



- (51) International Patent Classification:
H04L 12/801 (2013.01) *H04L 29/08* (2006.01)
- (21) International Application Number:
PCT/US2017/026928
- (22) International Filing Date:
11 April 2017 (11.04.2017)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
62/335,410 12 May 2016 (12.05.2016) US
15/283,346 01 October 2016 (01.10.2016) US
- (71) Applicant: INTEL CORPORATION [US/US]; 2200 Mission College Boulevard, Santa Clara, California 95054 (US).
- (72) Inventors: VEERAMANI, Karthik; 2226 NW Thorncroft Dr., Apt. 532, Hillsboro, Oregon 97124 (US). DIEFENBAUGH, Paul S.; 9016 NW Bartholomew Drive, Portland, Oregon 97229 (US). KUMAR, Arvind; 291 NW Pacific Grove Drive, Beaverton, Oregon 97006 (US).
- (74) Agent: KELLETT, Glen M.; Barnes & Thornburg LLP, 11 South Meridian Street, Indianapolis, IN 46204 (US).
- (81) Designated States (*unless otherwise indicated, for every kind of national protection available*): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC,

(54) Title: TECHNOLOGIES FOR INPUT COMPUTE OFFLOADING OVER A WIRELESS CONNECTION

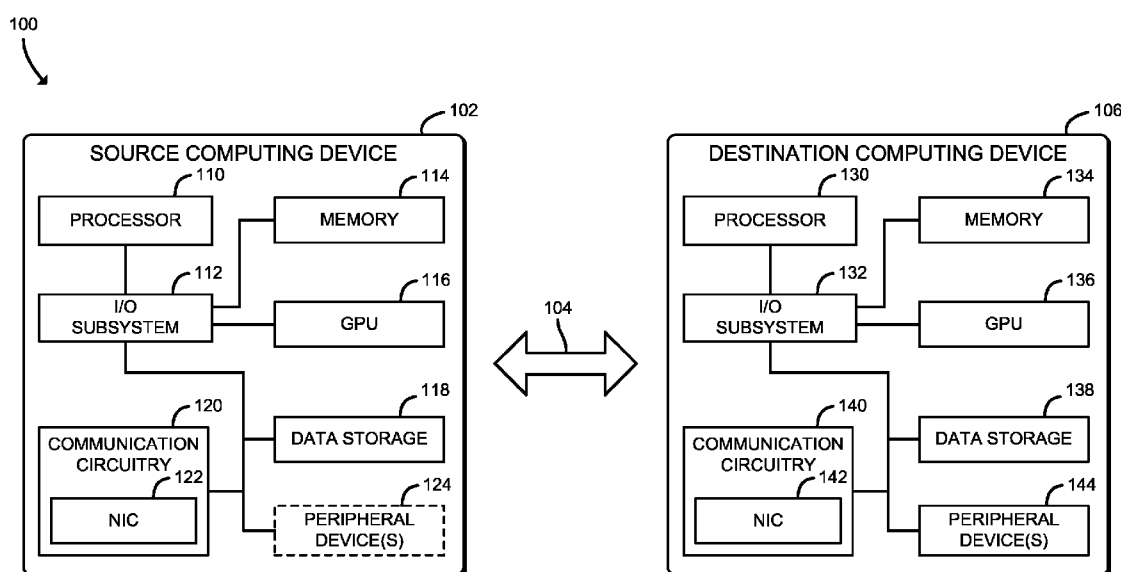


FIG. 1

(57) Abstract: Technologies for input compute offloading of digital content include a source computing device for wirelessly transmitting the digital content to a destination computing device. The destination computing device is configured to detect inputs initiated by a user on a display of the destination computing device and transmit input characteristics to the source computing device that are usable by the source computing device to render the digital content to include one or more objects based on the one or more input characteristics. The source computing device is configured to receive the input characteristics from the destination computing device and render the digital content to include one or more objects based on the one or more input characteristics. Other embodiments are described and claimed.

SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR,
TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

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

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))*

Published:

- *with international search report (Art. 21(3))*

TECHNOLOGIES FOR INPUT COMPUTE OFFLOADING OVER A WIRELESS CONNECTION

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] The present application claims priority to U.S. Utility Patent Application Serial No. 15/283,346, entitled “TECHNOLOGIES FOR INPUT COMPUTE OFFLOADING OVER A WIRELESS CONNECTION,” which was filed on October 01, 2016 and which claims priority under 35 U.S.C. § 119(e) to U.S. Provisional Patent Application Serial No. 62/335,410, entitled “TECHNOLOGIES FOR INPUT COMPUTE OFFLOADING OVER A WIRELESS CONNECTION,” which was filed on May 12, 2016.

BACKGROUND

[0002] Traditionally, playback of digital content (e.g., movies, music, pictures, games, etc.) has been constrained to the computing device (e.g., desktop computer, smartphone, tablet, wearable, gaming system, television, etc.) on which the digital content was stored. However, with the advent of cloud computing related technologies and increased capabilities of computing devices, services such as digital content streaming, casting, and mirroring have sped up the generation, sharing, and consumption of digital content as consumer devices capable of interacting with such content have become ubiquitous.

[0003] To deal with such vast amounts of data transfer in the on-demand landscape, various compression technologies have been implemented to support the streaming of digital content in real-time with reduced latency. Such compression technologies (i.e., codecs and containers) include Moving Picture Experts Group standards (e.g., MPEG-2, MPEG-4, H.264, etc.) and MPEG transport stream (MPEG-TS). Further, various network control protocols, such as real time streaming protocol (RTSP), for example, have been developed for establishing and controlling media sessions between endpoint computing devices. Finally, various transport protocols (e.g., real-time transport protocol (RTP)) usable by the endpoint computing devices have been established for providing end-to-end network transport functions suitable for transmission of the digital content in real-time.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] The concepts described herein are illustrated by way of example and not by way of limitation in the accompanying figures. For simplicity and clarity of illustration, elements illustrated in the figures are not necessarily drawn to scale. Where considered

appropriate, reference labels have been repeated among the figures to indicate corresponding or analogous elements.

[0005] FIG. 1 is a simplified block diagram of at least one embodiment of a system for input compute offloading over a wireless connection;

[0006] FIG. 2 is a simplified block diagram of at least one embodiment of an environment of the source computing device of the system of FIG. 1;

[0007] FIG. 3 is a simplified block diagram of at least one embodiment of an environment of the destination computing device of the system of FIG. 1;

[0008] FIG. 4 is a simplified block diagram of another embodiment of the environment of the source computing device of FIG. 2;

[0009] FIG. 5 is a simplified block diagram of another embodiment of the environment of the destination computing device of FIG. 3;

[0010] FIG. 6 is a simplified communication flow diagram of at least one embodiment for performing an input compute offloading capability exchange between the source computing device of FIGS. 2 and 4, and the destination computing device of FIGS. 3 and 5;

[0011] FIG. 7 is a simplified flow diagram of at least one embodiment for offloading input compute that may be executed by the source computing device of FIGS. 2 and 4; and

[0012] FIG. 8 is a simplified flow diagram of at least one embodiment for offloading input compute that may be executed by the destination computing device of FIGS. 3 and 5.

DETAILED DESCRIPTION OF THE DRAWINGS

[0013] While the concepts of the present disclosure are susceptible to various modifications and alternative forms, specific embodiments thereof have been shown by way of example in the drawings and will be described herein in detail. It should be understood, however, that there is no intent to limit the concepts of the present disclosure to the particular forms disclosed, but on the contrary, the intention is to cover all modifications, equivalents, and alternatives consistent with the present disclosure and the appended claims.

[0014] References in the specification to “one embodiment,” “an embodiment,” “an illustrative embodiment,” etc., indicate that the embodiment described may include a particular feature, structure, or characteristic, but every embodiment may or may not necessarily include that particular feature, structure, or characteristic. Moreover, such phrases are not necessarily referring to the same embodiment. Further, when a particular feature, structure, or characteristic is described in connection with an embodiment, it is submitted that it is within the knowledge of one skilled in the art to affect such feature,

structure, or characteristic in connection with other embodiments whether or not explicitly described. Additionally, it should be appreciated that items included in a list in the form of “at least one of A, B, and C” can mean (A); (B); (C); (A and B); (A and C); (B and C); or (A, B, and C). Similarly, items listed in the form of “at least one of A, B, or C” can mean (A); (B); (C); (A and B); (A and C); (B and C); or (A, B, and C).

[0015] The disclosed embodiments may be implemented, in some cases, in hardware, firmware, software, or any combination thereof. The disclosed embodiments may also be implemented as instructions carried by or stored on one or more transitory or non-transitory machine-readable (e.g., computer-readable) storage media, which may be read and executed by one or more processors. A machine-readable storage medium may be embodied as any storage device, mechanism, or other physical structure for storing or transmitting information in a form readable by a machine (e.g., a volatile or non-volatile memory, a media disc, or other media device).

[0016] In the drawings, some structural or method features may be shown in specific arrangements and/or orderings. However, it should be appreciated that such specific arrangements and/or orderings may not be required. Rather, in some embodiments, such features may be arranged in a different manner and/or order than shown in the illustrative figures. Additionally, the inclusion of a structural or method feature in a particular figure is not meant to imply that such feature is required in all embodiments and, in some embodiments, may not be included or may be combined with other features.

[0017] Referring now to FIG. 1, in an illustrative embodiment, a system 100 for transmitting (e.g., streaming, mirroring, casting, etc.) digital content (e.g., video content, audio content, streaming text content, etc.) includes a source computing device 102 communicatively coupled to a destination computing device 106 via a wireless communication channel 104. In use, the source computing device 102 transmits the digital content presently being displayed on, or otherwise presently being processed by, the source computing device 102 to the destination computing device 106 via the wireless communication channel 104. For example, the source computing device 102 may capture images of output presently being rendered on the screen of the source computing device (i.e., a screen capture).

[0018] As will be described in further detail, during output (i.e., display) by the destination computing device 106 of digital content received from the source computing device 102, a user of the destination computing device 106 may provide an input to the destination computing device 106 (e.g., via an input device) to initiate an action on an application (e.g., a writing/drawing application) presently executing on the source computing

device 102, which is transmitting to the destination computing device 106. During viewing of the digital content, the user may provide an input (e.g., via directly touching or using a stylus on a touchscreen display, pressing a key on a keyboard, moving a controllable element of a mouse, speaking an audible voice command captured by a microphone, etc.) to the destination computing device 106 in which an outcome of the input is expected.

[0019] For example, the expected outcome may be to render one or more objects (e.g., text, shapes, lines, graphics, etc.) on the display of the destination computing device 106 based on the detected local input. In an illustrative example, the user may draw or otherwise insert an object on a display (e.g., a touchscreen display) of the destination computing device 106 while the destination computing device 106 is displaying the digital content received from the source computing device 102 with the expectation that the destination computing device 106 is to display the object. In another illustrative example, the user may change a setting of an application presently executing on the source computing device 102 that is viewable on the display of the destination computing device 106. Under such conditions, the destination computing device 106 is configured to temporarily render the detected local input using one or more objects to allow the user to view/change the setting(s) and transmit one or more characteristics of the detected input to the source computing device 102. The input characteristics may include any data usable by the source computing device 102 to identify a location of the input relative to the location of the displayed image on the destination computing device 106, as well as any data usable to render the desired output at the corresponding location. Such input characteristics may include coordinates (e.g., screen/display coordinates, output content coordinates, input border coordinates, etc.), an input type (e.g., text, a shape, a graphic, etc.), font/line characteristics (e.g., types, styles, sizes, colors, weights, etc.), etc.

[0020] The source computing device 102 is configured to identify the received input characteristics and render the inputs on the display of the source computing device 102. The source computing device 102 is further configured to transmit the updated digital content. The source computing device 102 may additionally be configured to provide an indication to the destination computing device 106 indicating the digital content has been updated to include the input. Accordingly, upon receipt of the updated digital content or the indication, the destination computing device 106 can discontinue rendering the temporary overlay and output the updated digital content received from the source computing device 102. In other words, the destination computing device 106 can just display the received digital content that has been updated to include the input previously transmitted from the destination computing device 106.

[0021] It should be appreciated that the transmitting of digital content discussed herein is applicable to different types of transmission including, but not limited to, streaming of digital content, mirroring of digital content, and casting of digital content. As such, although the term “stream” or “streaming” may be used at times to describe a particular type of transmission, it should be appreciated that the corresponding transmission may be effected by mirroring, casting, or otherwise transmitting using another transmission modality. In typical streaming transmissions, the digital content is progressively transferred. For example, instead of downloading or retrieving the full digital content, a client device (e.g., the destination computing device 106) may actively play a portion of the digital content while downloading or retrieving other parts of the digital content. In typical mirroring transmissions, a source device shares its screen (or content that would be displayed on its screen) with a destination device. The digital content transmission during a mirroring session may use a progressive transfer or non-progressive data transfer (e.g., the digital content may be downloaded completely). In typical casting transmissions, a source device shares content with a destination device. The digital content transmission during a casting session may use a progressive transfer or non-progressive data transfer. Additionally, in some implementations, the source device may transmit a link or other location-indicator to digital content, which is subsequently retrieved by the destination device from a source different from the source device. (e.g., the digital content may be downloaded completely).

[0022] It should be further appreciated that, while the context of the present disclosure is described below as receiving input from a user (e.g., via directly touching or using a stylus on a touchscreen display, pressing a key on a keyboard, detecting movement of an element of a mouse, receiving an audible voice command by a microphone, etc.) to a display of the destination computing device 106, such functionality described herein may be usable for other forms of detected input that is capable of being characterized and rendered as described herein.

[0023] The source computing device 102 may be embodied as any type of computing device that is capable of performing the functions described herein, such as, without limitation, a portable computing device (e.g., smartphone, tablet, laptop, notebook, wearable, etc.) that includes mobile hardware (e.g., processor, memory, storage, wireless communication circuitry, etc.) and software (e.g., an operating system) to support a mobile architecture and portability, a computer, a server (e.g., stand-alone, rack-mounted, blade, etc.), a network appliance (e.g., physical or virtual), a web appliance, a distributed computing system, a processor-based system, a multiprocessor system, a set-top box, and/or any other computing/communication device capable of performing the functions described herein.

[0024] The illustrative source computing device 102 includes a processor (i.e., a CPU) 110, an input/output (I/O) subsystem 112, a memory 114, a graphics processing unit (GPU) 116, a data storage device 118, and communication circuitry 120, as well as, in some embodiments, one or more peripheral devices 124. Of course, the source computing device 102 may include other or additional components in other embodiments, such as those commonly found in a computing device. Additionally, in some embodiments, one or more of the illustrative components may be incorporated in, or otherwise form a portion of, another component. For example, in some embodiments, the memory 114, or portions thereof, may be incorporated in the processor 110. Further, in some embodiments, one or more of the illustrative components may be omitted from the source computing device 102.

[0025] The processor 110 may be embodied as any type of processor capable of performing the functions described herein. Accordingly, the processor 110 may be embodied as a single or multi-core processor(s), digital signal processor, microcontroller, or other processor or processing/controlling circuit. The memory 114 may be embodied as any type of volatile or non-volatile memory or data storage capable of performing the functions described herein. In operation, the memory 114 may store various data and software used during operation of the source computing device 102, such as operating systems, applications, programs, libraries, and drivers.

[0026] The memory 114 is communicatively coupled to the processor 110 via the I/O subsystem 112, which may be embodied as circuitry and/or components to facilitate input/output operations with the processor 110, the memory 114, and the GPU 116, as well as other components of the source computing device 102. For example, the I/O subsystem 112 may be embodied as, or otherwise include, memory controller hubs, input/output control hubs, firmware devices, communication links (i.e., point-to-point links, bus links, wires, cables, light guides, printed circuit board traces, etc.) and/or other components and subsystems to facilitate the input/output operations. In some embodiments, the I/O subsystem 112 may form a portion of a system-on-a-chip (SoC) and be incorporated, along with the processor 110, the memory 114, and other components of the source computing device 102, on a single integrated circuit chip.

[0027] The GPU 116 may be embodied as circuitry and/or components to handle specific types of tasks assigned to the GPU 116, such as image rendering, for example. To do so, the GPU 116 may include an array of processor cores or parallel processors (not shown), each of which can execute a number of parallel and concurrent threads. In some embodiments, the processor cores of the GPU 116 may be configured to individually handle 3D rendering tasks, blitter (e.g., 2D graphics), and/or video encoding/decoding tasks, by

providing electronic circuitry that can perform mathematical operations rapidly using extensive parallelism and multiple concurrent threads. It should be appreciated that, in some embodiments, the GPU 116 may have direct access to the memory 114, thereby allowing direct memory access (DMA) functionality in such embodiments.

[0028] The data storage device 118 may be embodied as any type of device or devices configured for short-term or long-term storage of data such as, for example, memory devices and circuits, memory cards, hard disk drives, solid-state drives, or other data storage devices. It should be appreciated that the data storage device 118 and/or the memory 114 (e.g., the computer-readable storage media) may store various data as described herein, including operating systems, applications, programs, libraries, drivers, instructions, etc., capable of being executed by a processor (e.g., the processor 110) of the source computing device 102.

[0029] The communication circuitry 120 may be embodied as any communication circuit, device, or collection thereof, capable of enabling communications between the source computing device 102 and other computing devices (e.g., the destination computing device 106 and/or other computing devices communicatively coupled to the source computing device 102) over a wired or wireless communication channel (e.g., the wireless communication channel 104). The communication circuitry 120 may be configured to use any one or more wired or wireless communication technologies and associated protocols (e.g., Ethernet, Wi-Fi®, Wi-Fi Direct®, Bluetooth®, Bluetooth® Low Energy (BLE), near-field communication (NFC), Worldwide Interoperability for Microwave Access (WiMAX), etc.) and/or certified technologies (e.g., Digital Living Network Alliance (DLNA), Miracast™, etc.) to affect such communication. The communication circuitry 120 may be additionally configured to use any one or more wireless and/or wired communication technologies and associated protocols to effect communication with other computing devices, such as over a network, for example.

[0030] The illustrative communication circuitry 120 includes a network interface controller (NIC) 122. The NIC 122 may be embodied as one or more add-in-boards, daughtercards, network interface cards, controller chips, chipsets, or other devices that may be used by the source computing device 102. In some embodiments, for example, the NIC 122 may be integrated with the processor 110, embodied as an expansion card coupled to the I/O subsystem 112 over an expansion bus (e.g., PCI Express), included as a part of a SoC that includes one or more processors, or included on a multichip package that also contains one or more processors.

[0031] The peripheral devices 124 may include any number of I/O devices, interface devices, and/or other peripheral devices. For example, in some embodiments, the peripheral devices 124 may include a display, a touch screen, graphics circuitry, a keyboard, a mouse, a microphone, a speaker, and/or other input/output devices, interface devices, and/or peripheral devices. The particular devices included in the peripheral devices 124 may depend on, for example, the type and/or intended use of the source computing device 102. The peripheral devices 124 may additionally or alternatively include one or more ports, such as a universal serial bus (USB) port, a high-definition multimedia interface (HDMI) port, etc., for connecting external peripheral devices to the source computing device 102.

[0032] In the illustrative embodiment, the wireless communication channel 104 is embodied as a direct line of communication (i.e., no wireless access point) between the source computing device 102 and the destination computing device 106. For example, the wireless communication channel 104 may be established over an ad hoc peer-to-peer connection, such as Wi-Fi Direct®, Intel® Wireless Display (WiDi), Bluetooth®, etc., using a wireless display standard (e.g., AirPlay®, Miracast™, DLNA, etc.) Alternatively, in some embodiments, the wireless communication channel 104 may be embodied as any type of wireless communication network, including a wireless local area network (WLAN), a wireless personal area network (WPAN), a cellular network (e.g., Global System for Mobile Communications (GSM), Long-Term Evolution (LTE), etc.), or any combination thereof. It should be appreciated that, in such embodiments, the wireless communication channel 104 may serve as a centralized network and, in some embodiments, may be communicatively coupled to another network (e.g., the Internet). Accordingly, in such embodiments, the wireless communication channel 104 may include a variety of virtual and/or physical network devices (not shown), such as routers, switches, network hubs, servers, storage devices, compute devices, etc., as needed to facilitate the transfer of data between the source computing device 102 and the destination computing device 106.

[0033] The destination computing device 106 may be embodied as any type of computation or computing device capable of performing the functions described herein, including, without limitation, a computer, a portable computing device (e.g., smartphone, tablet, laptop, notebook, wearable, etc.), a “smart” television, a cast hub, a cast dongle, a processor-based system, and/or a multiprocessor system. Similar to the illustrative source computing device 102, the destination computing device 106 includes a processor 130, an I/O subsystem 132, a memory 134, a GPU 136, a data storage device 138, communication circuitry 140 that includes a NIC 142, and one or more peripheral devices 144. As such, further descriptions of the like components are not repeated herein with the understanding

that the description of the corresponding components provided above in regard to the source computing device 102 applies equally to the corresponding components of the destination computing device 106.

[0034] Referring now to FIG. 2, in an illustrative embodiment, the source computing device 102 establishes an environment 200 during operation. The illustrative environment 200 includes a communication manager 210, a capability exchange negotiator 220, and a digital content adjustment manager 230. The various components of the environment 200 may be embodied as hardware, firmware, software, or a combination thereof. Additionally, in some embodiments, one or more of the illustrative components may form a portion of another component and/or one or more of the illustrative components may be independent of one another. Further, in some embodiments, one or more of the components of the environment 200 may be embodied as virtualized hardware components or emulated architecture, which may be established and maintained by the one or more processors and/or other hardware components of the source computing device 102.

[0035] In an illustrative embodiment, as shown in FIG. 4, the environment 200 may include a communication management circuit 210, a capability exchange negotiation circuit 220, and a digital content adjustment circuit 230. In such embodiments, each circuit 210, 220, 230 may be embodied as a dedicated circuit/hardware component or be embodied as a portion of another hardware component of the source computing device 102. For example, in some embodiments, one or more of the communication management circuit 210, the capability exchange negotiation circuit 220, and the digital content adjustment circuit 230 may form a portion of the one or more of the processor 110, the I/O subsystem 112, the GPU 116, the communication circuitry 120, and/or other components of the source computing device 102.

[0036] Additionally or alternatively, one or more of the communication management circuit 210, the capability exchange negotiation circuit 220, and/or the digital content adjustment circuit 230 may be implemented as special purpose hardware circuits or components. Such dedicated or special purpose hardware circuits or logic may complement certain software functions, which may facilitate the calling of such functions by various software programs or applications executed by the source computing device 102 to complete one or more tasks. It should be appreciated that the source computing device 102 may include other components, sub-components, modules, sub-modules, logic, sub-logic, and/or devices commonly found in a computing device, which are not illustrated in FIG. 2 for clarity of the description.

[0037] In the illustrative environment 200, the source computing device 102 further includes digital content data 202, input data 204, and encoder data 206, each of which may be stored in a memory and/or data storage device of the source computing device 102. Further, each of the digital content data 202, the input data 204, and the encoder data 206 may be accessed by the various components of the source computing device 102. Additionally, it should be appreciated that in some embodiments the data stored in, or otherwise represented by, each of the digital content data 202, the input data 204, and/or the encoder data 206 may not be mutually exclusive relative to each other. For example, in some implementations, data stored in the digital content data 202 may also be stored as a portion of one or more of the input data 204 and the encoder data 206, or vice versa. As such, although the various data utilized by the source computing device 102 is described herein as particular discrete data, such data may be combined, aggregated, and/or otherwise form portions of a single or multiple data sets, including duplicative copies, in other embodiments.

[0038] The communication manager 210, which may be embodied as hardware, firmware, software, virtualized hardware, emulated architecture, and/or a combination thereof as discussed above, is configured to manage (e.g., setup, maintain, etc.) connection paths between the source computing device 102 and other computing devices (e.g., the destination computing device 106). Additionally, the communication manager 210 is configured to facilitate inbound and outbound wired and/or wireless communications (e.g., network traffic, network packets, network flows, etc.) to and from the source computing device 102.

[0039] To do so, the communication manager 210 is configured to receive and process network packets from other computing devices (e.g., the destination computing device 106 and/or other computing device(s) communicatively coupled to the source computing device 102). Additionally, the communication manager 210 is configured to prepare and transmit network packets to another computing device (e.g., the destination computing device 106 and/or other computing device(s) communicatively coupled to the source computing device 102). The illustrative communication manager 210 includes an out-of-band communication manager 212 configured to manage out-of-band communication data flows across out-of-band communication channels (e.g., NFC, USB, etc.), such as may be used to transmit/receive the input characteristic data described herein.

[0040] The capability exchange negotiator 220, which may be embodied as hardware, firmware, software, virtualized hardware, emulated architecture, and/or a combination thereof as discussed above, is configured to perform the capability exchange negotiations between the source computing device 102 and other computing devices (e.g., the destination

computing device 106). It should be appreciated that the capability exchange negotiator 220 may be configured to perform the capability exchange during setup of the wireless communication channel 104.

[0041] The illustrative capability exchange negotiator 220 includes an input compute offload exchange negotiator 222 that is configured to determine whether the destination computing device 106 supports input compute offloading (see also, e.g., the communication flow 600 of FIG. 6 described below). In other words, the input compute offload exchange negotiator 222 is configured to perform a capability exchange with the destination computing device 106 to determine whether the destination computing device 106 supports input compute offloading (i.e., can detect user inputs and translate the detected inputs into input characteristics translatable by the source computing device 102), such as may be included in a particular header field or payload of the network packets transmitted from the destination computing device 106. For example, an input compute offload capability indicator may be any type of data that indicates whether the respective computing device is configured to support input compute offload capability, such as a Boolean value, for example. In such an embodiment, a not supported value, or value of “0”, may be used to indicate that input compute offload capability is not supported and a supported value, or value of “1”, may be used to indicate that input compute offload capability is supported.

[0042] For example, in an embodiment using the RTSP protocol to exchange computing device capabilities, the input compute offload capability indicator may be associated with an RTSP parameter (e.g., an “input_compute_offload_support” parameter) to be sent with a request message from the source computing device 102 and received with a response from the destination computing device 106 during initial configuration (i.e., negotiation and exchange of various parameters) of a communication channel (e.g., the wireless communication channel 104 of FIG. 1) between the source computing device 102 and the destination computing device 106. In some embodiments, whether the source computing device 102 supports input compute offloading, which inputs (i.e., input characteristics) are supported by the source computing device 102, whether the destination computing device 106 supports input compute offloading, and/or which inputs are supported by the destination computing device 106 may be stored in the input data 204.

[0043] It should also be appreciated that, in some embodiments, one or both of the source computing device 102 and the destination computing device 106 may support more than one set of input characteristics. In such embodiments, the capability exchange may further include a negotiation between the source computing device 102 and the destination computing device 106 to negotiate which input characteristics are supported and which of the

supported input characteristics are to be used during a particular digital content transmission session. Accordingly, in such embodiments, the supported input characteristics (e.g., of the source computing device 102 and/or the destination computing device 106) and/or which of the supported input characteristics are determined to be used during the particular streaming session may be stored in the input data 204.

[0044] The digital content adjustment manager 230, which may be embodied as hardware, firmware, software, virtualized hardware, emulated architecture, and/or a combination thereof as discussed above, is configured to output digital content from the source computing device 102 to a communicatively coupled destination computing device 106. To do so, the illustrative digital content adjustment manager 230 includes a digital content processing manager 232, an input characteristics identifier 234, and a digital content adjustment manager 236. It should be appreciated that each of the digital content processing manager 232, the input characteristics identifier 234, and/or the digital content adjustment manager 236 of the streaming packet manager 230 may be separately embodied as hardware, firmware, software, virtualized hardware, emulated architecture, and/or a combination thereof. For example, the digital content processing manager 232 may be embodied as a hardware component, while the input characteristics identifier 234 and/or the digital content adjustment manager 236 is embodied as a virtualized hardware component or as some other combination of hardware, firmware, software, virtualized hardware, emulated architecture, and/or a combination thereof.

[0045] The digital content processing manager 232 is configured to encode a frame of digital content (i.e., using an encoder of the source computing device 102) to be transmitted to the destination computing device 106 for display on the destination computing device 106. In some embodiments, the digital content to be streamed may be stored in the digital content data 202. Additionally or alternatively, in some embodiments, information associated with the encoder (e.g., which encoders/decoders are supported by the source computing device 102 and/or the destination computing device 106) may be stored in the encoder data 206. It should be appreciated that data of a frame of digital content may have a size that is too large to attach as a single payload of a network packet based on transmission size restrictions of the source computing device 102 and/or the destination computing device 106. For example, the frame size may be larger than a predetermined maximum transmission unit.

[0046] Accordingly, the digital content processing manager 232 (e.g., a packetizer of the source computing device 102) is configured to packetize the frame (i.e., the encoded frame) into a plurality of chunks, the total of which may be determined by a function of a total size of the frame and the predetermined maximum transmission unit size. Additionally,

the digital content processing manager 232 is configured to attach a header including identifying information to each of the chunks, forming a sequence of network packets for transmission to the destination computing device 106. Such packetization results in a first network packet that includes the first chunk of data, a number of intermediate network packets that include the intermediate chunks of frame data, and a last network packet that includes the last chunk of frame data required to be received by the destination computing device 106 (i.e., the end of the frame) before the destination computing device 106 can decode the frame based on the received chunks of the frame. The input characteristics identifier 234 is configured to identify input characteristics received from the destination computing device 106 and the digital content adjustment manager 236 is configured to adjust the digital content based on the identified input characteristics.

[0047] Referring now to FIG. 3, in an illustrative embodiment, the destination computing device 106 establishes an environment 300 during operation. The illustrative environment 300 includes a communication manager 310, a capability exchange negotiator 320, a digital content display manager 330, and an input detector 340. The various components of the environment 300 may be embodied as hardware, firmware, software, or a combination thereof. Additionally, in some embodiments, one or more of the illustrative components may form a portion of another component and/or one or more of the illustrative components may be independent of one another. Further, in some embodiments, one or more of the components of the environment 300 may be embodied as virtualized hardware components or emulated architecture, which may be established and maintained by the one or more processors and/or other hardware components of the source computing device 102.

[0048] In an illustrative embodiment, as shown in FIG. 5, one or more of the components of the environment 300 may be embodied as circuitry, physical hardware components, and/or a collection of electrical devices. For example, as shown, the environment 300 may include a communication management circuit 310, a capability exchange negotiation circuit 320, a digital content display circuit 330, and/or an input detection circuit 340. In such embodiments, each circuit 310, 320, 330, 340 may be embodied as a dedicated circuit/hardware component or be embodied as a portion of another hardware component of the source computing device 102. For example, in some embodiments, one or more of the communication management circuit 310, the capability exchange negotiation circuit 320, the digital content display circuit 330, and the input detection circuit 340 may form a portion of the one or more of the processor 130, the I/O subsystem 132, the GPU 136, the communication circuitry 140, and/or other components of the destination computing device 106.

[0049] Additionally or alternatively, one or more of the communication management circuit 310, the capability exchange negotiation circuit 320, the digital content display circuit 330, and/or the input detection circuit 340 may be implemented as special purpose hardware circuits or components. Such dedicated or special purpose hardware circuits or logic may complement certain software functions, which may facilitate the calling of such functions by various software programs or applications executed by the destination computing device 106 to complete one or more tasks.

[0050] Referring again to FIG. 3, in the illustrative environment 300, the destination computing device 106 further includes digital content data 302, input data 304, and decoder data 306, each of which may be stored in a memory and/or data storage device of the destination computing device 106. Further, each of the digital content data 302, the input data 304, and the decoder data 306 may be accessed by the various components of the destination computing device 106. Additionally, it should be appreciated that in some embodiments the data stored in, or otherwise represented by, each of the digital content data 302, the input data 304, and/or the decoder data 306 may not be mutually exclusive relative to each other.

[0051] For example, in some implementations, data stored in the digital content data 302 may also be stored as a portion of one or more of the input data 304 and the decoder data 306, or vice versa. As such, although the various data utilized by the destination computing device 106 is described herein as particular discrete data, such data may be combined, aggregated, and/or otherwise form portions of a single or multiple data sets, including duplicative copies, in other embodiments. It should be further appreciated that the destination computing device 106 may include additional and/or alternative components, sub-components, modules, sub-modules, and/or devices commonly found in a computing device, which are not illustrated in FIG. 3 for clarity of the description.

[0052] The communication manager 310, which may be embodied as hardware, firmware, software, virtualized hardware, emulated architecture, and/or a combination thereof as discussed above, is configured to manage (e.g., setup, maintain, etc.) connection paths between the source computing device 102 and other computing devices (e.g., the destination computing device 106). Additionally, the communication manager 310 is configured to facilitate inbound and outbound wired and/or wireless communications (e.g., network traffic, network packets, network flows, etc.) to and from the source computing device 102.

[0053] To do so, the communication manager 310 is configured to receive and process network packets from other computing devices (e.g., the destination computing

device 106 and/or other computing device(s) communicatively coupled to the destination computing device 106). Additionally, the communication manager 310 is configured to prepare and transmit network packets to another computing device (e.g., the source computing device 102 and/or other computing device(s) communicatively coupled to the destination computing device 106). The illustrative communication manager 310 includes an out-of-band communication manager 312 configured to manage out-of-band communication data flows across out-of-band communication channels (e.g., as may be managed by the capability exchange negotiator 320), such as may be used to transmit/receive the input characteristic data described herein.

[0054] The capability exchange negotiator 320, which may be embodied as hardware, firmware, software, virtualized hardware, emulated architecture, and/or a combination thereof as discussed above, is configured to perform the capability exchange negotiations between the destination computing device 106 and other computing devices (e.g., the source computing device 102). It should be appreciated that the capability exchange negotiator 320 may be configured to perform the capability exchange during setup of the wireless communication channel 104.

[0055] The illustrative capability exchange negotiator 320 includes an input compute offload exchange negotiator 322 that is configured to determine whether the source computing device 102 supports input compute offloading (see also, e.g., the communication flow 600 of FIG. 6 described below). In other words, the input compute offload exchange negotiator 320 is configured to perform a capability exchange with the source computing device 102 to determine whether the source computing device 102 supports input compute offloading (i.e., can translate input characteristics received from the source computing device 102), such as may be included in a particular header field or payload of the network packets transmitted to the source computing device 102. For example, an input compute offload capability indicator may be any type of data that indicates whether the respective computing device is configured to support input compute offload capability, such as a Boolean value, for example. In such an embodiment, a not supported value, or value of “0”, may be used to indicate that input compute offload capability is not supported and a supported value, or value of “1”, may be used to indicate that input compute offload capability is supported.

[0056] For example, in an embodiment using the RTSP protocol to exchange computing device capabilities, the input compute offload capability indicator may be associated with an RTSP parameter (e.g., an “input_compute_offload_support” parameter) received with a request message from the source computing device 102 and sent with a response from the destination computing device 106 during initial configuration (i.e.,

negotiation and exchange of various parameters) of a communication channel (e.g., the wireless communication channel 104 of FIG. 1) between the source computing device 102 and the destination computing device 106. In some embodiments, whether the source computing device 102 supports input compute offloading, which inputs (i.e., input characteristics) are supported by the source computing device 102, whether the destination computing device 106 supports input compute offloading, and/or which inputs are supported by the destination computing device 106 may be stored in the input data 204.

[0057] It should also be appreciated that, in some embodiments, one or both of the source computing device 102 and the destination computing device 106 may support more than one set of input characteristics. In such embodiments, the capability exchange may further include a negotiation between the source computing device 102 and the destination computing device 106 to negotiate which input characteristics are supported and which of the supported input characteristics are to be used during a particular digital content transmission session. Accordingly, in such embodiments, the supported input characteristics (e.g., of the source computing device 102 and/or the destination computing device 106) and/or which of the supported input characteristics are determined to be used during the particular streaming session may be stored in the input data 204.

[0058] The digital content display manager 330, which may be embodied as hardware, firmware, software, virtualized hardware, emulated architecture, and/or a combination thereof as discussed above, is configured to display digital content received from the communicatively coupled source computing device 102. To do so, the digital content display manager 330 is configured to depacketize received network packets (i.e., one or more network packets including at least a portion of data corresponding to a frame). For example, the digital content display manager 330 may be configured to strip the headers (i.e., the MPEG2-TS headers) from the received network packets and accumulate the payloads of the received frames of digital content. Such accumulated payloads may be stored in the digital content data 304, in some embodiments.

[0059] Accordingly, the digital content display manager 330 is further configured to decode the accumulated payloads (i.e., at the GPU 136 of the destination computing device 106 of FIG. 1) and render the decoded frame for output at an output device (e.g., a display) of the destination computing device 106. In some embodiments, information associated with the decoder (e.g., which encoders/decoders are supported by the source computing device 102 and/or the destination computing device 106) may be stored in the decoder data 306. Additionally or alternatively, in some embodiments, information corresponding to the decoded frame may be stored in the digital content data 302.

[0060] The input detector 340 is configured to detect input of a user of the destination computing device 106, such as may be initiated via directly touching or using a stylus on a touchscreen display, pressing a key on a keyboard, detecting movement of an element of a mouse, receiving an audible voice command by a microphone, etc. The detected input may be of any type of input in which the expected outcome of the input is to render one or more objects (e.g., text, shapes, lines, graphics, etc.) on the display of the destination computing device 106. The illustrative input detector 340 includes an input characteristics determination manager 342 configured to determine input characteristics of the detected input and an input characteristics reporting manager 344 configured to translate the determined input characteristics into information (i.e., data structures) usable by the source computing device 102 to replicate the detected input.

[0061] Referring now to FIG. 6, an embodiment of a communication flow 600 for input compute offloading capability negotiation includes the source computing device 102 and the destination computing device 106 communicatively coupled over a communication channel (e.g., the communication channel 104 of FIG. 1). The illustrative communication flow 600 includes a number of data flows, some of which may be executed separately or together, depending on the embodiment. In data flow 602, as described previously, the communication channel 104 (e.g., a TCP connection) is established between the source computing device 102 and the destination computing device 106. It should be appreciated that the establishment of the communication channel may be predicated on a distance between the source computing device 102 and the destination computing device 106. It should be further appreciated that the distance may be based on a type and communication range associated with the communication technology employed in establishing the communication channel 104.

[0062] In some embodiments, the source computing device 102 and the destination computing device 106 may have been previously connected to each other. In other words, the source computing device 102 and the destination computing device 106 may have previously exchanged pairing data, such as may be exchanged during a Wi-Fi® setup (e.g., manual entry of connection data, Wi-Fi Protected Setup (WPS), etc.) or Bluetooth® pairing (e.g., bonding). To do so, in some embodiments, the source computing device 102 or the destination computing device 106 may have been placed in a discovery mode for establishing the connection. Additionally or alternatively, in some embodiments, the source computing device 102 and the destination computing device 106 may use an out-of-band technology (e.g., NFC, USB, etc.) to transfer information by a channel other than the communication channel 104. Accordingly, it should be appreciated that, in such embodiments, the

information used to establish the communication channel 104, or the OOB channel, may be stored at the source computing device 102 and/or the destination computing device 106.

[0063] In data flow 604, the source computing device 102 transmits a message to the destination computing device 106 (e.g., using RTSP messages) that includes a request for input compute offloading detection capability of the destination computing device 106. In data flow 606, the destination computing device 106 responds to the request message received from the source computing device with a response message that includes the input compute offloading capability of the destination computing device 106. In data flow 608, the source computing device 102 saves the input compute offloading capability of the destination computing device 106 received in data flow 606.

[0064] It should be appreciated that, in some embodiments, more than one input compute offloading capability may be supported by the source computing device 102 and/or the destination computing device 106. Accordingly, as described previously, the input compute offloading capability response may include an indication as to whether the destination computing device 106 supports certain input characteristics of input compute offloading, as well as an indication as to how the destination computing device 106 supports, translates, transmits, etc. such input characteristics (e.g., a particular field of a header message, a particular designator in a payload of a message, etc.). In such embodiments, a negotiation flow may be performed between the source computing device 102 and the destination computing device 106 to establish which of the supported input compute offloading capabilities will be used during the streaming session.

[0065] In data flow 610, the destination computing device 106 transmits a message to the source computing device 102 that includes a request for input compute offloading capability of the source computing device 102. In data flow 612, the source computing device 102 responds to the request message with a response message that includes the input compute offloading capability of the source computing device 102. In data flow 614, the destination computing device 106 saves the input compute offloading capability of the source computing device 102 received in data flow 612. In data flow 616, the source computing device 102 and the destination computing device 106 establish a streaming session and initiate the transmission/receipt of digital content.

[0066] Referring now to FIG. 7, in use, the source computing device 102 may execute a method 700 for input compute offloading of digital content to be transmitted to a destination computing device (e.g., the destination computing device 106 of FIG. 1). It should be appreciated that prior to execution of the method 700, a communication channel (e.g., the communication channel 104 of FIG. 1) has already been established and capabilities

have been exchanged between the source computing device 102 and the destination computing device 106 (e.g., as described in the illustrative communication flow 600 of FIG. 6). It should be further appreciated that at least a portion of method 700 may be embodied as various instructions stored on a computer-readable media, which may be executed by the processor 110, the GPU 116, the communication circuitry 120 (e.g., the NIC 122), and/or other components of the source computing device 102 to cause the source computing device 102 to perform the method 600. The computer-readable media may be embodied as any type of media capable of being read by the source computing device 102 including, but not limited to, the memory 114, the data storage device 118, a local memory (not shown) of the NIC 122, other memory or data storage devices of the source computing device 102, portable media readable by a peripheral device of the source computing device 102, and/or other media.

[0067] The method 700 begins in block 702, in which the source computing device 102 determines whether to transmit digital content to the destination computing device 106 (e.g., streaming content from the source computing device, mirroring content presently being displayed on the source computing device 102, casting content from the source computing device, etc.). If the source computing device 102 determines not to transmit digital content to the destination computing device 106 (e.g., digital content stored on the source computing device 102 has not yet been selected for transmission), the method 700 returns to block 702 to continue to monitor whether to transmit the digital content. It should be appreciated that in some embodiments any applications (e.g., operating system, software applications, etc.) presently being displayed (e.g., via a graphical user interface (GUI) of the source computing device 102) may be transmitted to the destination computing device 106 in the form of digital content (e.g., frames of the presently displayed screen of the source computing device 102)

[0068] Otherwise, the method 700 advances to block 704, in which the source computing device 102 processes digital content for transmission to the destination computing device 106. For example, the source computing device 102 may encode the digital content (i.e., the frames of the digital content), such as by using an RTSP encoder, and packetize the encoded frame into a streaming packet for transmission (e.g., chunking the frame and affixing each chunk as a streaming packet payload with a header). In block 706, the source computing device 102 transmits one or more of the processed streaming packets to the destination computing device 106 (e.g., via a queue of network packets, messages, etc.). As discussed above, in other embodiments, the source computing device 102 may transmit the digital content using other transmission modalities including, but not limited to, mirroring of

the digital content, casting of the digital content, and/or other digital content transmission technique.

[0069] In block 708, the source computing device 102 determines whether any input characteristics have been received from the destination computing device 106. If not, the method 700 loops back to block 702 to determine whether to continue transmitting digital content to the destination computing device 106; otherwise, the method 700 advances to block 710. In some embodiments, the operating system of the source computing device 102 may receive the indication from the destination computing device 106 and subsequently notify any listening application of the received input characteristics using the same or similar notification methodologies as the destination computing device 106 would use to notify any listening application of an input detected local to the source computing device 102.

[0070] In block 710, the source computing device 102 identifies the input characteristics received from the destination computing device 106. In block 712, the source computing device 102 renders one or more objects to a display (e.g., via the GPU 116 of FIG. 1) based on the input characteristics for output to a display of the source computing device 102. For example, the source computing device 102 may capture an image of the content presently displayed on the display of the source computing device 102 (i.e., including the rendered output of the input characteristics). In some embodiments, the captured image may be compressed as a video stream and transmitted to the destination computing device 106. In some embodiments, the source computing device 102 may additionally transmit an indication to the destination computing device 102 that is usable to identify that the received digital content now includes the received input characteristics.

[0071] Referring now to FIG. 8, in use, the destination computing device 106 may execute a method 800 for input compute offloading of digital content being transmitted from a source computing device (e.g., the source computing device 106 of FIG. 1). It should be appreciated that a communication channel (e.g., the communication channel 104 of FIG. 1) has been established between the destination computing device 106 and the source computing device 102. It should be further appreciated that at least a portion of the method 800 may be embodied as various instructions stored on a computer-readable media, which may be executed by the processor 130, the GPU 136, the communication circuitry 140 (e.g., the NIC 142), and/or other components of the destination computing device 106 to cause the destination computing device 106 to perform the method 800. The computer-readable media may be embodied as any type of media capable of being read by the destination computing device 106 including, but not limited to, the memory 134, the data storage device 138, a local memory of the NIC 142 (not shown), other memory or data storage devices of the destination

computing device 106, portable media readable by a peripheral device of the destination computing device 106, and/or other media.

[0072] The method 800 begins in block 802, in which the destination computing device 106 determines whether a network packet that includes digital content (e.g., a frame of digital content) to be rendered by the destination computing device 106 has been received from the source computing device 102. If so, the method 800 advances to block 804, in which the destination computing device 106 processes (e.g., depacketizes, decodes, etc.) the received network packet. In block 806, the destination computing device 106 renders the processed digital content for display to an output device (e.g., one of the peripheral devices 144) of the destination computing device 106. To do so, in some embodiments, the GPU 136 may provide the rendered frame to the output device of the destination computing device 106 for display of video content on a display of the destination computing device 106 or produce audible sound of audio content from a speaker of the destination computing device 106, for example.

[0073] In block 808, the destination computing device 106 determines whether any user input has been detected (e.g., via directly touching or using a stylus on a touchscreen display, pressing a key on a keyboard, detecting movement of an element of a mouse, receiving an audible voice command by a microphone, etc.) such that an action (e.g., drawing an object, typing text, etc.) on the digital content being displayed on the destination computing device 106 is expected to be seen on the display of the destination computing device 106. For example, such inputs may be detected via a touch sensor/display and transmitted to the operating system for processing at the processor (e.g., the processor 130 of FIG. 1) or the GPU (e.g., the GPU 136 of FIG. 1) for touch processing. If no user input has been detected, the method 800 returns to block 802 in which the destination computing device 106 determines whether another network packet that includes digital content for output from the destination computing device 106 has been received; otherwise, the method 800 advances to block 810.

[0074] In block 810, the destination computing device 106 identifies the input characteristics of the detected user input. In some embodiments, the input characteristics may be identified via a vendor or operating system provided middleware (e.g., Windows Direct Inking framework). Accordingly, the input characteristics supported may be based on the capabilities of the middleware to translate the input into determinable characteristics, such as input coordinates (e.g., screen/display coordinates, output content coordinates, input border coordinates, etc.), input types (e.g., text, shape, etc.), font types, styles, sizes, colors, weights, etc. In such embodiments, the destination computing device 106 stack may program

(e.g., through operating system provided hooks) lower level kernel operations (e.g., fast-linking) running on the GPU (e.g., the GPU 136 of FIG. 1) to store the capabilities and/or preferences. In some embodiments, prior to the operating system processing the input, a fast-link kernel running on the GPU 136 may pass the input data to graphics shaders via a methodology that allows the destination computing device 106 to render the temporary overlay based on the input data. In block 812, the destination computing device 106 displays (i.e., renders and outputs) a temporary overlay displaying a result (e.g., object, text, etc.) of the detected user input and the associated input characteristics.

[0075] In block 814, the destination computing device 106 transmits the identified input characteristics to the source computing device 102 from which the digital content is being received (i.e., after the operating system processes the input data). It should be appreciated that, in some embodiments, the destination computing device 106 may continue to render the temporary overlay until it receives an indication from the source computing device 102 indicating the digital content now includes the intended object(s). Additionally or alternatively, in some embodiments, the temporary overlay may timeout, or otherwise only be rendered by the destination computing device 106 for a certain duration of time. For example, in some embodiments, the temporary overlay may be removed after a fixed or variable period of time, such as may be determined subsequent to the source computing device 102 having transmitted the frames that contain input related to the digital content.

EXAMPLES

[0076] Illustrative examples of the technologies disclosed herein are provided below. An embodiment of the technologies may include any one or more, and any combination of, the examples described below.

[0077] Example 1 includes a destination computing device for input compute offloading of digital content, the destination computing device comprising a digital content display manager to output digital content received from a wirelessly coupled source computing device to a display of the destination computing device; and an input detector to (i) detect an input by an input device of the destination computing device and (ii) identify one or more input characteristics of the detected input, wherein the input characteristics define information related to the detected input usable to determine an action to be taken subsequent to the input detection, wherein the digital content display manager is further to display, in response to detection of the input, a temporary overlay on the display of the destination computing device based on the detected input; and further comprising a communication manager to transmit the one or more input characteristics to the source computing device,

wherein the one or more input characteristics are usable to render the digital content transmitted by the source computing device to include one or more objects based on the one or more input characteristics.

[0078] Example 2 includes the subject matter of Example 1, and wherein the digital content comprises a video stream composed of a plurality of captured images of a display of the source computing device, wherein each of the captured images includes a screen capture of at least a portion of the display of the source computing device at the time in which the image was captured.

[0079] Example 3 includes the subject matter of any of Examples 1 and 2, and wherein the input characteristics include one or more of a display coordinate, an output content coordinate, an input border coordinate, a text characteristic, a shape characteristic, a font characteristic, or a line characteristic.

[0080] Example 4 includes the subject matter of any of Examples 1-3, and wherein to transmit the input characteristics to the source computing device comprises to transmit the input characteristics via an out-of-band communication channel.

[0081] Example 5 includes the subject matter of any of Examples 1-4, and further comprising a capability exchange negotiator to (i) exchange input compute offloading capabilities between the source computing device and the destination computing device and (ii) determine which of the exchanged input compute offloading capabilities are to be used during the transmission of the digital content.

[0082] Example 6 includes the subject matter of any of Examples 1-5, and wherein to detect the input comprises to detect one of a finger movement on a touchscreen display of the destination computing device, a stylus movement on the touchscreen display, a key press on a keyboard of the destination computing device, a movement of an element of a mouse of the destination computing device, or an audible voice command by a microphone of the destination computing device.

[0083] Example 7 includes the subject matter of any of Examples 1-6, and wherein the action includes outputting one or more objects to the display via the temporary overlay.

[0084] Example 8 includes the subject matter of any of Examples 1-7, and wherein the one or more objects includes one or more of a text character, a shape, a line, or a graphic.

[0085] Example 9 includes the subject matter of any of Examples 1-8, and wherein the action includes to changing a setting of an application presently executing on the source computing device that corresponds to the digital content received from the source computing device.

[0086] Example 10 includes the subject matter of any of Examples 1-9, and wherein the digital content display manager is further to remove the temporary overlay after an elapsed period of time subsequent to the display of the temporary overlay.

[0087] Example 11 includes the subject matter of any of Examples 1-10, and wherein the digital content comprises digital content mirrored from the source computing device.

[0088] Example 12 includes the subject matter of any of Examples 1-11, and wherein the digital content comprises digital content cast from the source computing device.

[0089] Example 13 includes a method for input compute offloading of digital content, the method comprising outputting, by a destination computing device, digital content received from a wirelessly coupled source computing device to a display of the destination computing device; detecting, by the destination computing device, an input by an input device of the destination computing device; identifying, by the destination computing device, one or more input characteristics of the detected input, wherein the input characteristics define information related to the detected input usable to determine an action to be taken subsequent to the input detection; displaying, by the destination computing device and in response to detection of the input, a temporary overlay on the display of the destination computing device based on the detected input; and transmitting, by the destination computing device, the one or more input characteristics to the source computing device, wherein the one or more input characteristics are usable to render the digital content transmitted by the source computing device to include one or more objects based on the one or more input characteristics.

[0090] Example 14 includes the subject matter of Example 13, and wherein outputting the digital content comprises outputting a video stream composed of a plurality of captured images of a display of the source computing device, wherein each of the captured images includes a screen capture of at least a portion of the display of the source computing device at the time in which the image was captured.

[0091] Example 15 includes the subject matter of any of Examples 13 and 14, and wherein transmitting the input characteristics includes transmitting one or more of a display coordinate, an output content coordinate, an input border coordinate, a text characteristic, a shape characteristic, a font characteristic, or a line characteristic.

[0092] Example 16 includes the subject matter of any of Examples 13-15, and wherein transmitting the input characteristics to the source computing device comprises transmitting the input characteristics via an out-of-band communication channel.

[0093] Example 17 includes the subject matter of any of Examples 13-16, and comprising: exchanging, by the destination computing device, input compute offloading

capabilities between the source computing device and the destination computing device; and determining, by the destination computing device, which of the exchanged input compute offloading capabilities are to be used during the transmission of the digital content.

[0094] Example 18 includes the subject matter of any of Examples 13-17, and wherein detecting the input comprises detecting one of a finger movement on a touchscreen display of the destination computing device, a stylus movement on the touchscreen display, a key press on a keyboard of the destination computing device, a movement of an element of a mouse of the destination computing device, or an audible voice command by a microphone of the destination computing device.

[0095] Example 19 includes the subject matter of any of Examples 13-18, and wherein the action includes outputting one or more objects to the display via the temporary overlay.

[0096] Example 20 includes the subject matter of any of Examples 13-19, and wherein outputting the one or more objects comprises outputting one or more of a text character, a shape, a line, or a graphic.

[0097] Example 21 includes the subject matter of any of Examples 13-20, and wherein the action includes changing a setting of an application presently executing on the source computing device that corresponds to the digital content received from the source computing device.

[0098] Example 22 includes the subject matter of any of Examples 13-21, and further comprising removing the temporary overlay subsequent to an elapsed period of time after display of the temporary overlay.

[0099] Example 23 includes the subject matter of any of Examples 13-22, and wherein the digital content comprises digital content mirrored from the source computing device.

[00100] Example 24 includes the subject matter of any of Examples 13-23, and wherein the digital content comprises digital content cast from the source computing device.

[00101] Example 25 includes a destination computing device comprising a processor; and a memory having stored therein a plurality of instructions that when executed by the processor cause the destination computing device to perform the method of any of Examples 13-24.

[00102] Example 26 includes one or more machine readable storage media comprising a plurality of instructions stored thereon that in response to being executed result in a destination computing device performing the method of any of Examples 13-24.

[00103] Example 27 includes a destination computing device for input compute offloading of digital content, the destination computing device comprising means for outputting digital content received from a wirelessly coupled source computing device to a display of the destination computing device; means for detecting an input by an input device of the destination computing device; means for identifying one or more input characteristics of the detected input, wherein the input characteristics define information related to the detected input usable to determine an action to be taken subsequent to the input detection; means for displaying, in response to detection of the input, a temporary overlay on the display of the destination computing device based on the detected input; and means for transmitting the one or more input characteristics to the source computing device, wherein the one or more input characteristics are usable to render the digital content transmitted by the source computing device to include one or more objects based on the one or more input characteristics.

[00104] Example 28 includes the subject matter of Example 27, and wherein the means for outputting the digital content comprises means for outputting a video stream composed of a plurality of captured images of a display of the source computing device, wherein each of the captured images includes a screen capture of at least a portion of the display of the source computing device at the time in which the image was captured.

[00105] Example 29 includes the subject matter of any of Examples 27 and 28, and wherein the means for transmitting the input characteristics includes means for transmitting one or more of a display coordinate, an output content coordinate, an input border coordinate, a text characteristic, a shape characteristic, a font characteristic, or a line characteristic.

[00106] Example 30 includes the subject matter of any of Examples 27-29, and wherein the means for transmitting the input characteristics to the source computing device comprises means for transmitting the input characteristics via an out-of-band communication channel.

[00107] Example 31 includes the subject matter of any of Examples 27-30, and further comprising means for exchanging input compute offloading capabilities between the source computing device and the destination computing device; and means for determining which of the exchanged input compute offloading capabilities are to be used during the transmission of the digital content.

[00108] Example 32 includes the subject matter of any of Examples 27-31, and wherein the means for detecting the input comprises means for detecting one of a finger movement on a touchscreen display of the destination computing device, a stylus movement on the touchscreen display, a key press on a keyboard of the destination computing device, a

movement of an element of a mouse of the destination computing device, or an audible voice command by a microphone of the destination computing device.

[00109] Example 33 includes the subject matter of any of Examples 27-32, and wherein the action includes means for outputting one or more objects to the display via the temporary overlay.

[00110] Example 34 includes the subject matter of any of Examples 27-33, and wherein the means for outputting the one or more objects comprises means for outputting one or more of a text character, a shape, a line, or a graphic.

[00111] Example 35 includes the subject matter of any of Examples 27-34, and wherein the action includes means for changing a setting of an application presently executing on the source computing device that corresponds to the digital content received from the source computing device.

[00112] Example 36 includes the subject matter of any of Examples 27-35, and further comprising means for removing the temporary overlay subsequent to an elapsed period of time after display of the temporary overlay.

[00113] Example 37 includes the subject matter of any of Examples 27-36, and wherein the digital content comprises digital content mirrored from the source computing device.

[00114] Example 38 includes the subject matter of any of Examples 27-37, and wherein the digital content comprises digital content cast from the source computing device.

[00115] Example 39 includes a source computing device for input compute offloading of digital content, the source computing device comprising a digital content adjustment manager to transmit digital content to a destination computing device wirelessly coupled to the source computing device; a communication manager to receive one or more input characteristics from the destination computing device, wherein the input characteristics define one or more characteristics of an input initiated by a user on a display of the destination computing device; and a digital content adjustment manager to render the digital content to include one or more objects based on the one or more input characteristics.

[00116] Example 40 includes the subject matter of Example 39, and wherein the digital content comprises a video stream composed of a plurality of screen capture images of the source computing device, wherein each of the screen capture images includes a screen capture of a display of the source computing device at the time in which the display was captured.

[00117] Example 41 includes the subject matter of any of Examples 39 and 40, and wherein the input characteristics include one or more of a display coordinate, an output

content coordinate, an input border coordinate, a text characteristic, a shape characteristic, a font characteristic, or a line characteristic.

[00118] Example 42 includes the subject matter of any of Examples 39-41, and wherein the input characteristics are received from the destination computing device via an out-of-band communication channel.

[00119] Example 43 includes the subject matter of any of Examples 39-42, and further comprising a capability exchange negotiator to (i) exchange input compute offloading capabilities between the source computing device and the destination computing device and (ii) determine which of the exchanged input compute offloading capabilities are to be used during the transmission of the digital content.

[00120] Example 44 includes the subject matter of any of Examples 39-43, and wherein the digital content comprises digital content mirrored from the source computing device.

[00121] Example 45 includes the subject matter of any of Examples 39-44, and wherein the digital content comprises digital content cast from the source computing device.

[00122] Example 46 includes a method for input compute offloading of digital content, the method comprising transmitting, by a source computing device, digital content to a destination computing device wirelessly coupled to the source computing device; receiving, by the source computing device, one or more input characteristics from the destination computing device, wherein the input characteristics define characteristics of an input initiated by a user on a display of the destination computing device; and rendering, by the source computing device, the digital content to include one or more objects based on the one or more input characteristics.

[00123] Example 47 includes the subject matter of Example 46, and wherein transmitting the digital content comprises transmitting a video stream composed of a plurality of screen capture images of the source computing device, wherein each of the screen capture images includes a screen capture of a display of the source computing device at the time in which the display was captured.

[00124] Example 48 includes the subject matter of any of Examples 46 and 47, and wherein receiving the input characteristics from the destination computing device comprises receiving one or more of a display coordinate, an output content coordinate, an input border coordinate, a text characteristic, a shape characteristic, a font characteristic, or a line characteristic.

[00125] Example 49 includes the subject matter of any of Examples 46-48, and wherein receiving the input characteristics from the destination computing device comprises receiving the input characteristics via an out-of-band communication channel.

[00126] Example 50 includes the subject matter of any of Examples 46-49, and further comprising exchanging, by the source computing device, input compute offloading capabilities between the source computing device and the destination computing device; and determining, by the source computing device, which of the exchanged input compute offloading capabilities are to be used during the transmission of the digital content.

[00127] Example 51 includes the subject matter of any of Examples 46-50, and wherein the digital content comprises digital content mirrored from the source computing device.

[00128] Example 52 includes the subject matter of any of Examples 46-51, and wherein the digital content comprises digital content cast from the source computing device.

[00129] Example 53 includes a source computing device comprising a processor; and a memory having stored therein a plurality of instructions that when executed by the processor cause the source computing device to perform the method of any of Examples 46-52.

[00130] Example 54 includes one or more machine readable storage media comprising a plurality of instructions stored thereon that in response to being executed result in a source computing device performing the method of any of Examples 46-52.

[00131] Example 55 includes a source computing device for input compute offloading of digital content, the source computing device comprising means for transmitting digital content to a destination computing device wirelessly coupled to the source computing device; means for receiving one or more input characteristics from the destination computing device, wherein the input characteristics define characteristics of an input initiated by a user on a display of the destination computing device; and means for rendering the digital content to include one or more objects based on the one or more input characteristics.

[00132] Example 56 includes the subject matter of Example 55, and wherein the means for transmitting the digital content comprises means for transmitting a video stream composed of a plurality of screen capture images of the source computing device, wherein each of the screen capture images includes a screen capture of a display of the source computing device at the time in which the display was captured.

[00133] Example 57 includes the subject matter of any of Examples 55 and 56, and wherein the means for receiving the input characteristics from the destination computing device comprises means for receiving one or more of a display coordinate, an output content

coordinate, an input border coordinate, a text characteristic, a shape characteristic, a font characteristic, or a line characteristic.

[00134] Example 58 includes the subject matter of any of Examples 55-57, and wherein the means for receiving the input characteristics from the destination computing device comprises means for receiving the input characteristics via an out-of-band communication channel.

[00135] Example 59 includes the subject matter of any of Examples 55-58, and further comprising means for exchanging input compute offloading capabilities between the source computing device and the destination computing device; and means for determining which of the exchanged input compute offloading capabilities are to be used during the transmission of the digital content.

[00136] Example 60 includes the subject matter of any of Examples 55-59, and wherein the digital content comprises digital content mirrored from the source computing device.

[00137] Example 61 includes the subject matter of any of Examples 55-60, and wherein the digital content comprises digital content cast from the source computing device.

WHAT IS CLAIMED IS:

1. A destination computing device for input compute offloading of digital content, the destination computing device comprising:

a digital content display manager to output digital content received from a wirelessly coupled source computing device to a display of the destination computing device; and

an input detector to (i) detect an input by an input device of the destination computing device and (ii) identify one or more input characteristics of the detected input, wherein the input characteristics define information related to the detected input usable to determine an action to be taken subsequent to the input detection,

wherein the digital content display manager is further to display, in response to detection of the input, a temporary overlay on the display of the destination computing device based on the detected input; and

further comprising a communication manager to transmit the one or more input characteristics to the source computing device, wherein the one or more input characteristics are usable to render the digital content transmitted by the source computing device to include one or more objects based on the one or more input characteristics.

2. The destination computing device of claim 1, wherein the digital content comprises a video stream composed of a plurality of captured images of a display of the source computing device, wherein each of the captured images includes a screen capture of at least a portion of the display of the source computing device at the time in which the image was captured.

3. The destination computing device of claim 1, wherein the input characteristics include one or more of a display coordinate, an output content coordinate, an input border coordinate, a text characteristic, a shape characteristic, a font characteristic, or a line characteristic.

4. The destination computing device of claim 3, wherein to transmit the input characteristics to the source computing device comprises to transmit the input characteristics via an out-of-band communication channel.

5. The destination computing device of claim 1, further comprising a capability exchange negotiator to (i) exchange input compute offloading capabilities between the source computing device and the destination computing device and (ii) determine which of the exchanged input compute offloading capabilities are to be used during the transmission of the digital content.

6. The destination computing device of claim 1, wherein to detect the input comprises to detect one of a finger movement on a touchscreen display of the destination computing device, a stylus movement on the touchscreen display, a key press on a keyboard of the destination computing device, a movement of an element of a mouse of the destination computing device, or an audible voice command by a microphone of the destination computing device.

7. The destination computing device of claim 1, wherein the action includes outputting one or more objects to the display via the temporary overlay.

8. The destination computing device of claim 7, wherein the one or more objects includes one or more of a text character, a shape, a line, or a graphic.

9. The destination computing device of claim 1, wherein the action includes to changing a setting of an application presently executing on the source computing device that corresponds to the digital content received from the source computing device.

10. The destination computing device of claim 1, wherein the digital content display manager is further to remove the temporary overlay after an elapsed period of time subsequent to the display of the temporary overlay.

11. A method for input compute offloading of digital content, the method comprising:

outputting, by a destination computing device, digital content received from a wirelessly coupled source computing device to a display of the destination computing device;

detecting, by the destination computing device, an input by an input device of the destination computing device;

identifying, by the destination computing device, one or more input characteristics of the detected input, wherein the input characteristics define information

related to the detected input usable to determine an action to be taken subsequent to the input detection;

displaying, by the destination computing device and in response to detection of the input, a temporary overlay on the display of the destination computing device based on the detected input; and

transmitting, by the destination computing device, the one or more input characteristics to the source computing device,

wherein the one or more input characteristics are usable to render the digital content transmitted by the source computing device to include one or more objects based on the one or more input characteristics.

12. The method of claim 11, wherein outputting the digital content comprises outputting a video stream composed of a plurality of captured images of a display of the source computing device, wherein each of the captured images includes a screen capture of at least a portion of the display of the source computing device at the time in which the image was captured.

13. The method of claim 11, wherein transmitting the input characteristics includes transmitting one or more of a display coordinate, an output content coordinate, an input border coordinate, a text characteristic, a shape characteristic, a font characteristic, or a line characteristic.

14. The method of claim 11, wherein transmitting the input characteristics to the source computing device comprises transmitting the input characteristics via an out-of-band communication channel.

15. The method of claim 11, further comprising:
exchanging, by the destination computing device, input compute offloading capabilities between the source computing device and the destination computing device; and
determining, by the destination computing device, which of the exchanged input compute offloading capabilities are to be used during the transmission of the digital content.

16. The method of claim 11, wherein detecting the input comprises detecting one of a finger movement on a touchscreen display of the destination computing device, a stylus

movement on the touchscreen display, a key press on a keyboard of the destination computing device, a movement of an element of a mouse of the destination computing device, or an audible voice command by a microphone of the destination computing device.

17. The method of claim 11, wherein the action includes outputting one or more objects to the display via the temporary overlay.

18. The method of claim 17, wherein outputting the one or more objects comprises outputting one or more of a text character, a shape, a line, or a graphic.

19. The method of claim 11, wherein the action includes changing a setting of an application presently executing on the source computing device that corresponds to the digital content received from the source computing device.

20. The method of claim 11, further comprising removing the temporary overlay subsequent to an elapsed period of time after display of the temporary overlay.

21. A destination computing device comprising:
a processor; and
a memory having stored therein a plurality of instructions that when executed by the processor cause the destination computing device to perform the method of any of claims 11-20.

22. One or more machine readable storage media comprising a plurality of instructions stored thereon that in response to being executed result in a destination computing device performing the method of any of claims 11-20.

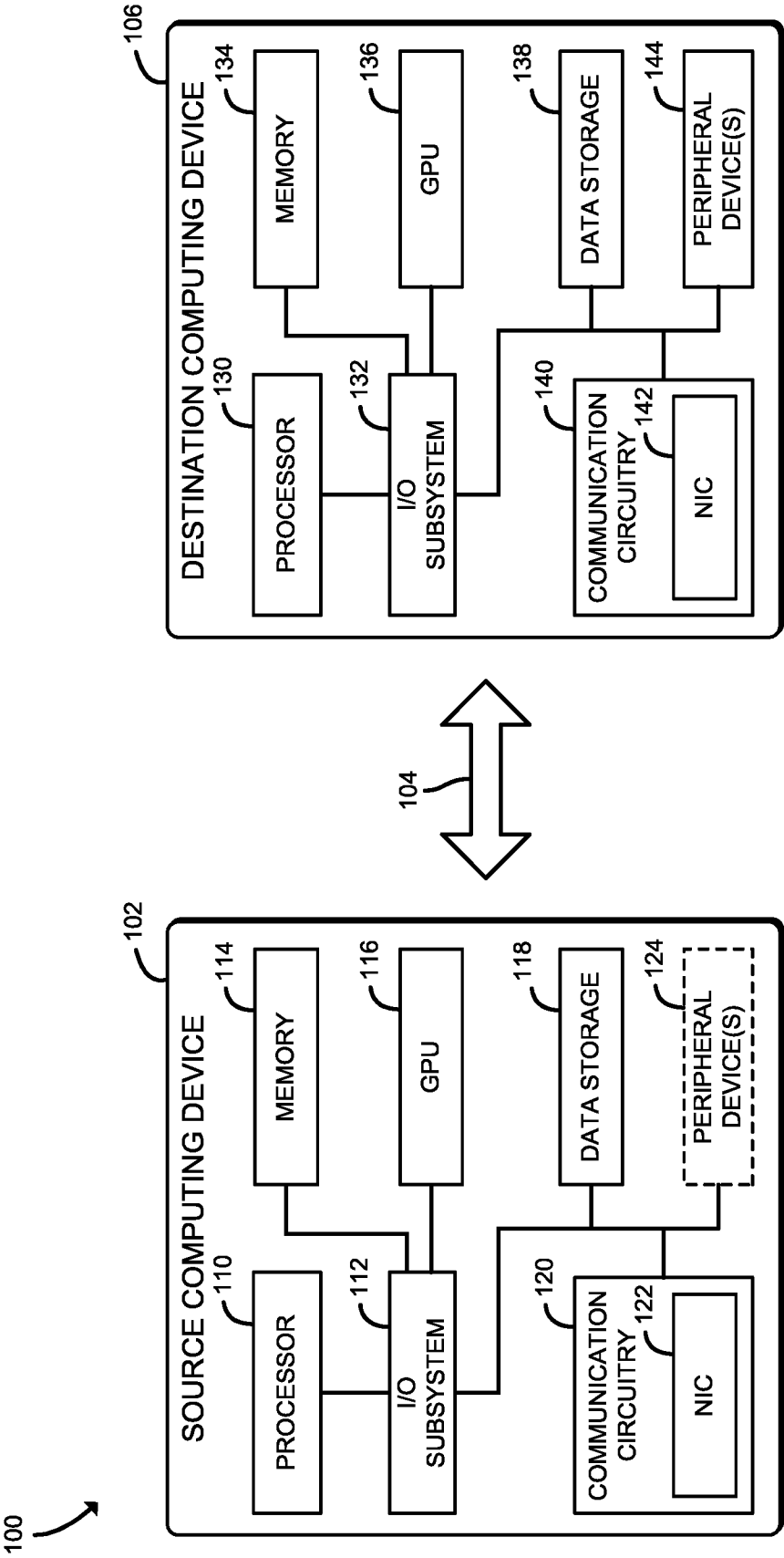


FIG. 1

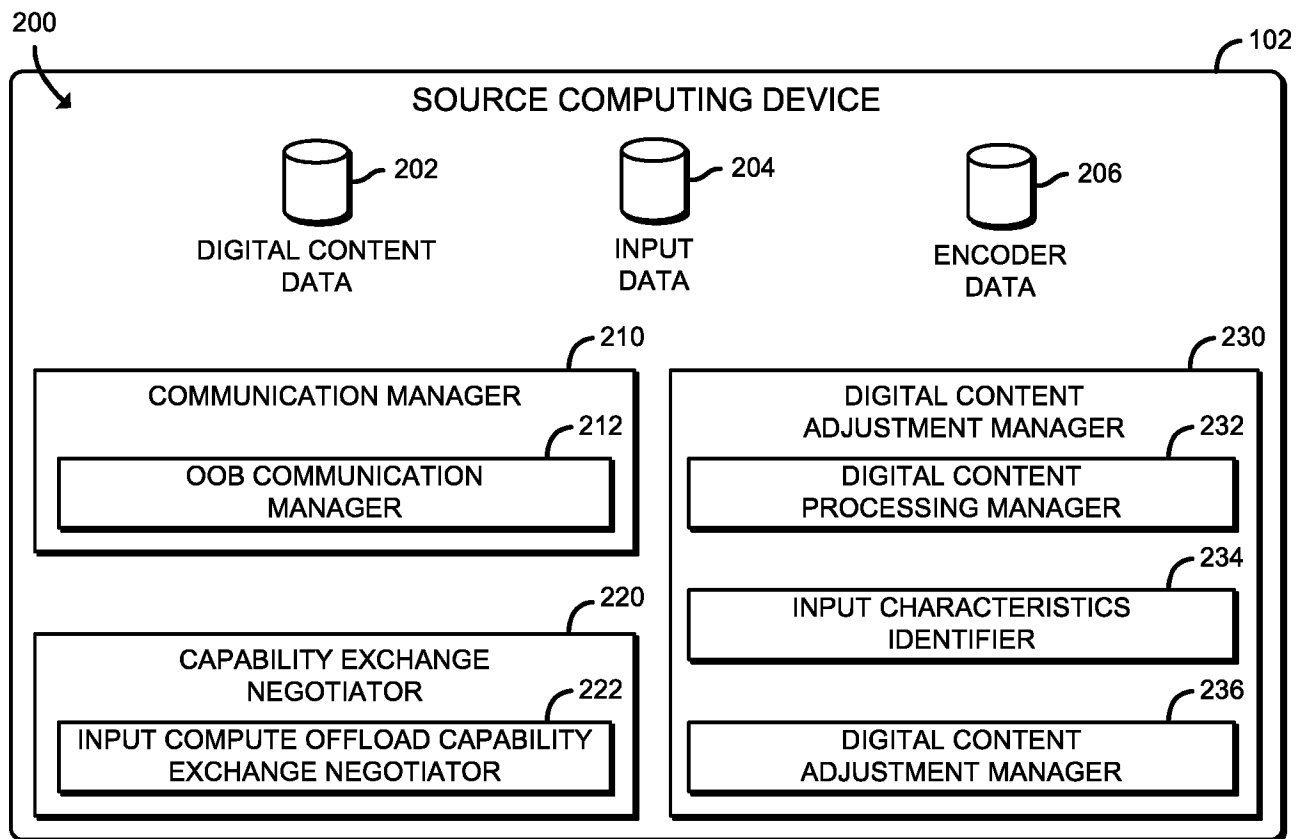


FIG. 2

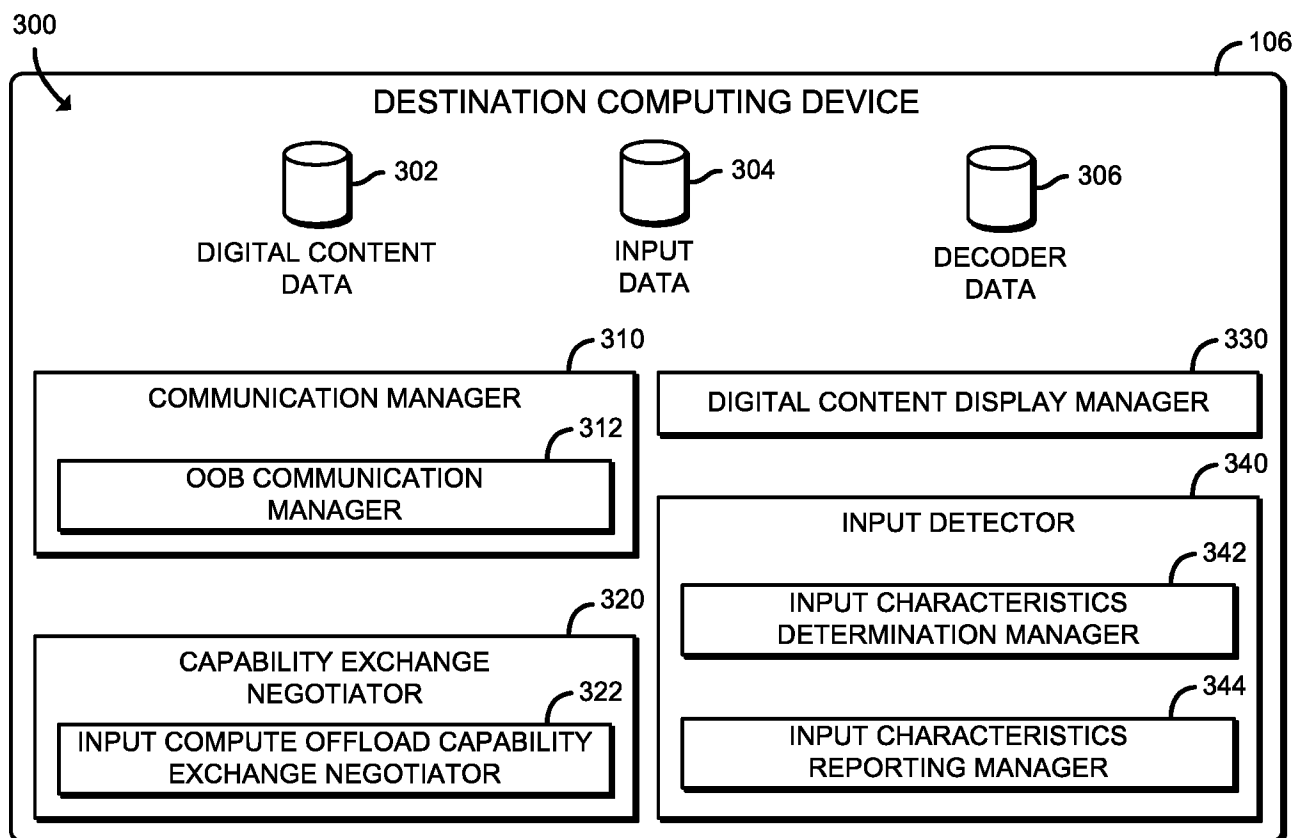


FIG. 3

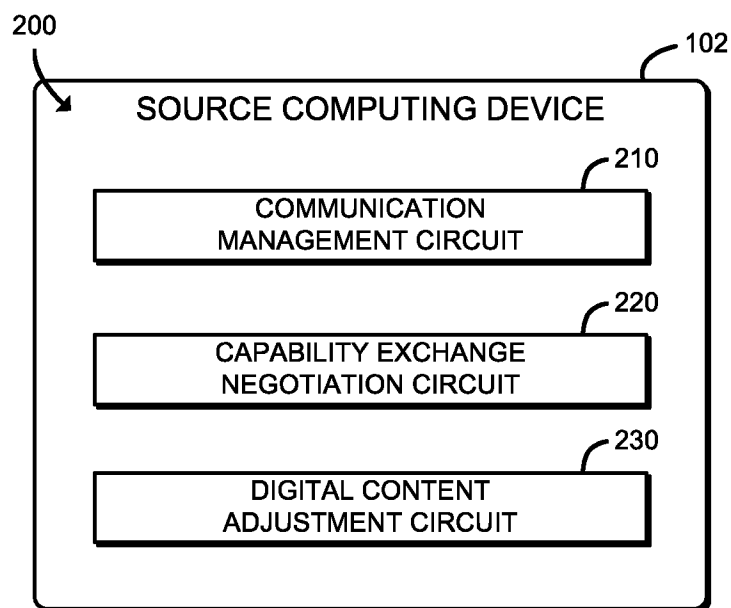


FIG. 4

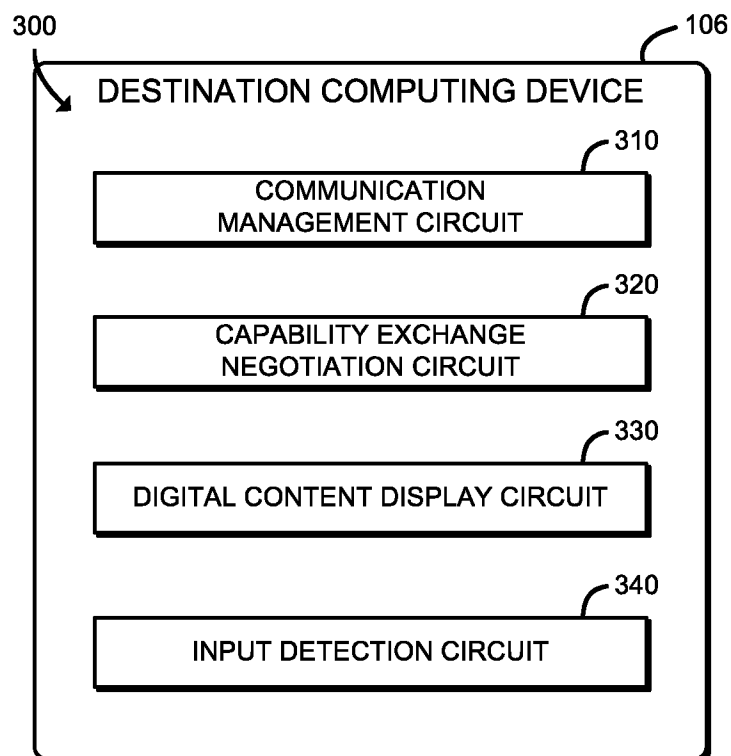


FIG. 5

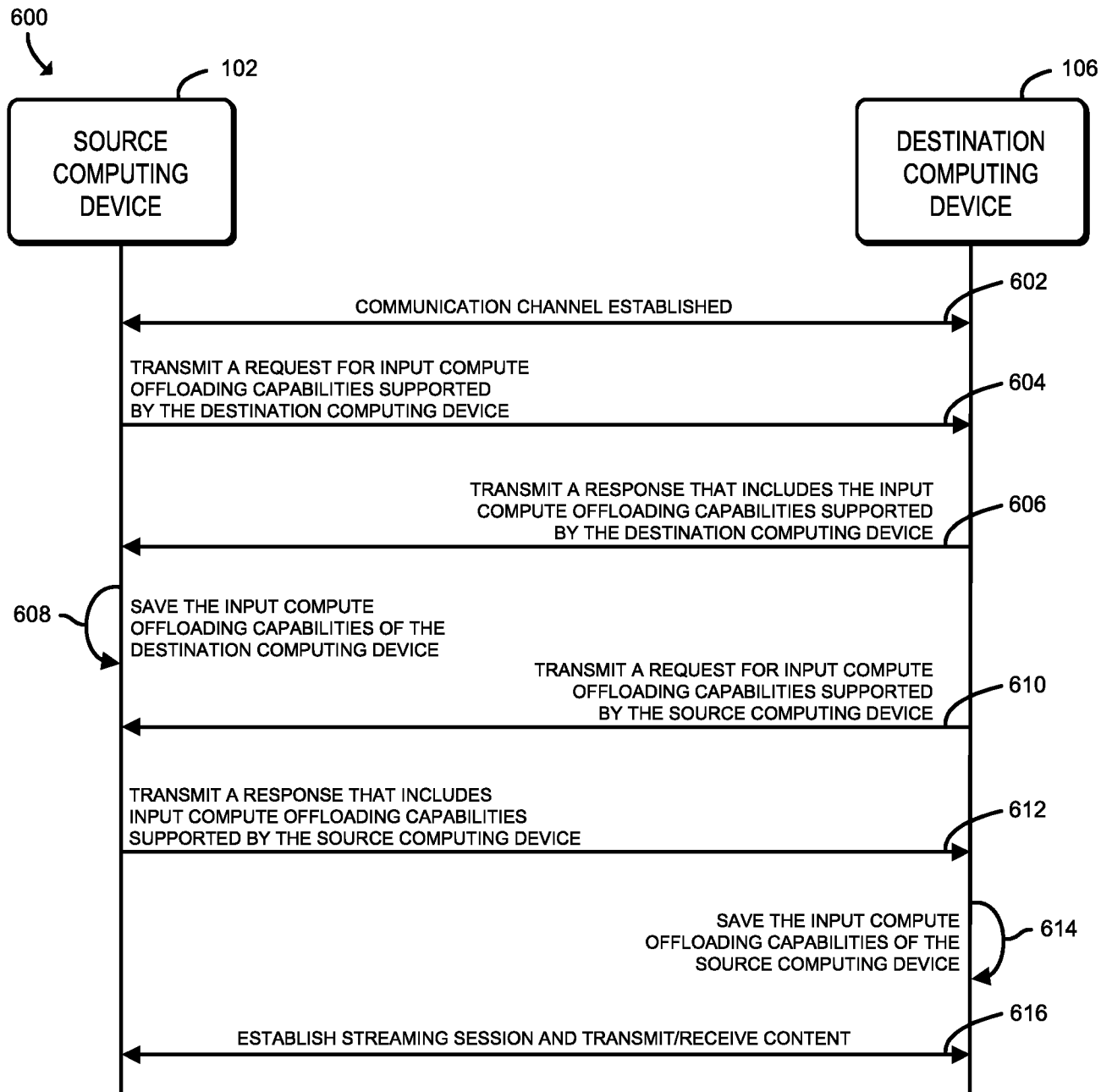


FIG. 6

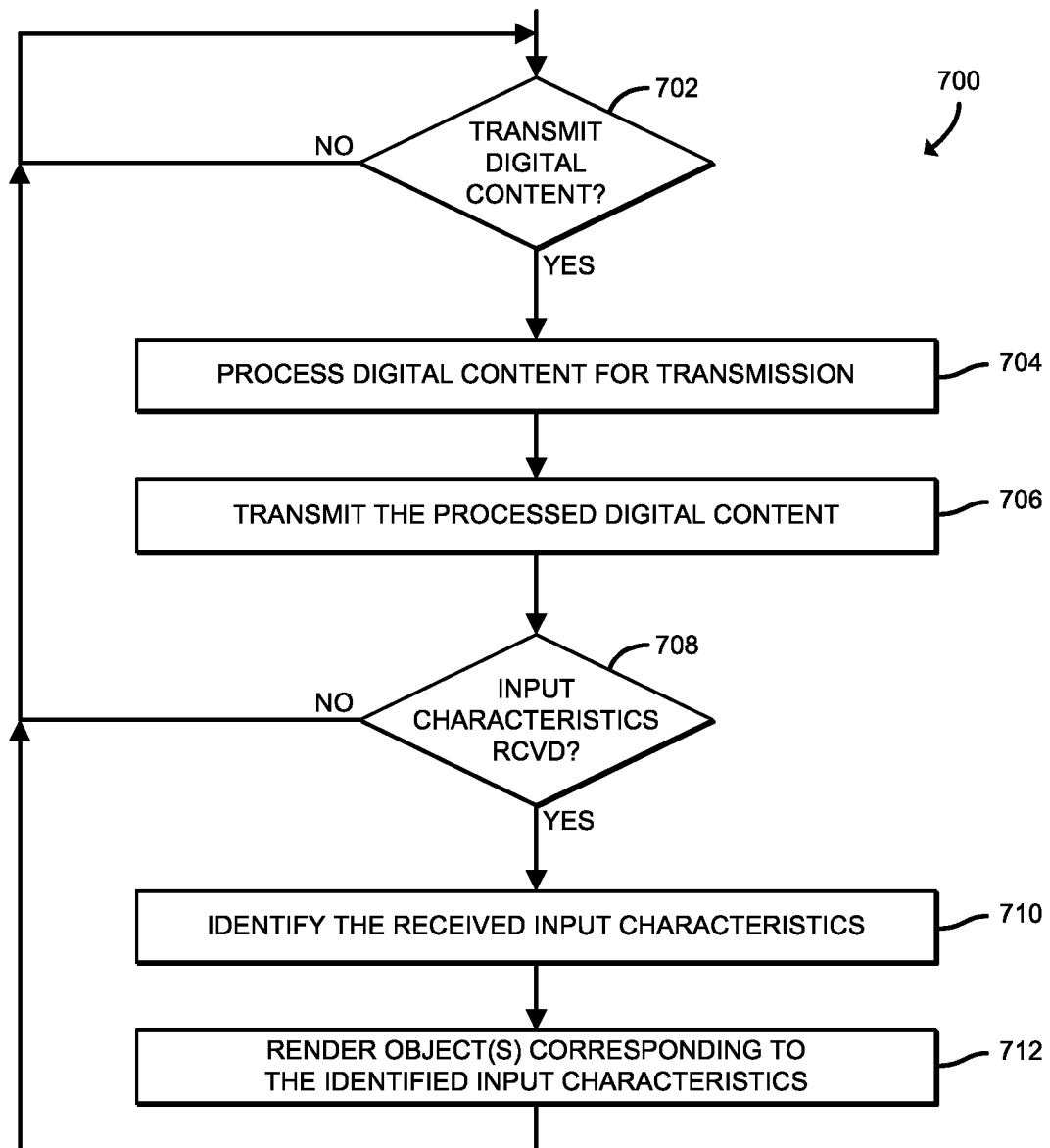


FIG. 7

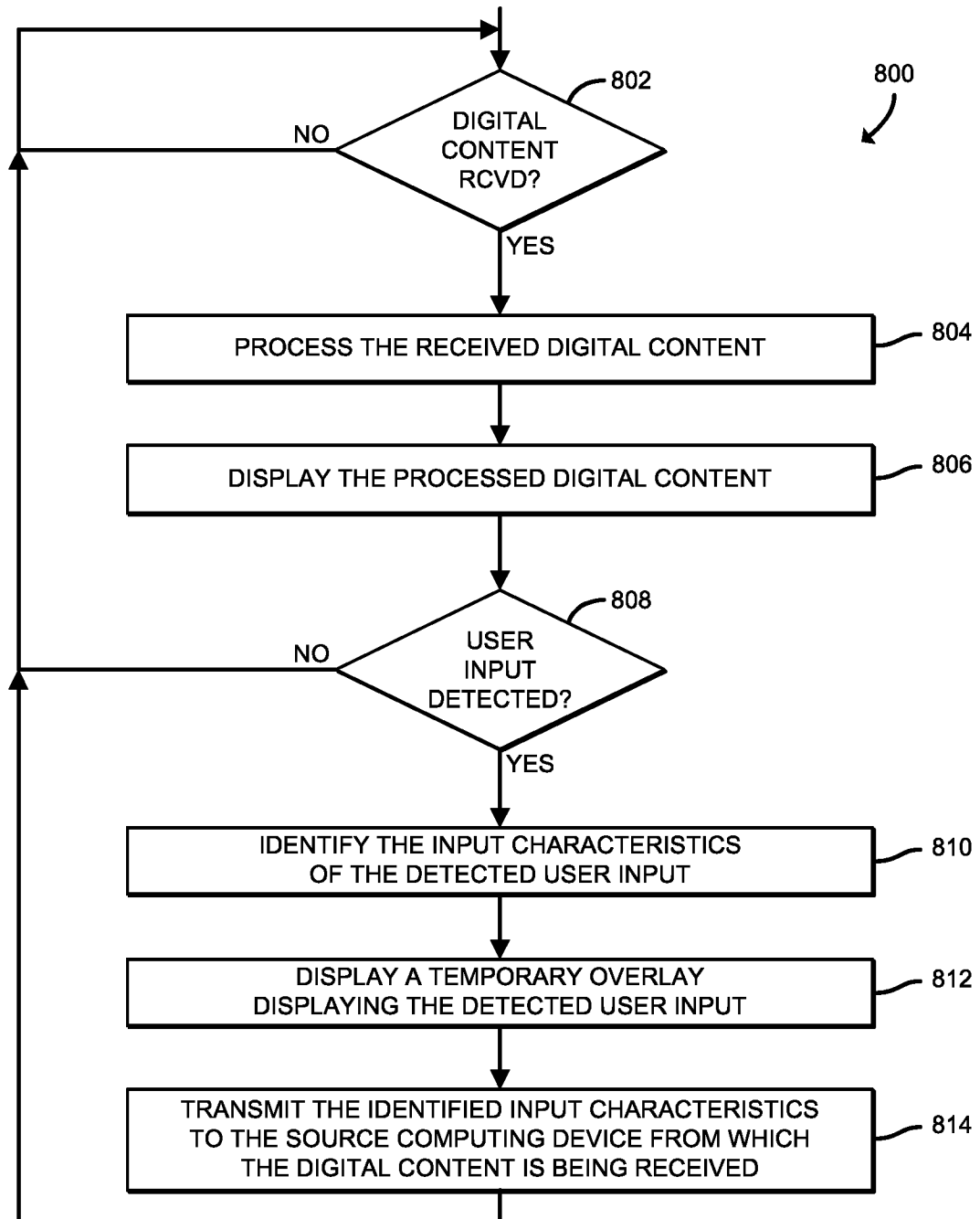


FIG. 8

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2017/026928**A. CLASSIFICATION OF SUBJECT MATTER****H04L 12/801(2013.01)i, H04L 29/08(2006.01)i**

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

B. FIELDS SEARCHEDMinimum documentation searched (classification system followed by classification symbols)
H04L 12/801; H04N 1/00; G02B 27/01; H04W 8/00; G06F 17/28; G06F 1/16; H04L 29/08Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility modelsElectronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: input compute offloading, source computing device, destination computing device, input characteristics, temporary overlay, capability exchange negotiator**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2013-0185052 A1 (MICROSOFT CORPORATION) 18 July 2013 See paragraphs [0016], [0022]-[0023], [0026], [0033], [0047]-[0050]; claims 5, 19-20; and figures 1-2, 5.	1-22
Y	US 2015-0130688 A1 (GOOGLE INC.) 14 May 2015 See title; paragraphs [0028], [0068]-[0069], [0087], [0107]; and figures 2, 4D.	1-22
Y	US 2014-0122558 A1 (NVIDIA CORPORATION) 01 May 2014 See paragraphs [0007], [0018]-[0019], [0029], [0034]; and figure 2.	5, 10, 15, 20
A	US 2015-0220110 A1 (USQUARE SOFT INC.) 06 August 2015 See paragraphs [0018], [0062]-[0063], [0065]; and figure 14.	1-22
A	US 2014-0139689 A1 (SONY ELECTRONICS INC. et al.) 22 May 2014 See paragraphs [0004], [0006], [0016]-[0017]; and figure 2.	1-22

☐ 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

29 June 2017 (29.06.2017)

Date of mailing of the international search report

24 July 2017 (24.07.2017)

Name and mailing address of the ISA/KR

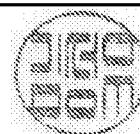
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

KIM, Seong Woo

Telephone No. +82-42-481-3348



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2017/026928

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2013-0185052 A1	18/07/2013	US 2008-0243473 A1 US 2013-0338997 A1 US 8515728 B2 US 8645121 B2 US 9298704 B2	02/10/2008 19/12/2013 20/08/2013 04/02/2014 29/03/2016
US 2015-0130688 A1	14/05/2015	CN 105745567 A KR 10-2016-0071417 A WO 2015-073412 A1	06/07/2016 21/06/2016 21/05/2015
US 2014-0122558 A1	01/05/2014	CN 103795704 A DE 10 2013 017638 A1 TW 201432471 A	14/05/2014 30/04/2014 16/08/2014
US 2015-0220110 A1	06/08/2015	CA 2841371 A1 CN 106030443 A EP 2902856 A2 EP 2902856 A3 JP 2017-511082 A KR 10-2016-0117443 A WO 2015-113152 A1	31/07/2015 12/10/2016 05/08/2015 21/10/2015 13/04/2017 10/10/2016 06/08/2015
US 2014-0139689 A1	22/05/2014	CN 101582963 A EP 2112815 A2 EP 2112815 A3 EP 2613515 A1 JP 2009-268092 A JP 2014-082773 A JP 5431775 B2 US 2009-0263012 A1 US 2012-0320225 A1 US 2013-0286229 A1 US 8310556 B2 US 8542285 B2 US 8730336 B2	18/11/2009 28/10/2009 05/05/2010 10/07/2013 12/11/2009 08/05/2014 05/03/2014 22/10/2009 20/12/2012 31/10/2013 13/11/2012 24/09/2013 20/05/2014