



- (51) International Patent Classification: **H03M 7/40** (2006.01)
- (21) International Application Number: PCT/US2016/029890
- (22) International Filing Date: 28 April 2016 (28.04.2016)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data: 62/175,232 13 June 2015 (13.06.2015) US
- (71) Applicant: **HOTPATHZ, INC.** [US/US]; 29 Taff Comers Ctr. Ste. 160, Williston, VT 05495 (US).
- (72) Inventor: **MONAHAN, Jay**; 29 Taff Comers Ctr. Ste. 160, Williston, VT 05495 (US).
- (74) Agent: **MCCABE, Justin, W.**; Dunkiel Saunders Elliott Raudvogel & Hand PLLC, 91 College St., Burlington, VT 05401 (US).
- (81) Designated States (*unless otherwise indicated, for every kind of national protection available*): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU,

[Continued on next page]

(54) Title: SYSTEMS AND METHODS FOR EMBEDDING USER INTERACTABILITY INTO A VIDEO

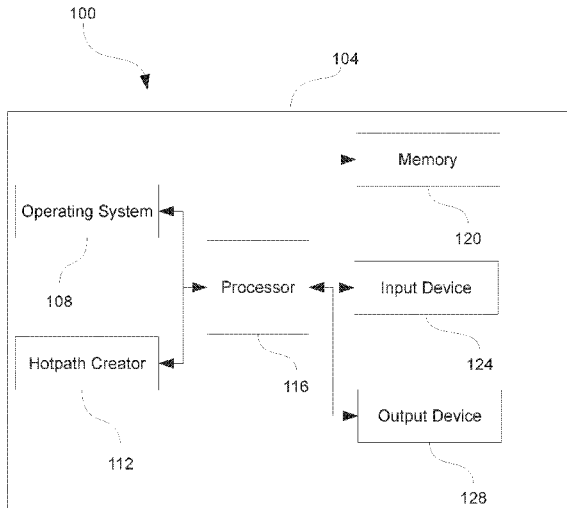
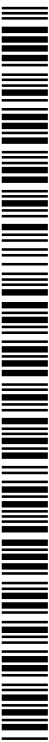


FIG. 1

(57) Abstract: Systems and methods disclosed herein embed a hotpath data stream within a video feed. The system and method can define a data micro-format for "moveable clickable areas" (MCAs) for any desired object within a video and then embedding this data into either: a) a data track when using a video container format that supports arbitrary metadata tracks, b) the subtitle data stream, c) the video data stream, d) the audio data stream, or combinations of the same. The MCA contains information about the object, including a dataset regarding the MCA's size and relative location as well as other information, such as, but not limited to, a hyperlink, a text box, an image, a scoring algorithm, and an expectant motion.



LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, **Published:**
SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, — *with international search report (Art. 21(3))*
GW, KM, ML, MR, NE, SN, TD, TG).

SYSTEMS AND METHODS FOR EMBEDDING USER INTERACTABILITY INTO A VIDEO

RELATED APPLICATION DATA

[0001] This application claims priority to U.S. Provisional Application No. 62/175,232, filed June 13, 2015, and titled “Systems And Methods For Embedding User Interactability Into A Video”, which is incorporated herein in its entirety.

FIELD OF THE INVENTION

[0002] The present invention generally relates to video streaming. In particular, the present invention is directed to Systems and Methods for Embedding Information into a Video.

BACKGROUND

[0003] The digital world is rapidly evolving from the display of flat, static information (e.g., websites) to dynamic video display of moving information in two and three dimensional space. As the world moves this direction, there is a need to create new ways to interact with video because video, and especially online video, is a very useful communication tool—for social interactions as well as for education, training, and other purposes. While many have continued to differentiate between delivered video, e.g., television delivered via a cable wire or via satellite, and online video delivered via an internet connection, in truth the two are merging and essentially becoming one in the same. Thus, users can interact with their delivered video in many of the same ways they interact with online videos (e.g., play on demand, etc.) such that users are beginning to expect certain functionality to be co-delivered with the video—functionality such as clickable hyperlinks within the video frame, external links that lead the viewer to a specific website or other internet address, pop-outs that display information or images, or playing video or audio alongside of or instead of the video previously being played.

[0004] With reference specifically to hyperlinks, they are widely used in electronic text documents and typically implemented with web browsers. A hyperlink may be considered as a connection between an element, such as a word, phrase, symbol or object in a document, such as a hypertext document, with a different element in the same document, another document, file or script. The hyperlink may be activated by a user clicking or otherwise selecting the hyperlink. When the user clicks on the hyperlink, the browser may be redirected to the element or other document. The concept of hyperlink may also be used on images, particularly as a “map” tag on images in hypertext

markup language. For example, when a user clicks on a region having the map tag, the browser is redirected to the linked webpage.

[0005] There are prior art attempts to embed certain information, such as hyperlinks, in a video stream. Some have focused on approaches that use a separate file that can be delivered alongside a video stream, the separate file containing a database of objects of interest, tracked by time, or video frame (a number in a sequence of video frames) and rectangular Cartesian coordinate system. Current video compression algorithms such as H.266 or VP9, create significant problems for this approach as the time and/or the database frame location of the object is often different from the actual location in the video bitstream – particularly in cases where the object of interest is moving rapidly or moves sideways in the video frame. In practice, this means that the hyperlink is often significantly misaligned with the object of interest upon playback. Additionally, synchronizing a separate data file that contains a hyperlink and rectangular coordinate location of the object of interest, and that is not directly associated with or integrated with the video file, as the video file streams in real time, is fraught with potential for missed timing and misalignment of the data to the object of interest, thus creating a poor user experience.

SUMMARY OF THE INVENTION

[0006] In an exemplary embodiment, a method of allowing a user to interact with a video stream is disclosed, the method comprising: accessing the video stream, the video stream including a plurality of objects; developing a hotpath data stream, the hotpath data stream including a plurality of moveable clickable areas, wherein each of the plurality of moveable clickable areas are associated with a corresponding respective one of the plurality of objects; embedding the hotpath data stream with the video stream in a file container; decoding the file container such that the user can interact with ones of the plurality of objects via its corresponding respective one of the plurality of moveable clickable areas.

[0007] In another exemplary embodiment a system for allowing a user to retrieve information related to objects found in a video stream is disclosed, the system comprising: a computing device, the computing device including a processor having a set of instructions, the set of instructions configured to: access the video stream, the video stream including a plurality of objects; develop a hotpath data stream, the hotpath data stream including information and a moveable clickable area associated with each of the plurality of objects; embed the hotpath data stream with the video stream in a file container.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] For the purpose of illustrating the invention, the drawings show aspects of one or more embodiments of the invention. However, it should be understood that the present invention is not limited to the precise arrangements and instrumentalities shown in the drawings, wherein:

FIG. 1 is a block diagram of an embedding system according to an embodiment of the present invention;

FIG. 2 is an illustration of the development of a hotpath data stream on a frame by frame basis;

FIG. 3 is a block diagram of an exemplary process of embedding data in a video stream according to an embodiment of the present invention;

FIG. 4 is a block diagram of the build of a video container or “wrapper” that incorporates hotpath data according to an embodiment of the present invention;

FIG. 5 is another block diagram of the build of a video container that incorporates a hotpath data stream according to an embodiment of the present invention;

FIG. 6 is an image of an embedded a hotpath data stream within a video frame using an unused variable length code according to an embodiment of the present invention;

FIG. 7 is a table showing the coding for embedding a hotpath data stream in an audio file; and

FIG. 8 is a block diagram of a computing environment suitable for use with systems and methods of the present invention.

[0009] At a high level, systems and methods for embedding information in a video stream according to the present disclosure embeds a hotpath data stream within a video feed. The system and method can define a data micro-format for “moveable clickable areas” (MCAs) for any desired object within a video and then embedding this data into either: a) a data track when using a video container format that supports arbitrary metadata tracks, b) the subtitle data stream, c) the video data stream, d) the audio data stream, or combinations of the same. The MCA contains information about the object, including a dataset regarding the MCA’s size and relative location as well as other information, such as, but not limited to, a hyperlink, a text box, an image, a scoring algorithm, and an expectant motion. The system allows the data to be carried with the video instead of in a separate database file, the data is more easily matched with the object(s) of interest in the video, and a larger amount of data can be put into the video (data associated with a single object or many objects) with no degradation in video loading or streaming speed.

[0010] Turning now to the figures, and specifically with reference to FIG. 1, there is shown an exemplary system 100 (FIG. 1) that facilitates embedding information, such as, but not limited to, hyperlinks and MCAs within a video stream. At a high level, system 100 includes a computer 104 that generally has, among other possible components, an operating system 108, a hotpath creator 112, a processor 116, a memory 120, an input device 124, and an output device 128.

[0011] In general, computer 104 may be one of various computing or entertainment devices, such as, but not limited to, a personal computer, personal digital assistant, a set top box (e.g., Web TV, Internet Protocol TV, etc.), a mobile device (e.g., a smartphone, a tablet, etc.), and a system 700 (shown and described in FIG. 7). Computer 104 may include additional applications aside from operating system 108 and hotpath creator 112. For example, computer 104 can include applications that record, play back, and store video streams, such as, but not limited to, the VLC media player, by VideoLAN Organization of Paris, France. A browser program also may be used with or without other applications to play back video streams or to facilitate access to one or more embedded piece of information contained in the video, e.g., hyperlinks, redirection to websites, etc.

[0012] Input device 124 and output device 128 may be included with computer 104 so as to facilitate the receipt of video streams from external sources. Input device 124 and output device 128 may include particular encoders, decoders, and communication interfaces (e.g., wired, wireless, etc.) that support various communication protocols, transportation protocols, and other protocols used to receive the video stream and other data. Input device 124 and output device 128 may also further provide functionality to send other data and information. In an exemplary embodiment, output device 128 is a display that is capable of showing a video stream with MCAs associated to particular objects in the video stream.

[0013] With reference to FIGS. 1 and 2, hotpath creator 112 may implement use of a display 120 so that a user can identify or select an object or objects in a series of frames and associate information, such as a hyperlink, to the object(s), as further discussed below. For example, and as shown in FIG. 2, multiple objects 132, object 132A (a car) and 132B (a soccer ball), appear in a series of frames 136 (e.g., 136A-D). Each object 132 has an area 140 (e.g., 140A and 140B) associated with the object. A user could, for example, associate information, such as the type of object 132, projected path of the object, etc., with each the object, by selecting each object at each frame 136.

[0014] At a high level, a hotpath data stream is a series of two or more data sets that include the location (defined herein as at least a time and position), other desired information, and a size of a moving clickable area (MCA) (MCA may be two or three dimensional and take on most any desired shape) that is associated with and is positioned around or over an object of interest in the video. At a minimum there is one data set for the start and one data set for the ending time of each object that is desired to be tracked in a video frame sequence. However, in a preferred embodiment, a hotpath data stream includes more than two data sets related to each object of interest during the time the object appears in the video because additional data sets allow for the MCA shape to move through a series of locations with more accurate fluid movement, MCA resizing, and for the MCA to follow the objects on non-trivial paths. The data set is also configured to allow for multiple types of movement, e.g., linear, curvilinear, spline, etc.

[0015] During the hotpath data stream development, hotpath creator 112 acquires an array of object related information, e.g., locations, MCAs, other information, by monitoring the user's interactions with objects in the video. In an exemplary embodiment, the video is stopped or paused by the user, automatically upon the appearance of new object, or on a scheduled basis, e.g., every 10 frames, so as allow the user to select desired object(s) and to associate information and an MCA with the object(s). For objects that have already been selected and associated with an MCA, a user can reposition and resize the MCA as desired so as to provide for better tracking of the object. At the first selection of the object or at subsequent alterations to the MCA associated with the object, the location is recorded, and any additional content the user desires to associate with the object can be added or removed. During playback, the MCA can be moved or modified during which hotpath creator 112 records the movement of the MCA as a new dataset for that object. In another exemplary embodiment, the user can move the video to the desired playback time and reposition and resize the MCA.

[0016] In another exemplary embodiment, an automated object tracking system allows the user to follow and resize desired objects. In an embodiment of an automatic object tracking system, a software program, such as MatLab[®], produced by Mathworks, Inc. of Natick, MA, can be used to detect the moving objects in each video frame, follow the objects over time, and provide a hotpath data stream from the information collected.

[0017] The desired path accuracy of the MCA for each object is dependent upon, among other things, the frequency by which a user, or an algorithmic hotpath data set generation program,

specifies the location and size of the MCA. Thus, each MCA may have a few locations per second, e.g., 10 locations per second. In certain situations, for example, if the object is moving in a relatively straight line, only datasets at an object's appearing time and the moment before the object disappears need be recorded as the MCA can be programmed to interpolate the linear path between its two positions. In other embodiments, the MCA can be programmed to follow other types of paths between the locations, e.g., curvilinear.

[0018] The data sets developed by hotpath creator 112 can be configured to work with the same video at different resolutions, and as such, the values used to indicate the position and size of each MCA are, in a preferred embodiment, not absolute pixel locations, but can also be relative numbers that correspond to percentages relating to the video's frame size.

[0019] In an exemplary embodiment, the data set format is flexible and expandable so as to allow for additional attributes of each object to be associated with the object and that can be incorporated into multiple types of existing metadata or subtitle mechanisms of the various video file formats, available now, or in the future. For example, a suitable data set format can be an extendable JSON-style text format. In an exemplary embodiment, a text-based, JSON-style comma-separated name/value pairs format is used. In this embodiment, an example of JSON-style data format for two objects, e.g., Object One and Object Two, in a rectangular coordinate space would be as follows:

[0020] Data Set Object One: { "url": "http://www.somewhere.com/", "locations": [{ "time": "00:01:21:00", "x": 15.3, "y": 46.2, "width": 24.5, "height": 18.7 }, { "time": "00:01:33:00", "x": 19.8, "y": 55.2, "width": 28.1, "height": 22.6 }] }

[0021] Data Set Two { "url": "http://www.otherplace.com/", "note": "some note here", "locations": [{ "time": "00:02:45:00", "x": 75.1, "y": 56.7, "width": 32.8, "height": 21.4 }, { "time": "00:03:26:00", "x": 64.9, "y": 91.5, "width": 12.8, "height": 15.2 }, { "time": "00:03:48:00", "x": 75.1, "y": 88.2, "width": 11.9, "height": 14.6 }] }

[0022] As set forth above, Object One has a first data set that includes a hyperlink, e.g., <http://www.somewhere.com/>, and defines a first MCA which starts at time 1:21 at position 15.3, 46.2 with size 24.5 x 18.7, and moves to time 1:33 and position 19.8, 55.2 with size 28.1 x 22.6. Object Two has three data sets in contrast to Object One's two data sets. Object Two includes a hyperlink, e.g., <http://www.otherplace.com/>, and moves from point A to B to C during its considerably longer lifetime (in this case from time 2:45 to time 3:48). In this example, data sets

associated with Object Two include an additional attribute, i.e., a “note”. The data format used is flexible in that it can add any number of additional attributes to the hotpath data stream. For example, if added support for curved motion paths is desired (as opposed to linear projections), the data set can include an attribute that would result in the MCA’s motion being curvilinear between locations. Similarly, the shape of an MCA can be modified to be customized to the user’s desires, such as, but not limited to, ovals, rectangles, stars, and human-shaped.

[0023] Although data space is generally always a concern, there are methodologies to reduce the hotpath data stream size, such as, but not limited to, leaving out the quotes around most of the attributes, avoiding spaces, using 't' instead of 'time' and 'w/h' instead of 'width/height' etc. Ultimately, compression is not generally needed, since the size of the hotpath data stream compared to the video data size is insignificant, even if there are thousands of objects being tracked in the video.

[0024] Hotpaths data streams are particularly useful for video applications that require a large degree of interaction with the user. Current applications include, but are not limited to, a driver training app wherein the user must notice and indicate they have seen, in the correct sequence, a variety of items included in a film of an actual drive in a congested area; a quarterback training app wherein the user must recognize and indicate they have seen the intended movements of key defensive players pre-snap and then indicate the choices available to them post-snap in a film of an actual scrimmage; a law enforcement training app wherein the user must recognize and indicate they have properly assessed a variety of threats contained in a film of a crowd; a baseball training app wherein the user must recognize and indicate their understanding of signals from the batting coach, first base coach, etc. in a film of an actual at-bat situation; a physical therapy educational app, wherein the user must recognize and suggest treatments based on key characteristics of the gait of a patient videotaped in a clinic; a virtual surgery training app wherein a surgeon can select and interact with various elements in a video of a surgery performed by someone else; and an aircraft maintenance training app wherein the user must indicate an understanding of the location, sequence, and priority of items requiring attention.

[0025] Turning now to FIG. 3, there is shown an exemplary method 200 for the embedding of a hotpath data stream within a video stream. At a high level, method 200 allows for the inclusion of data and information that relates to objects found in video stream and for the retrieval of this data and information by a user. In exemplary embodiments, method 200 combines multiple data streams

without significantly impacting playback speed or recording/playback quality by selective combining the hotpath data stream with the other streams or in lieu of other streams. Additionally, by embedding the hotpath data stream within the video stream, the two streams can be tightly coupled to each other, providing consistent object / hotpath registration and tracking.

[0026] At step 204, a video stream is accessed by the user. The video stream can be stored or recorded locally, retrieved from an external server or other storage device, or can be retrieved or produced via methods known in the art. The video stream can include, among other things, images, still frames, audio, or combinations of the same. In exemplary embodiments, the video stream is a prerecorded scene that depicts a desired setting for training an end user of the video stream with the embedded hotpath data stream.

[0027] At step 208, a hotpath data stream is developed. In an exemplary embodiment, a hotpath data stream is developed by a hotpath creator, such as hotpath creator 112 described above with reference to FIG. 1. In an exemplary embodiment, hotpath creator includes an automated object tracking system, such as the system described herein.

[0028] At step 212, the hotpath data stream developed at step 208 is associated/embedded within the video stream. In an exemplary embodiment, the hotpath data stream is placed into a multimedia file container via one of the association/embedding methodologies discussed in more detail below with reference to FIGS. 3 and 4.

[0029] As shown in FIG. 4, a multimedia file container 300, which can also be called a wrapper, “wraps up”, for example, a video stream 304, an audio stream 308, a subtitle stream 312, and a hotpath stream 316 (each of the aforementioned can also be called a “track”) into a chronological, time-linear, single delivery stream or file by including, with the streams, a header 320 that includes data regarding how each of the streams is to be treated upon display to a user. In an exemplary embodiment, file container 300 encapsulates each stream and allows for the interleaving of audio, video, and other data inside a single package. In this embodiment, the use of header 320 allows file container 300 to administer overhead tasks such as packet framing, ordering, interleaving, error detection, and periodic timestamps for seeking video, audio, or subtitle information.

[0030] Typically, and as shown in FIG. 5, prior to the creation of the file container, each stream is compressed and coded by an encoder or codec. The codec encodes the stream on the originating end (during file container creation) and decodes it on the receiving end (during display). In general,

a codec describes how video or audio data is to be compressed and decompressed. Codecs are traditionally licensed exclusively to a certain format. For example, the WMV video codec is only used in Windows Media containers. File container types include, but are not limited to, QuickTime[®] (a video player from Apple, Inc. of Cupertino, CA) (.MOV), Matroska, AVI, WMV, and MPEG-4. Common codecs (audio, video, subtitle, etc.) include, but are not limited to, MPEG-4, H.264, DivX, AAC, Vorbis (audio codec), and SRT. Some file containers, including, but not limited to, OGG and FFmpeg, are multimedia container formats, meaning they can contain additional streams beyond the traditional video, audio, and subtitle streams. As shown in FIG. 5 and with continued reference to FIG. 4, audio stream 308 may be delivered to audio encoder 408, video stream 304 may be delivered to video encoder 404, subtitle stream 312 may be delivered to subtitle encoder 412, and hotpath stream 316 may be delivered to hotpath encoder 416.

[0031] As shown in FIG. 5, after the hotpath stream has been encoded, the encoded stream can be included in the file container via several different embedding options, including, metadata embedding 420, subtitle embedding 424, video embedding 424, and audio embedding 428.

[0032] In an exemplary embodiment of metadata embedding 420, a multimedia container format is chosen that supports arbitrary time-based metadata tracks. Some available container formats meeting these criteria include, but are not limited to, MOV, WebM, and OGG. In this embodiment, when playback is desired, an appropriate decoding application, such as VLC media player (mentioned above) that includes the appropriate codecs (including a hotpath codec), would be used to play back the video stream, audio stream, subtitle stream (if used), and hotpath data stream. The hotpath codec specifies the format of the hotpath data stream on the creating end (as discussed above) and provides information on the interpretation, display, and actions of the hotpath data stream on the receiving end. In an exemplary embodiment, the file container developed can also include metadata that specifies how the data streams inside the file container are organized and which codecs are necessary so as to decipher the data streams.

[0033] In an alternative embodiment, the encoded hotpath data stream can be embedded using subtitle embedding 428, which treats the hotpath data stream similar to a subtitle track, which is supported by many video container file formats, such as, but not limited to, MP4 and MKV. In this embodiment, the hotpath data stream is embedded and multiplexed (interleaved) into the video file along with the video and audio tracks instead of one of the subtitle tracks. Thus, rather than subtitles showing an alternative language, e.g., French, Spanish, the hotpath data stream is presented. In a

preferred embodiment, a multimedia file header, such as header 320 (FIG. 4) includes information to alert the playback device to use the hotpath data stream contained in (one of) the subtitle data streams. This approach is generally compatible with many video container formats, and is useful for video container formats that do not support arbitrary metadata tracks (i.e., would not support metadata embedding 420). During playback, an appropriate decoding application would be used to appropriately play back the video stream with the embedded hotpath data stream (now substituted for the subtitle track).

[0034] In yet another embodiment, in the instance where the desired video container format does not support the addition of arbitrary metadata tracks or subtitle tracks, or if the user wishes to preserve the subtitle track(s) for their intended use, the hotpath data stream can alternatively be encoded in either the video or audio data streams using either video embedding 428 or audio embedding 432.

[0035] Video embedding 428 entails encoded the hotpath data stream onto the video stream by re-purposing unused variable length codes (VLC) present in the video stream. To understand this approach, and with reference to FIG. 6, the video stream can be modeled as a linear sequence of JPEG images displayed one after another at a video rate. (This analogy does not take into account the excessive redundancy from frame to frame. MPEG takes advantage of this redundancy to further compressive the file. Nevertheless, the analogy works for illustration purposes.) Although the JPEG standard (a commonly used method of lossy compression for digital images) defines 162 different VLCs for alternating component (AC) coefficients, many codes are not used during image compression. By statistically determining VLC usage, unused codes can be found and repurposed to store hotpath data. In an exemplary embodiment of video embedding 428, to encode the hotpath data stream into the video data stream, a map of “unused” VLCs is developed by frame number and position within the frame, the hotpath data stream file is compressed, and sequentially inserted into the previously mapped unused VLCs. The presence of repurposed VLCs and the code mapping relationships to the hotpath data stream can be stored in the header of the multimedia container. This method of embedding permits file-sized hotpath data streams to be inserted into in a compressed video domain such as MPEG-4 or H.256, with little quality distortion. In an exemplary embodiment, the hotpath data stream is encoded using Huffman coding (a lossless data compression algorithm) and embedded in the “unused” VLCs that have been previously mapped/identified. In general, decoding includes two primary steps: 1) VLC mapping reconstruction using the mapping relationship information contained in the video data stream header, and 2) decompression of the

hotpath data stream using a decoding table contained in the video container header. To facilitate decoding, a Huffman table (also called Huffman tree), which can be used to decompress the hotpath data stream, is stored in the multimedia file container header, such as header 320 (FIG. 4). The file container header allows the decoder to sequentially reconstitute the hotpath data stream from the repurposed VLCs header. In exemplary embodiment, a Huffman table is read from the file header. The presence of this table allows for the examination of the video data stream header. In this embodiment, the file header includes information on where in the video stream to look for VLCs. The implementation time of this approach is comparable to the previously discussed embodiments where a separate hotpath track is included in the file container.

[0036] A visual representation of video embedding 428 is shown in FIG. 6, which shows an illustration of a video frame 500, including an object 504 and a VLC 508. As depicted in FIG. 6, as VLC 508 is encountered in the video, it is decompressed or expanded using the Huffman table. A hotpath data stream 512 is reconstituted from VLC 508 thus providing the data related to object 504, for example, the size and shape of the MCA, a note, an object type, etc..

[0037] Returning to FIG. 5 and with reference to FIG. 6, another option for embedding a hotpath data stream is using audio embedding 432, so that the hotpath data stream is embedded in an audio stream, such as, but not limited to, AAC, that is included in the multimedia container. Techniques for embedding information in digital audio include, but are not limited to, parity coding, phase coding, spread spectrum, echo hiding, and least significant bit (LSB) insertion. In an exemplary embodiment, LSB is used to embed a hotpath data stream in the left or right audio channel of a video/audio recording. With this technique, the LSB of binary sequences of each sample of a digital audio file is replaced with the part of the hotpath information.

[0038] As shown in FIG. 7, as an example of use of LSB to embed the hotpath data stream in the audio stream, the letter “D” is embedded in the last bit of the audio code, which has an ASCII code of 68, or 01000100, inside eight bytes of an audio file. In column 604 there is shown the original audio code. In column 608 there is shown the exemplary text to embed (e.g., the letter “D”). In column 612 there is shown the embedded text (far right number of each row). The LSB coding method can inject a low level of noise into the audio track (~ 6bB/bit), which may impact some uses. However, for certain videos, e.g., those with road noise or other significant background noise, more than 1 bit may be used.

[0039] For other applications where high quality sound is desired, variations of LSB coding can be employed such as the zigzag LSB coding method where information is inserted into the last bit of the audio in a zigzag fashion. In this embodiment, on average, only half of the bits are used thereby maintaining higher audio quality. As with the previously described method of embedding the hotpath information into the video data stream, the header(s) within the audio stream are used to signal the presence of embedded hotpath stream.

[0040] Regardless of whether the hotpath information is contained in a separate hotpath data stream, subtitle stream, video stream, or audio stream, because the hotpath data stream is contained in the multimedia file container and any file compression or interleaving is synchronized with the video stream, the hotpath data stream can be readily synchronized to objects and events occurring in the video. As the video is played, the hotpath data stream can be read by the video player having the appropriate codecs. In an exemplary embodiment, when video playback reaches a time when an object appears that has an MCA, the MCA may be highlighted or otherwise visible to the user so that the user would know that the object could be tapped (when computer 104 is implemented as or includes a touchscreen) or clicked for more information.

[0041] Moreover, during playback, the video player would move an object's MCA according to each dataset related to the object. In certain embodiments, the video player would determine the position of the object in-between datasets using an expectant motion algorithm, until the object no longer was associated with an MCA (the object need not necessarily be off the video display). The system described herein does not have a limit to the number of objects that can be tracked using MCAs.

[0042] As discussed above with reference to each associating/embedding option, there is a concomitant decoding option upon distribution (step 216). Typically decoding is performed by the media/video player employed by the user, which has the appropriate codecs to decode the information for playback.

[0043] FIG. 8 shows a diagrammatic representation of one embodiment of computing system in the exemplary form of a system 700, e.g., computing device 104, within which a set of instructions that cause a processor 705 to perform any one or more of the aspects and/or methodologies, such as method 200, or to perform the encoding, decoding, and embedding functions described in the present disclosure. It is also contemplated that multiple computing devices, such as computing device 104, or mobile devices, or combinations of computing devices and mobile devices, may be utilized to

implement a specially configured set of instructions for causing the performance of any one or more of the aspects and/or methodologies of the present disclosure.

[0044] System 700 includes a processor 705 and a memory 710 that communicate with each other via a bus 715. Bus 715 may include any of several types of communication structures including, but not limited to, a memory bus, a memory controller, a peripheral bus, a local bus, and any combinations thereof, using any of a variety of architectures. Memory 710 may include various components (e.g., machine-readable media) including, but not limited to, a random access memory component (e.g., a static RAM “SRAM” or a dynamic RAM “DRAM”), a read-only component, and any combinations thereof. In one example, a basic input/output system 720 (BIOS), including basic routines that help to transfer information between elements within system 700, such as during start-up, may be stored in memory 710. Memory 710 may also include (e.g., stored on one or more machine-readable media) instructions (e.g., software) 725 embodying any one or more of the aspects and/or methodologies of the present disclosure. In another example, memory 710 may further include any number of program modules including, but not limited to, an operating system, one or more application programs, other program modules, program data, and any combinations thereof.

[0045] System 700 may also include a storage device 730. Examples of a storage device (e.g., storage device 730) include, but are not limited to, a hard disk drive for reading from and/or writing to a hard disk, a magnetic disk drive for reading from and/or writing to a removable magnetic disk, an optical disk drive for reading from and/or writing to an optical media (e.g., a CD or a DVD), a solid-state memory device, and any combinations thereof. Storage device 730 may be connected to bus 715 by an appropriate interface (not shown). Example interfaces include, but are not limited to, SCSI, advanced technology attachment (ATA), serial ATA, universal serial bus (USB), IEEE 7395 (FIREWIRE), and any combinations thereof. In one example, storage device 730 may be removably interfaced with system 700 (e.g., via an external port connector (not shown)). Particularly, storage device 730 and an associated non-transitory machine-readable medium 735 may provide nonvolatile and/or volatile storage of machine-readable instructions, data structures, program modules, and/or other data for system 700. In one example, instructions 725 may reside, completely or partially, within non-transitory machine-readable medium 735. In another example, instructions 725 may reside, completely or partially, within processor 705.

[0046] System 700 may also include a connection to one or more systems or software modules included with system 70. Any system or device may be interfaced to bus 715 via any of a variety of

interfaces (not shown), including, but not limited to, a serial interface, a parallel interface, a game port, a USB interface, a FIREWIRE interface, a direct connection to bus 715, and any combinations thereof. Alternatively, in one example, a user of system 700 may enter commands and/or other information into system 700 via an input device (not shown). Examples of an input device include, but are not limited to, an alpha-numeric input device (e.g., a keyboard), a pointing device, a joystick, a gamepad, an audio input device (e.g., a microphone, a voice response system, etc.), a cursor control device (e.g., a mouse), a touchpad, an optical scanner, a video capture device (e.g., a still camera, a video camera), a touch screen (as discussed above), and any combinations thereof.

[0047] A user may also input commands and/or other information to system 700 via storage device 730 (e.g., a removable disk drive, a flash drive, etc.) and/or a network interface device 745. A network interface device, such as network interface device 745, may be utilized for connecting system 700 to one or more of a variety of networks, such as network 750, and one or more remote devices 755 connected thereto. Examples of a network interface device include, but are not limited to, a network interface card, a modem, and any combination thereof. Examples of a network include, but are not limited to, a wide area network (e.g., the Internet, an enterprise network), a local area network (e.g., a network associated with an office, a building, a campus, or other relatively small geographic space), a telephone network, a direct connection between two computing devices, and any combinations thereof. A network, such as network 750, may employ a wired and/or a wireless mode of communication. In general, any network topology may be used. Information (e.g., data, instructions 725, etc.) may be communicated to and/or from system 700 via network interface device 755.

[0048] System 700 may further include a video display adapter 760 for communicating a displayable image to a display device 765. Examples of a display device 765 include, but are not limited to, a liquid crystal display (LCD), a cathode ray tube (CRT), a plasma display, and any combinations thereof.

[0049] In addition to display device 765, system 700 may include a connection to one or more other peripheral output devices including, but not limited to, an audio speaker, a printer, and any combinations thereof. Peripheral output devices may be connected to bus 715 via a peripheral interface 770. Examples of a peripheral interface include, but are not limited to, a serial port, a USB connection, a FIREWIRE connection, a parallel connection, a wireless connection, and any combinations thereof.

[0050] In an exemplary embodiment, a method of allowing a user to interact with a video stream is disclosed, the method comprising: accessing the video stream, the video stream including a plurality of objects; developing a hotpath data stream, the hotpath data stream including a plurality of moveable clickable areas, wherein each of the plurality of moveable clickable areas are associated with a corresponding respective one of the plurality of objects; embedding the hotpath data stream with the video stream in a file container; decoding the file container such that the user can interact with ones of the plurality of objects via its corresponding respective one of the plurality of moveable clickable areas. In certain embodiments, the embedding is metadata embedding. In certain embodiments, the hotpath data stream is embedded as an arbitrary time-based metadata track. In certain embodiments, the embedding is subtitle embedding. In certain embodiments, a plurality of subtitle streams are provided with the video stream, and wherein one of the plurality of subtitle streams is replaced by the hotpath data stream. In certain embodiments, the embedding is video embedding. In certain embodiments, the video stream includes a plurality of unused variable length codes, and wherein the hotpath data stream is encoded into the video stream by re-purposing the plurality of unused variable length codes. In certain embodiments, the plurality of unused variable length codes are statistically determined. In certain embodiments, a map of the plurality of unused variable length codes developed according to a video stream frame number and position within the video stream frame. In certain embodiments, the hotpath data stream is encoded in the video stream using Huffman coding. In certain embodiments, the embedding is audio embedding. In certain embodiments, the audio embedding uses least significant bit (LSB) insertion to embed the hotpath data stream in an audio stream.

[0051] In another exemplary aspect, a system for allowing a user to retrieve information related to objects found in a video stream is disclosed, the system comprising: a computing device, the computing device including a processor having a set of instructions, the set of instructions configured to: access the video stream, the video stream including a plurality of objects; develop a hotpath data stream, the hotpath data stream including information and a moveable clickable area associated with each of the plurality of objects; embed the hotpath data stream with the video stream in a file container. In certain embodiments, decoding the file container such that the user can interact with ones of the plurality of objects via its corresponding respective one of the plurality of moveable clickable areas. In certain embodiments, embedding the hotpath data stream is performed using metadata embedding. In certain embodiments, the hotpath data stream is embedded as an arbitrary time-based metadata track. In certain embodiments, embedding the hotpath data stream is performed

using subtitle embedding. In certain embodiments, a plurality of subtitle streams are provided with the video stream, and wherein one of the plurality of subtitle streams is replaced by the hotpath data stream. In certain embodiments, embedding the hotpath data stream is performed using video embedding. In certain embodiments, the video stream includes a plurality of unused variable length codes, and wherein the hotpath data stream is encoded into the video stream by re-purposing the plurality of unused variable length codes. In certain embodiments, the plurality of unused variable length codes are statistically determined. In certain embodiments, a map of the plurality of unused variable length codes developed according to a video stream frame number and position within the video stream frame. In certain embodiments, the hotpath data stream is encoded in the video stream using Huffman coding. In certain embodiments, embedding the hotpath data stream is performed using audio embedding. In certain embodiments, the audio embedding uses least significant bit (LSB) insertion to embed the hotpath data stream in an audio stream.

[0052] Exemplary embodiments have been disclosed above and illustrated in the accompanying drawings. It will be understood by those skilled in the art that various changes, omissions and additions may be made to that which is specifically disclosed herein without departing from the spirit and scope of the present invention.

What is claimed is:

1. A method of allowing a user to interact with a video stream, the method comprising:
accessing the video stream, the video stream including a plurality of objects;
developing a hotpath data stream, the hotpath data stream including a plurality of
moveable clickable areas, wherein each of the plurality of moveable clickable areas
are associated with a corresponding respective one of the plurality of objects;
embedding the hotpath data stream with the video stream in a file container;
decoding the file container such that the user can interact with ones of the plurality of
objects via its corresponding respective one of the plurality of moveable clickable
areas.
2. The method according to claim 1, wherein the embedding is metadata embedding.
3. The method according to claim 2, wherein the hotpath data stream is embedded as an
arbitrary time-based metadata track.
4. The method according to claim 1, wherein the embedding is subtitle embedding.
5. The method according to claim 4, wherein a plurality of subtitle streams are provided with
the video stream, and wherein one of the plurality of subtitle streams is replaced by the
hotpath data stream.
6. The method according to claim 1, wherein the embedding is video embedding.
7. The method according to claim 6, wherein the video stream includes a plurality of unused
variable length codes, and wherein the hotpath data stream is encoded into the video stream
by re-purposing the plurality of unused variable length codes.
8. The method according to claim 7, wherein the plurality of unused variable length codes are
statistically determined.
9. The method according to claim 7, wherein a map of the plurality of unused variable length
codes developed according to a video stream frame number and position within the video
stream frame.
10. The method according to claim 9, the hotpath data stream is encoded in the video stream
using Huffman coding.
11. The method according to claim 1, wherein the embedding is audio embedding.
12. The method according to claim 11, wherein the audio embedding uses least significant bit
(LSB) insertion to embed the hotpath data stream in an audio stream.

13. A system for allowing a user to retrieve information related to objects found in a video stream, the system comprising:
 - a computing device, the computing device including a processor having a set of instructions, the set of instructions configured to:
 - access the video stream, the video stream including a plurality of objects;
 - develop a hotpath data stream, the hotpath data stream including information and a moveable clickable area associated with each of the plurality of objects;
 - embed the hotpath data stream with the video stream in a file container.
14. The system according to claim 13, decoding the file container such that the user can interact with ones of the plurality of objects via its corresponding respective one of the plurality of moveable clickable areas.
15. The system according to claim 13, wherein embedding the hotpath data stream is performed using metadata embedding.
16. The system according to claim 16, wherein the hotpath data stream is embedded as an arbitrary time-based metadata track.
17. The system according to claim 13, wherein embedding the hotpath data stream is performed using subtitle embedding.
18. The system according to claim 17, wherein a plurality of subtitle streams are provided with the video stream, and wherein one of the plurality of subtitle streams is replaced by the hotpath data stream.
19. The system according to claim 13, wherein embedding the hotpath data stream is performed using video embedding.
20. The system according to claim 19, wherein the video stream includes a plurality of unused variable length codes, and wherein the hotpath data stream is encoded into the video stream by re-purposing the plurality of unused variable length codes.
21. The system according to claim 20, wherein the plurality of unused variable length codes are statistically determined.
22. The system according to claim 21, wherein a map of the plurality of unused variable length codes developed according to a video stream frame number and position within the video stream frame.
23. The system according to claim 22, the hotpath data stream is encoded in the video stream using Huffman coding.

24. The system according to claim 13, wherein embedding the hotpath data stream is performed using audio embedding.
25. The system according to claim 24, wherein the audio embedding uses least significant bit (LSB) insertion to embed the hotpath data stream in an audio stream.

1/8

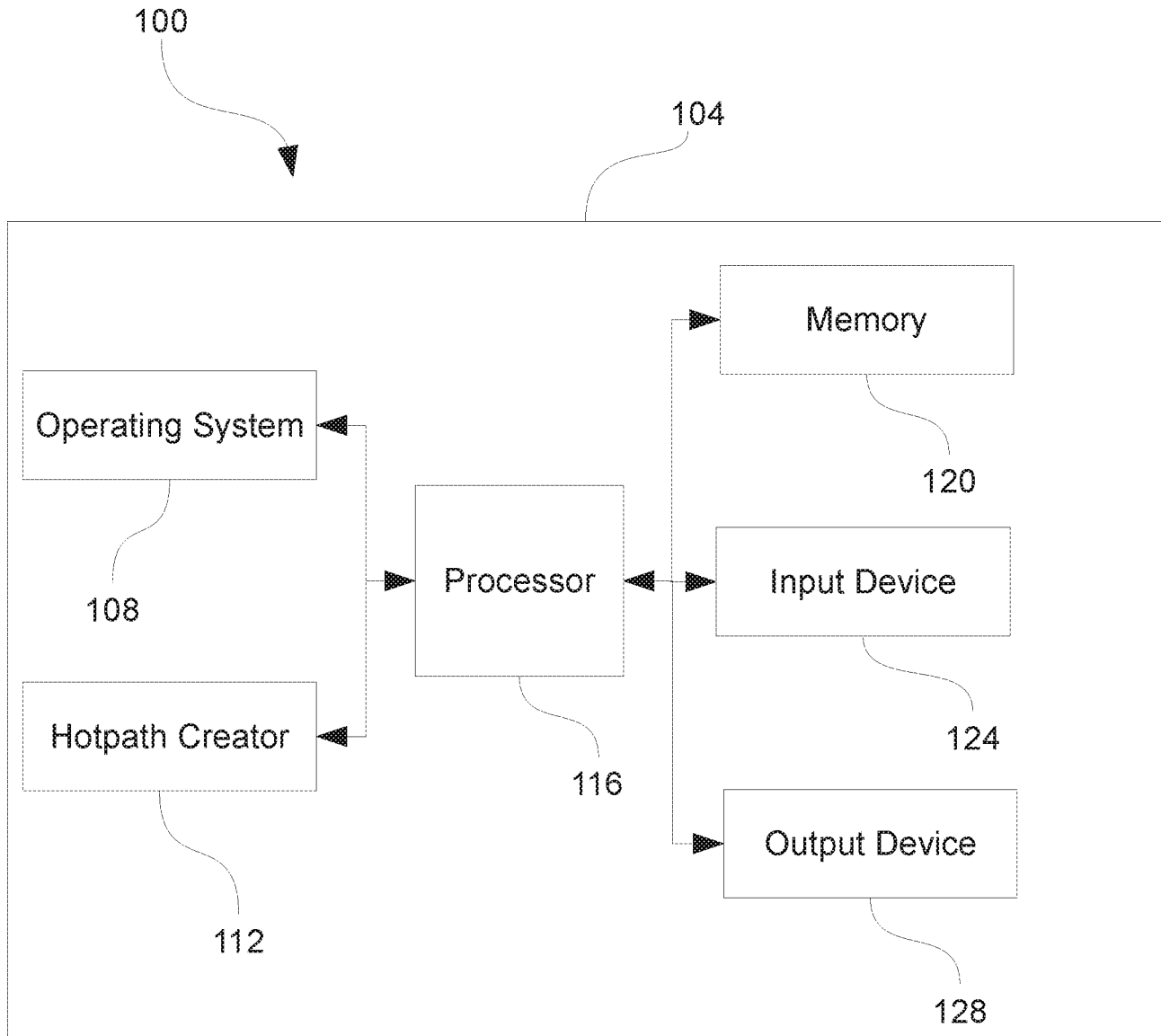


FIG. 1

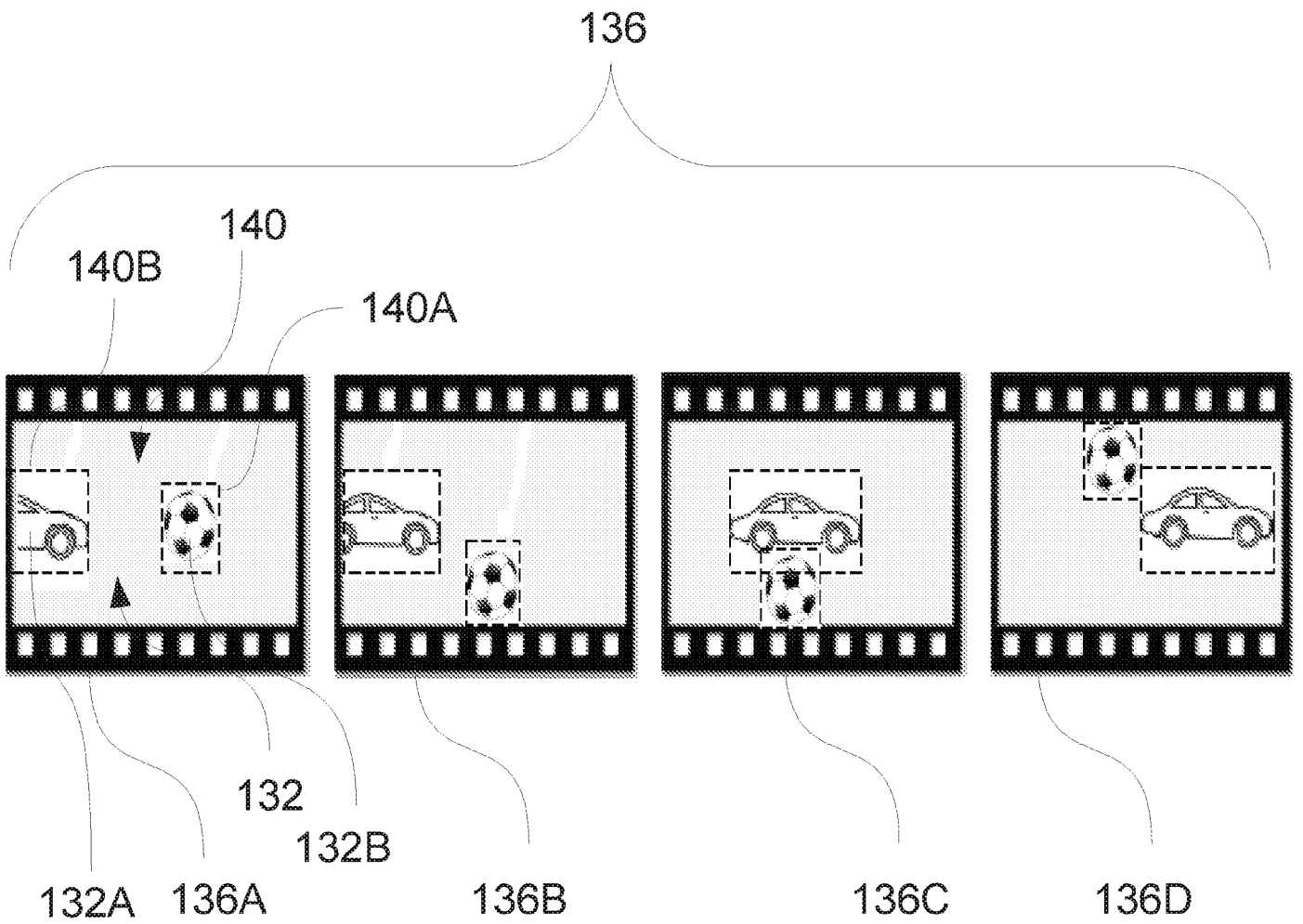


FIG. 2

3/8

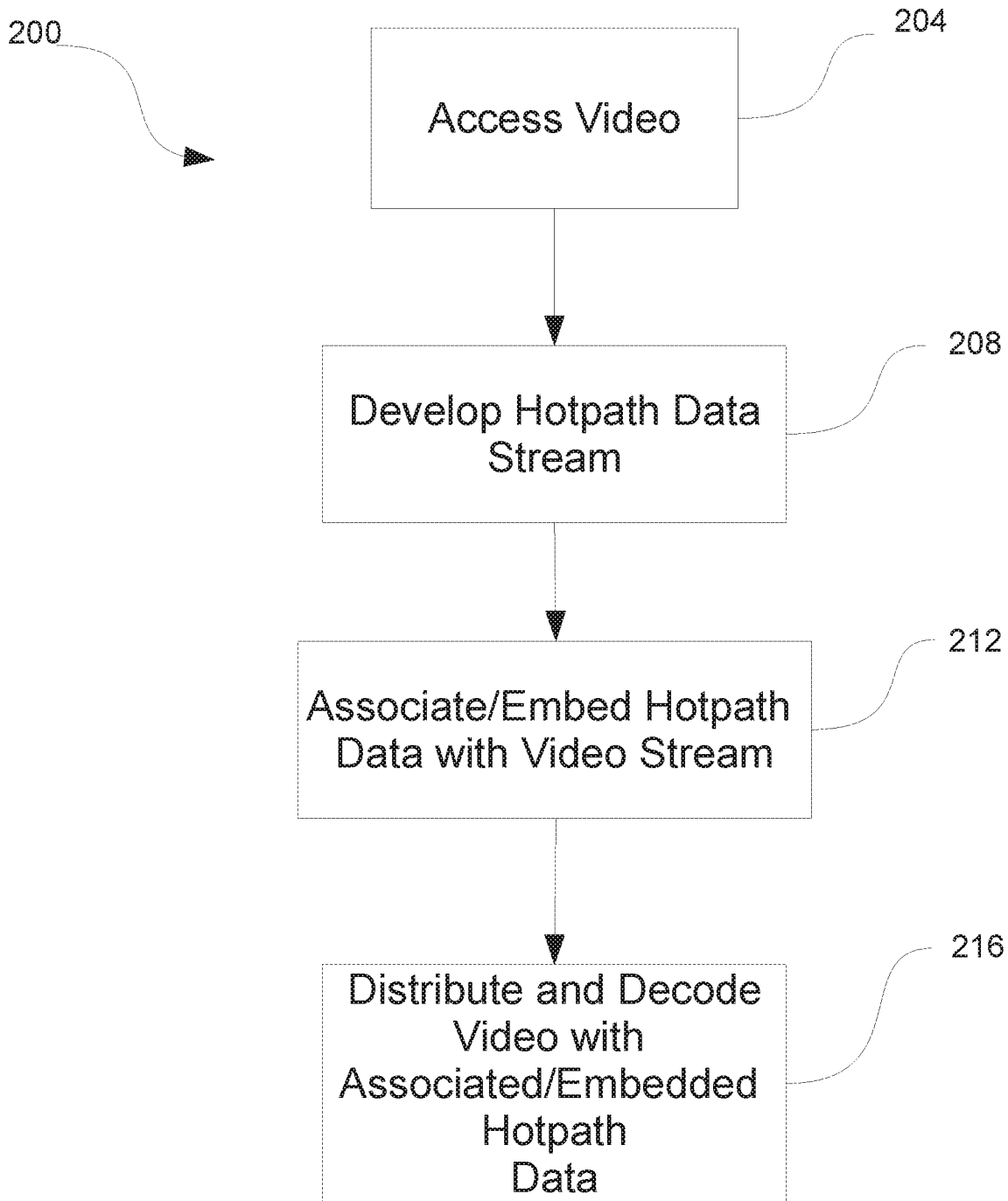


FIG. 3

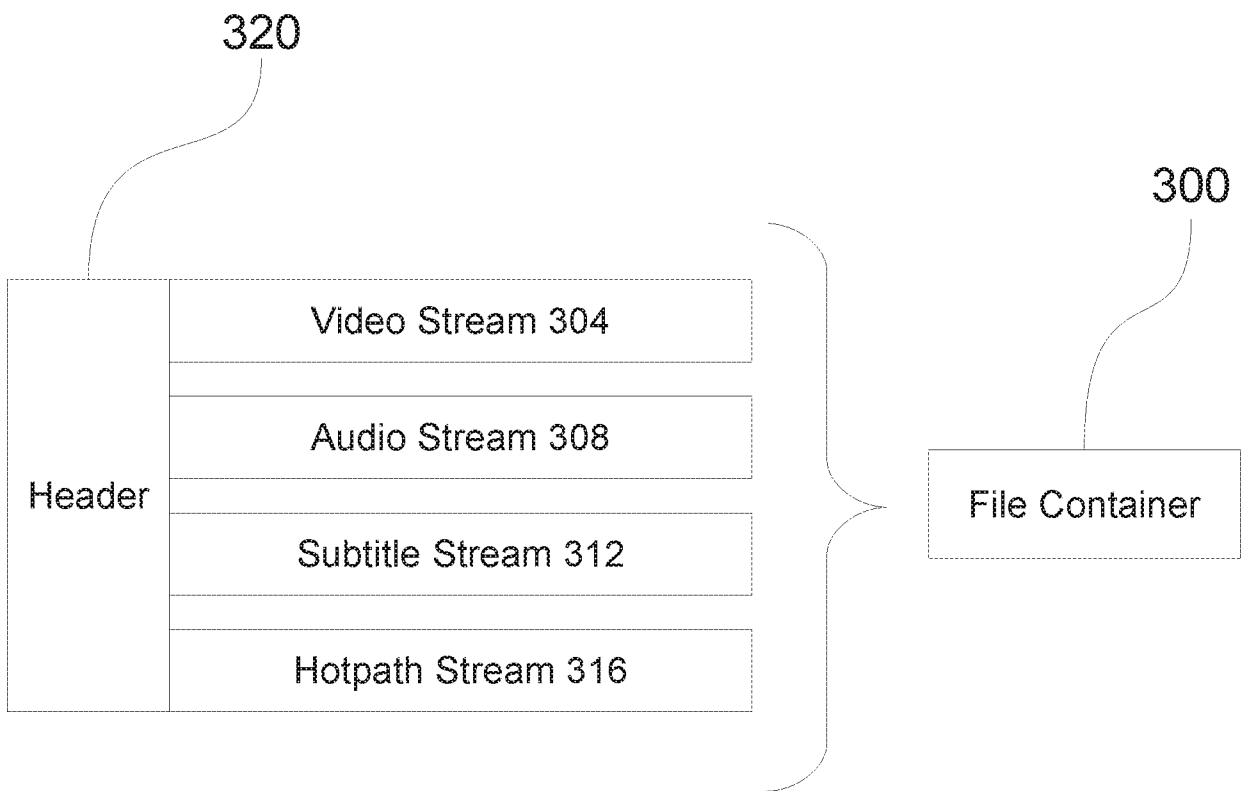


FIG. 4

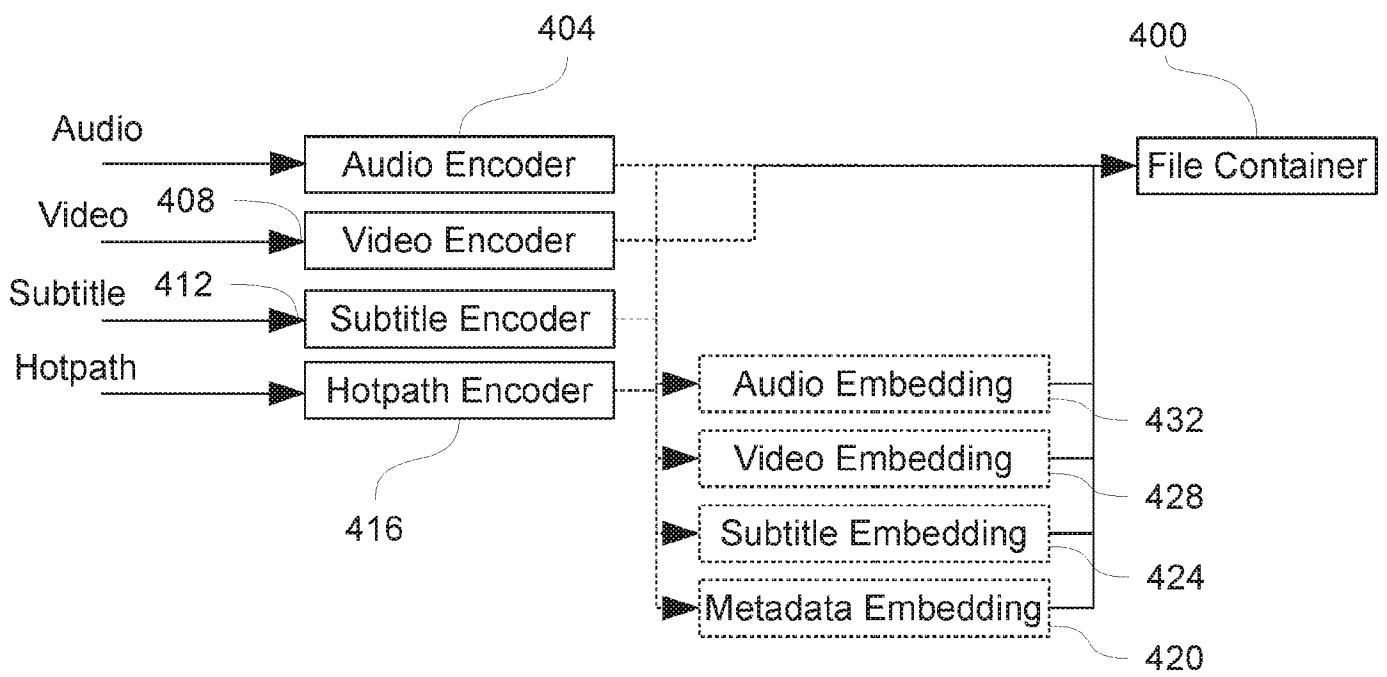


FIG. 5

6/8

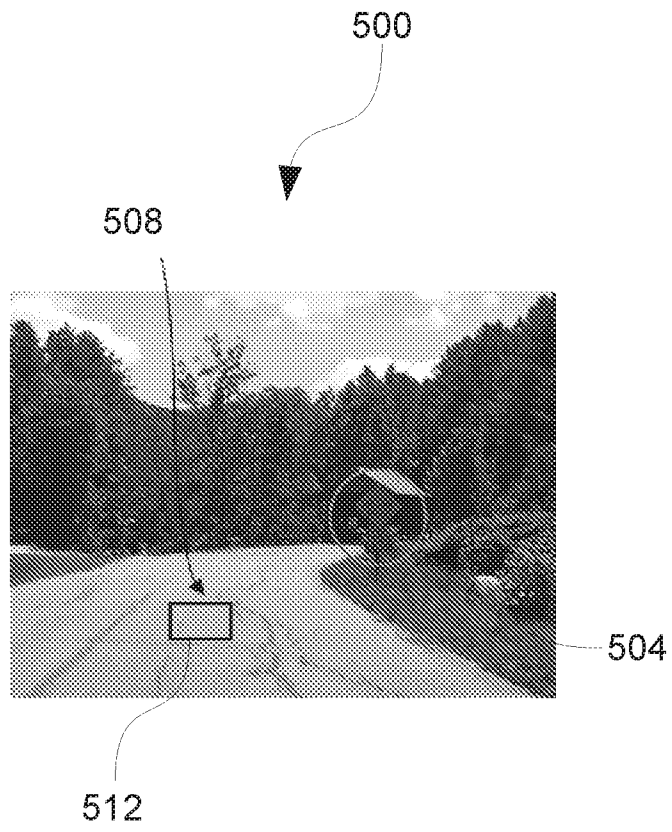


FIG. 6

Original audio bytes	Text hotpath data to embed	Text hotpath data embedded in audio bytes
10010010	0	10010010
01010011	1	01010011
10011011	0	10011010
11010011	0	11010010
10001010	0	10001010
00000010	1	00000011
01110010	0	01110010
00101011	0	00101010

FIG. 7

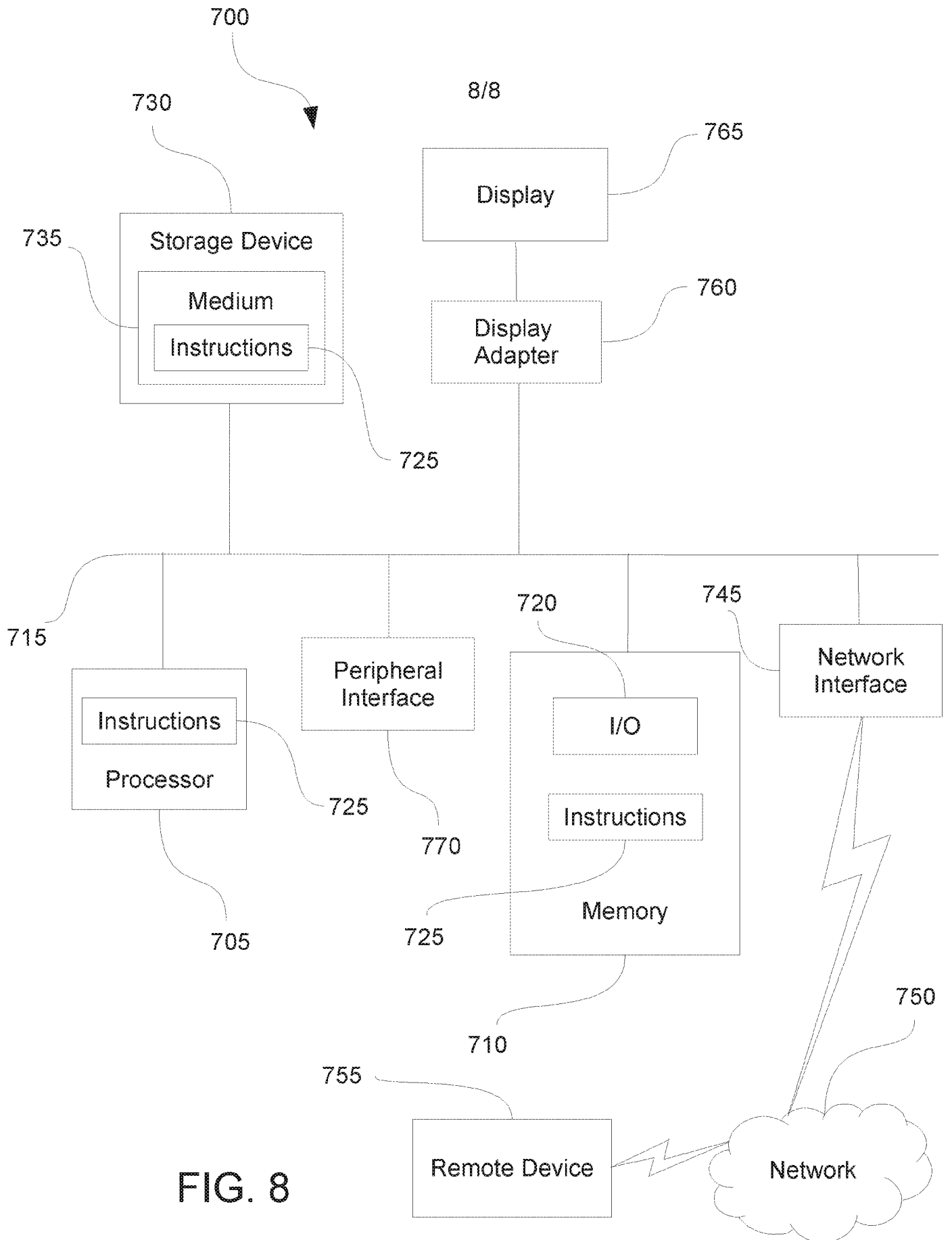


FIG. 8

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US16/29890

A. CLASSIFICATION OF SUBJECT MATTER

IPC(8) - H03M 7/40 (2016.01)

CPC - H03M 7/40

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)

IPC(8): G06Q 30/02, G06F 9/44, H04N 7/025, H04N 21/218, H03M 7/40 (2016.01)

CPC: G06Q 30/02, G06F 9/44, H04N 7/025, H04N21/20, H03M 7/40

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

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

PatSeer (US, EP, WO, JP, DE, GB, CN, FR, KR, ES, AU, IN, CA, INPADOC Data);

IP.com; IEEE; Google/Google Scholar

KEYWORDS: Video, audio, Huffman, VLC.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	PURNAMASARI et al. "Clickable and Interactive Video System Using HTML5"; ICOIN IEEE; Publication [online]. 12 February 2014 [retrieved 27 June 2016]. Retrieved from the Internet: <URL: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6799697&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D6799697>; pp. 232-237.	1-9, 11, 13-22, 24
Y		10, 12, 23, 25
Y	US 2014/0181882 A1 (CANON KABUSHIKI KAISHA) 26 June 2014; paragraphs [0060], [0109].	10, 23
Y	EP 1 332 470 B1 (MEYER, T et al.) 08 September 2010; paragraph [0056].	12, 25

Further documents are listed in the continuation of Box C.

See patent family annex.

* Special categories of cited documents:

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search

27 June 2016 (27.06.2016)

Date of mailing of the international search report

29 JUL 2016

Name and mailing address of the ISA/

Mail Stop PCT, Attn: ISA/US, Commissioner for Patents

P.O. Box 1450, Alexandria, Virginia 22313-1450

Facsimile No. 571-273-8300

Authorized officer

Shane Thomas

PCT Helpdesk: 571-272-4300

PCT OSP: 571-272-7774