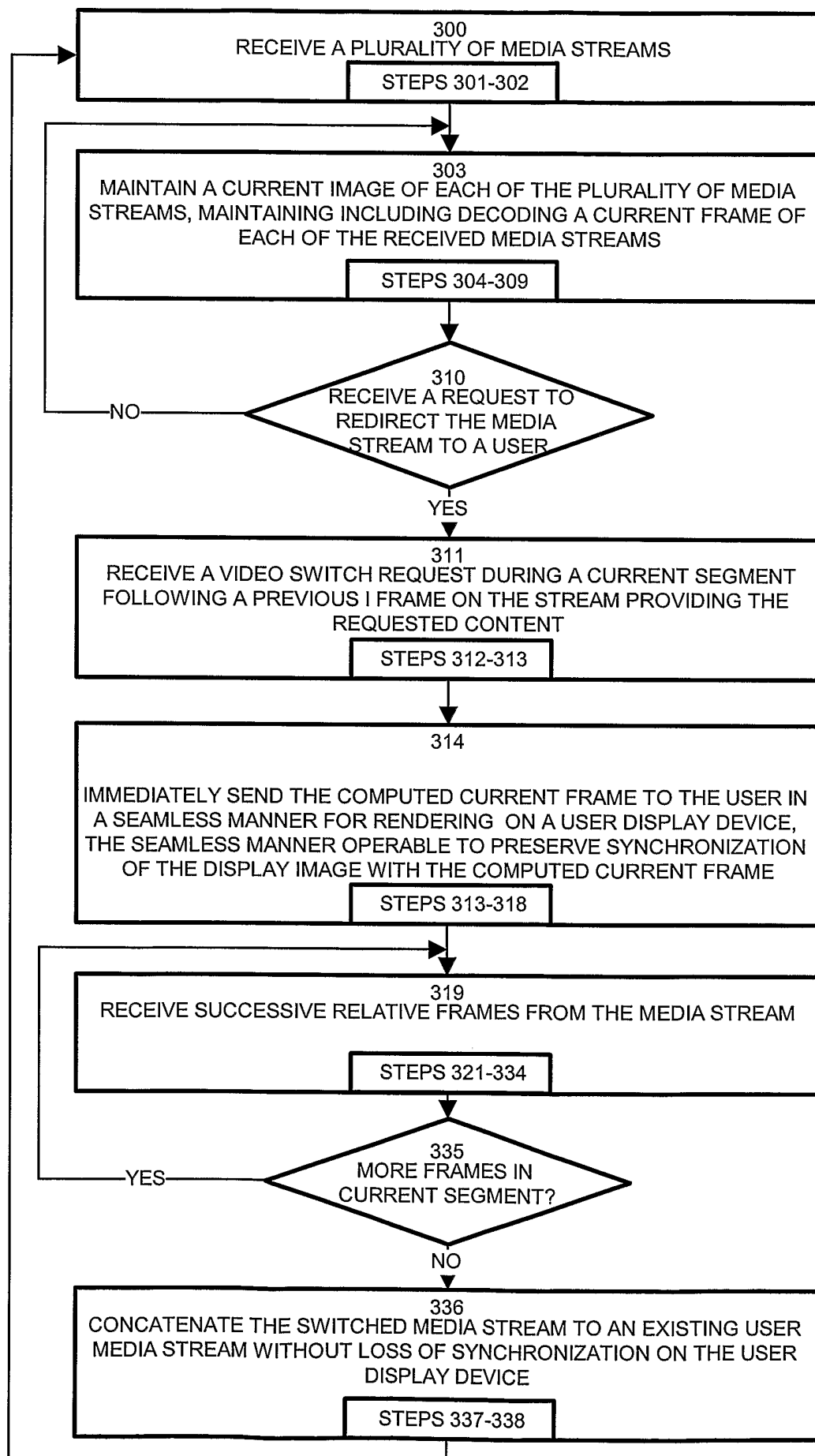


Fig. 5

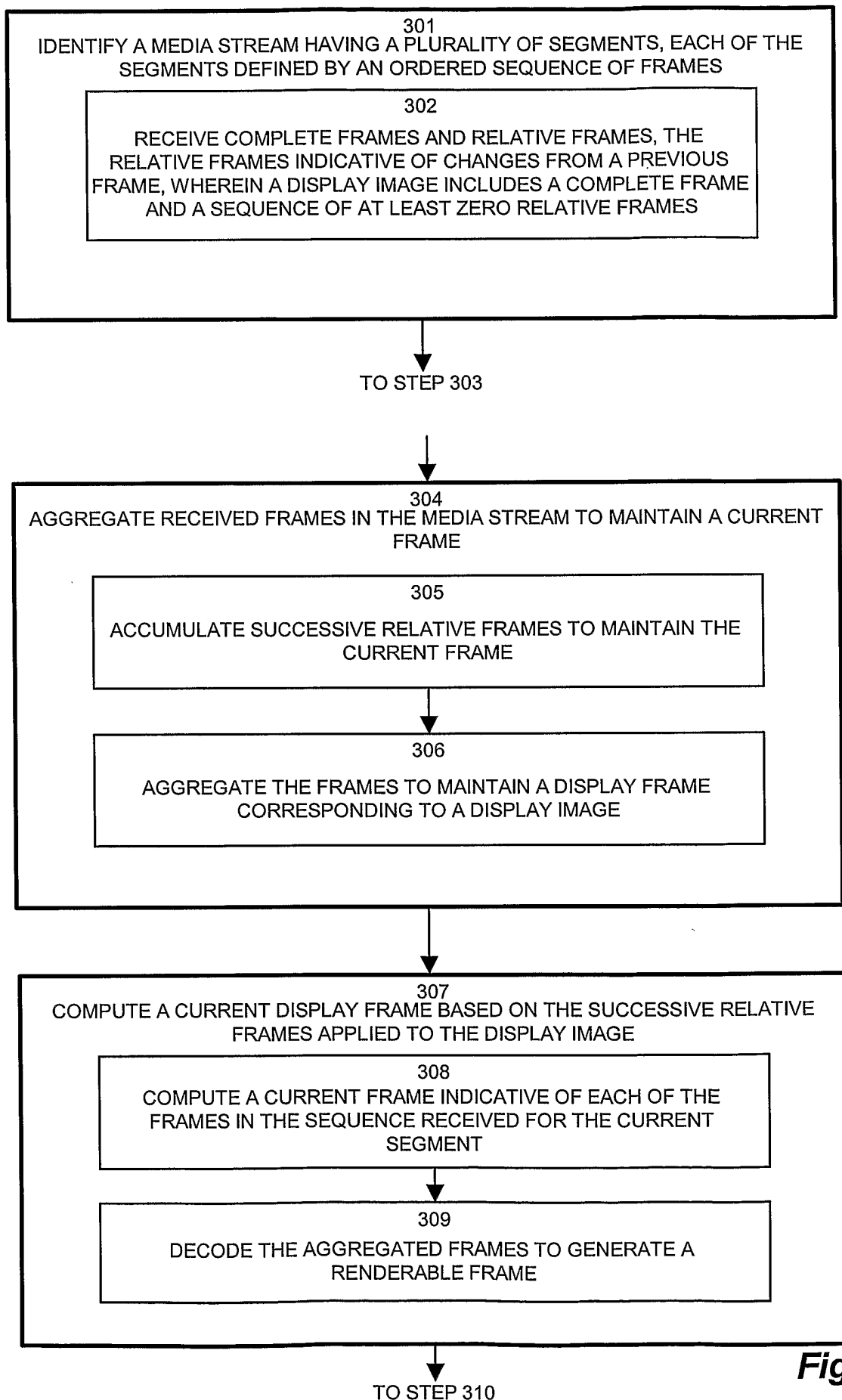
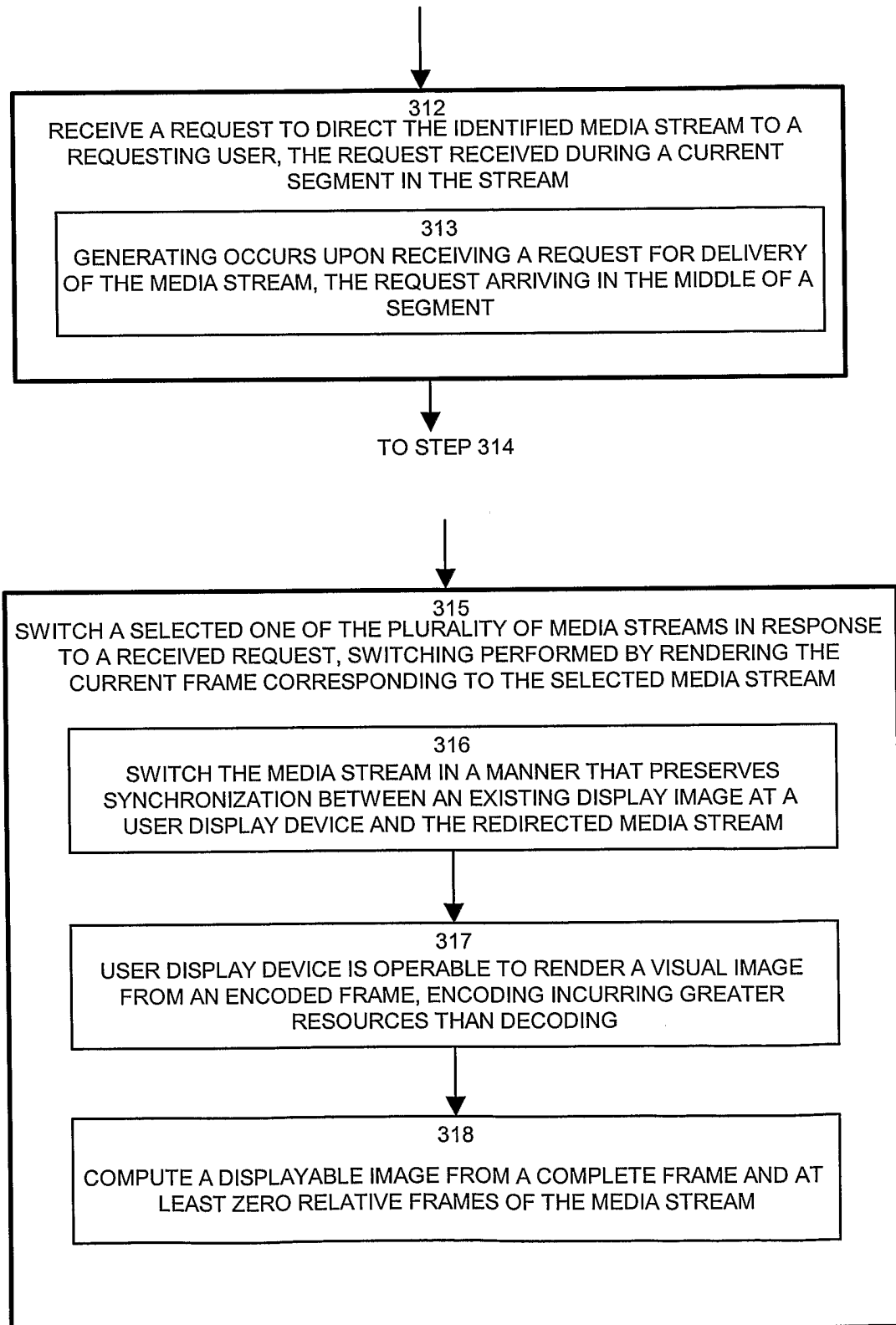


Fig. 6

**Fig. 7**

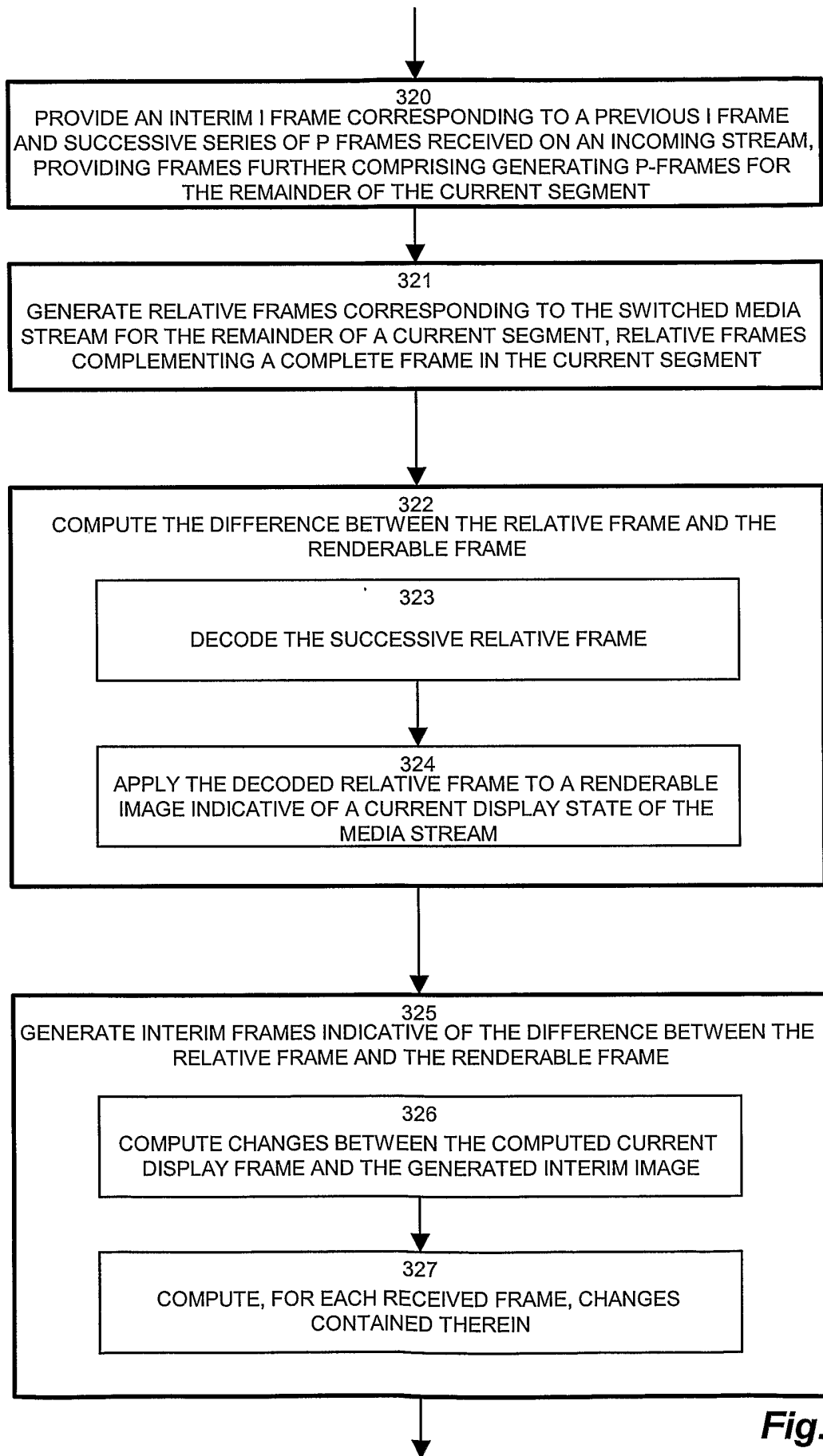


Fig. 8

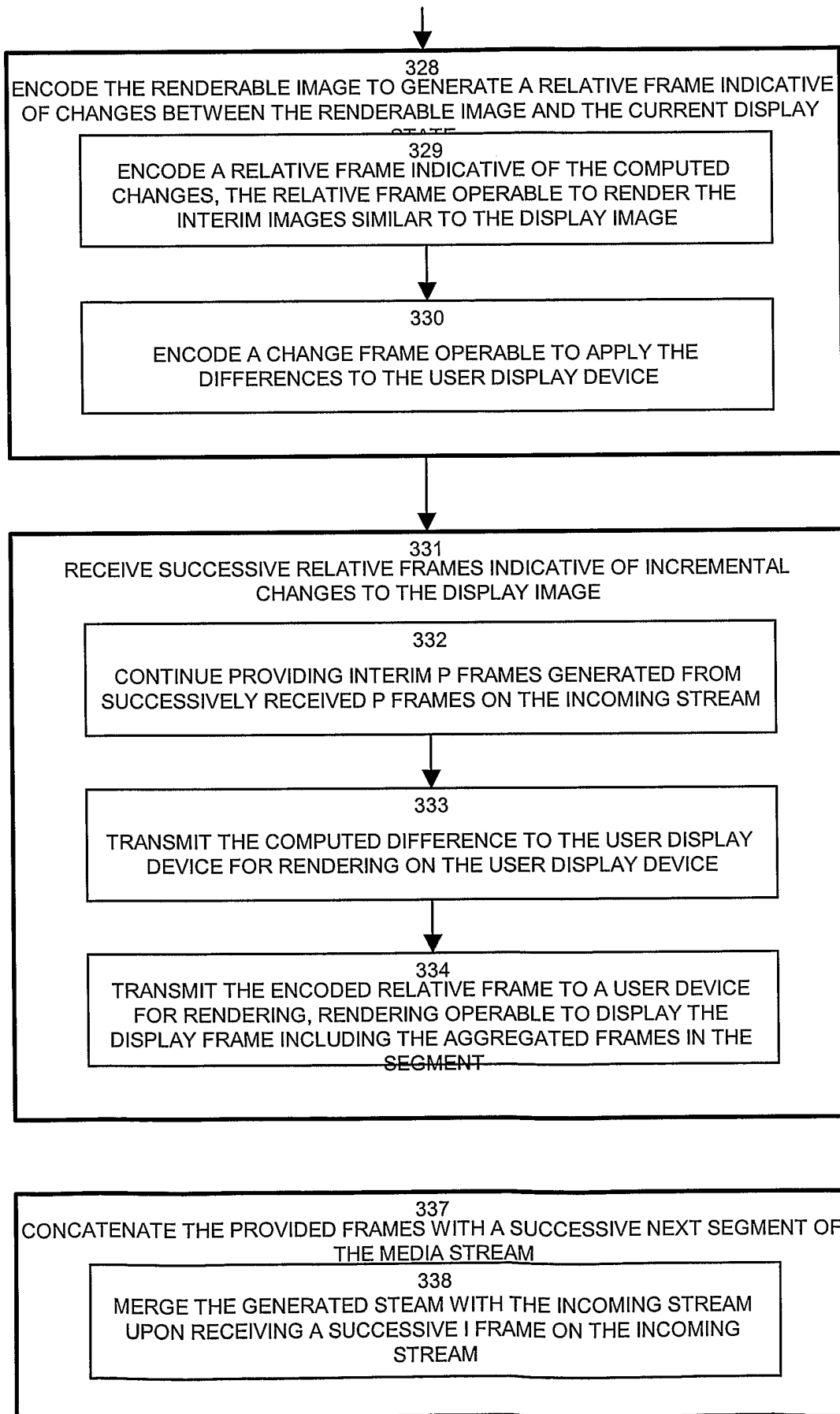


Fig. 9