METHODS AND SYSTEMS FOR MAPPING POINTING DEVICE ON DEPTH MAP

Inventors: Andrey Valik, Miass (RU); Pavel Zaitsev, Miass (RU); Dmitry Morozov, Miass (RU); Alexander Argutin, Miass (RU)

Assignee: 3DIVI, Miass (RU)

Appl. No.: 13/541,684
Filed: Jul. 4, 2012

ABSTRACT

Disclosed are methods for determining and tracking a current location of a handheld pointing device, such as a remote control for an entertainment system, on a depth map generated by a gesture recognition control system. The methods disclosed herein enable identifying a user’s hand gesture, and generating corresponding motion data. Further, the handheld pointing device may send motion, such as acceleration or velocity, and/or orientation data such as pitch, roll, and yaw angles. The motion data of user’s hand gesture and motion data (orientation data) as received from the handheld pointing device are then compared, and if they correspond to each other, it is determined that the handheld pointing device is in active use by the user as it is held by a particular hand. Accordingly, a location of the handheld pointing device on the depth map can be determined.
HANDHELD POINTING DEVICE 220

MOTION SENSOR 510

ORIENTATION SENSOR 520

COMMUNICATION MODULE 530

FIG. 5
START

610 GENERATE DEPTH MAP

620 IDENTIFY USER ON DEPTH MAP

630 GENERATE VIRTUAL SKELETON OF USER

640 DETERMINE COORDINATES OF USER HANDS

650 DETERMINE MOTIONS OF USER HANDS

660 GENERATE MOTION DATA OF USER HANDS

670 ACQUIRE MOTION DATA OF HANDHELD ELECTRONIC DEVICE

680 COMPARE MOTION DATA OF HANDHELD ELECTRONIC DEVICE AND MOTION DATA OF USER HANDS

690 SELECTIVELY ASSIGN COORDINATES OF USER HAND TO HANDHELD POINTING DEVICE, IF MOTION DATA OF HANDHELD ELECTRONIC DEVICE CORRESPOND TO MOTION DATA OF ONE USER HAND

END

FIG. 6
METHODS AND SYSTEMS FOR MAPPING POINTING DEVICE ON DEPTH MAP

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is Continuation-in-Part of Russian Patent Application Serial No. 2011127116, filed on Jul. 4, 2011, which is incorporated herein by reference in its entirety for all purposes.

TECHNICAL FIELD

[0002] This disclosure relates generally to human-computer interfaces and, more particularly, to the technology for determining a location of a handheld pointing device, such as a remote controller for a game console, on a depth map generated by a gesture recognition control system.

BACKGROUND

[0003] The approaches described in this section could be pursued, but are not necessarily approaches that have previously been conceived or pursued. Therefore, unless otherwise indicated, it should not be assumed that any of the approaches described in this section qualify as prior art merely by virtue of their inclusion in this section.

[0004] Technologies associated with human-computer interaction have evolved over the last several decades. There are currently many various input devices and associated interfaces to enable computer users to control and provide data to their computers. Keyboards, pointing devices, joysticks, and touchscreens are just some examples of input devices that can be used to interact with various software products. One of the rapidly growing technologies in this field is the gesture recognition technology which enables the users to interact with the computer naturally, using body language, without any mechanical devices. In particular, the users can make inputs or generate commands using gestures or motions made by hands, arms, fingers, legs, and so forth. For example, using the concept of gesture recognition, it is possible to point a finger at the computer screen so that the pointer will move accordingly.

[0005] There currently exist various gesture recognition control systems (also known as motion sensing input systems) which, generally speaking, include a depth sensing camera, which captures scene images in real time, and a computing unit, which interprets captured scene images so as to generate various commands based on identification of user gestures. Typically, the gesture recognition control systems have very limited computation resources and also small resolution of the depth sensing camera so it is difficult to identify and track motions of relatively small objects such as handheld pointing devices. Various handheld pointing devices may play an important role for human-computer interaction, especially, for gaming software applications. The pointing devices may refer to controller wands or remote control devices enabling the users to generate specific commands by pressing dedicated buttons arranged thereon or by making predetermined gestures. Accordingly, the computer can be controlled via the gesture recognition technology (i.e., by processing data related to motion and location of the handheld pointing devices) and also receipt of specific commands originated by pressing dedicated buttons.

[0007] Typically, the gesture recognition control systems, when enabled, monitor and track all gestures performed by users with the help of handheld pointing devices. However, to enable the gesture recognition control systems to identify and track a motion of a relatively small handheld pointing device, a high resolution depth sensing camera and immoderate computational resources are used. Moreover, the present day gesture recognition control systems lack sufficient accuracy or generate unwanted latency when there are tracked gestures performed by relatively small handheld pointing devices. Alternatively, the handheld pointing devices may include specific auxiliary devices, such as a lighting sphere, to facilitate their identification and tracking. Either one of these approaches is disadvantageous and increases costs of the gesture recognition control systems. In view of the foregoing, there is still a need for improvements of gesture recognition control systems that will enhance interaction effectiveness and reduce required computational resources.

SUMMARY

[0008] This summary is provided to introduce a selection of concepts in a simplified form that are further described in the Detailed Description below. This summary is not intended to identify key or essential features of the claimed subject matter, nor is it intended to be used as an aid in determining the scope of the claimed subject matter.

[0009] The present disclosure refers to gesture recognition control systems configured to identify various user gestures and generate corresponding control commands. More specifically, the technology disclosed herein can determine and track a current location of a handheld pointing device based upon comparison of user gestures captured by a depth sensing camera and motion data of a handheld pointing device acquired by a communication module. The present technology allows determining a current position of a handheld pointing device on a depth map using typical computational resources and without a necessity to use dedicated auxiliary devices such as a lighting sphere.

[0010] The gesture recognition control system includes a depth sensing camera, which is used for generation of a depth map, and also a computing unit configured to process the depth map in real time to identify a user, user gestures, one or more user body parts, a user skeleton, motion data associated with user gestures, orientation data associated with user gestures, generate one or more commands associated with the identified gestures, and so forth. The gesture recognition control system further includes a communication module which may receive motion data of a handheld pointing device and optionally orientation data of the handheld pointing device. Once motion data (and optionally the orientation data) of the handheld pointing device and motion data associated with a user gesture, such as a gesture of a user's arm, correspond to each other, the gesture recognition control system assigns a current location (coordinates) of the user hand to the handheld pointing device so that its exact location is determined and can be further tracked.

[0011] The gesture recognition control system may be operatively coupled to or integrated with a computer, display, game console, and so forth. Accordingly, the determined and tracked location of the handheld pointing device may be used to control a display screen, game, or any other software application running on the computer.

[0012] Thus, the present disclosure discloses various methods for determining and tracking a current location of hand-
held pointing device in real time and also corresponding systems that can be used to implement these methods. Below is provided a simplified summary of one or more aspects regarding these methods in order to provide a basic understanding of such aspects as a prelude to the more detailed description that is presented later.

[0013] According to an aspect, there is provided a method for determining a position of a handheld pointing device. An example method may comprise: determining one or more motions of one or more user hands on a depth map, generating motion data associated with the one or more motions of the one or more user hands, acquiring motion data of a handheld pointing device, determining that the motion of the handheld pointing device is associated with the one or more motions of the one or more user hands, and determining a current position of the handheld pointing device on the depth map.

[0014] According to various embodiments, the method may further comprise generating a virtual skeleton of the user. The virtual skeleton may comprise at least one virtual limb of the user. The method may further comprise determining coordinates of the one or more user hands, wherein the coordinates are associated with the virtual skeleton, and generating motion data of the one or more user hands. The determination that the motion of the handheld pointing device is associated with the one or more motions of the one or more user hands may comprise comparing motion data of the one or more user hands and motion data of the handheld pointing device.

[0015] According to further embodiments, the method may further comprise determining which hand is holding the handheld pointing device. The method may further comprise selectively assigning the coordinates of the hand holding the handheld pointing device to the handheld pointing device. The method may further comprise determining an orientation of the handheld pointing device based upon the coordinates of various virtual skeleton joints related to the hand holding the handheld pointing device.

[0016] According to further embodiments, the method may further comprise generating a vector associated with the orientation of the handheld pointing device. The handheld pointing device may be selected from a group comprising: a cellular phone, a smart phone, a remote controller, a video game console, a handheld game console, a computer, and a tablet computer. The motion data associated with a motion of the handheld pointing device may comprise one or more of acceleration data, velocity data, and inertial data.

[0017] According to further embodiments, the method may further comprise acquiring orientation data of the handheld pointing device. The orientation data may be generated by one or more orientations sensors of the handheld pointing device. The orientation data may comprise one or more of the following: a pitch angle, a roll angle, and a yaw angle.

[0018] According to further embodiments, the method may further comprise determining that the handheld pointing device is in active use by the user. The handheld pointing device is in active use by the user when the handheld pointing device is held and moved by the user and when the user is identified on the depth map.

[0019] According to yet another embodiment, the method may further comprise identifying the user on the depth map. The method may further comprise tracking motions of the one or more user hands.

[0020] In further examples, the above methods steps are stored on a non-transitory machine-readable medium comprising instructions, which perform the steps when implemented by one or more processors. In yet further examples, subsystems or devices can be adapted to perform the recited steps. Other features, examples, and embodiments are described below.

BRIEF DESCRIPTION OF THE DRAWINGS

[0021] Embodiments are illustrated by way of example, and not by limitation in the figures of the accompanying drawings, in which like references indicate similar elements and in which:

[0022] FIG. 1 shows an example system environment for providing a real time human-computer interface.

[0023] FIG. 2 is a general illustration of scene environment for controlling an electronic device by recognition of gestures made by a user.

[0024] FIG. 3A shows a simplified view of an exemplary virtual skeleton associated with a user.

[0025] FIG. 3B shows a simplified view of an exemplary virtual skeleton associated with a user holding a handheld pointing device.

[0026] FIG. 4 shows an environment suitable for implementing methods for determining a position of a handheld pointing device.

[0027] FIG. 5 shows a simplified diagram of a handheld pointing device, according to an example embodiment.

[0028] FIG. 6 is a process flow diagram showing a method for determining a position of the handheld pointing device, according to an example embodiment.

[0029] FIG. 7 is a diagrammatic representation of an example machine in the form of a computer system within which a set of instructions for the machine to perform any one or more of the methodologies discussed herein is executed.

DETAILED DESCRIPTION

[0030] The following detailed description includes references to the accompanying drawings, which form a part of the detailed description. The drawings show illustrations in accordance with example embodiments. These example embodiments, which are also referred to herein as “examples,” are described in enough detail to enable those skilled in the art to practice the present subject matter. The embodiments can be combined, other embodiments can be utilized, or structural, logical, and electrical changes can be made without departing from the scope of what is claimed. The following detailed description is therefore not to be taken in a limiting sense, and the scope is defined by the appended claims and their equivalents. In this document, the terms “a” and “an” are used, as is common in patent documents, to include one or more than one. In this document, the term “or” is used to refer to a nonexclusive “or,” such that “A or B” includes “A but not B,” “B but not A,” and “A and B,” unless otherwise indicated.

[0031] The techniques of the embodiments disclosed herein may be implemented using a variety of technologies. For example, the methods described herein may be implemented in software executing on a computer system or in hardware utilizing either a combination of microprocessors or other specially designed application-specific integrated circuits (ASICs), programmable logic devices, or various combinations thereof. In particular, the methods described
The embodiments described herein relate to computer-implemented methods for determining and tracking a current location of a handheld pointing device.

In general, one or more depth sensing cameras (and, optionally, video cameras) can be used to generate a depth map of a physical scene. The depth map analysis and interpretation can be performed by a computing unit operated coupled to or embedding the depth sensing camera. Some examples of computing units may include a desktop computer, laptop computer, tablet computer, gaming console, audio system, video system, cellular phone, smart phone, personal digital assistant (PDA), set-top box (STB), television set, smart television system, or any other wired or wireless electronic device. The computing unit may include or be operatively coupled to a communication unit which may communicate with various handheld pointing devices and, in particular, receive motion data of handheld pointing devices.

The term “handheld pointing device,” as used herein, refers to an input device or any other suitable remote controlling device which can be used for making an input. Some examples of handheld pointing devices include a remote controller, cellular phone, smart phone, video game console, handheld game console, computer (e.g., a tablet computer), and so forth. Regardless of what type of handheld pointing device is used, it may include various motion detectors, such as acceleration sensors, gyroscopes, or other detectors configured to measure velocity, momentum, and acceleration such as pitch, roll, and yaw (in other words, acceleration for X, Y, and Z movement in Cartesian axes), and/or orientation sensors to generate orientation data including pitch angles, roll angles, and yaw angles. In operation, the handheld pointing device determines motion data, which includes velocities and/or acceleration levels, and transmits it to the computing unit over a wired or wireless network.

The computing unit, in turn, interprets the depth map such that it may identify the user, generate a corresponding virtual skeleton of the user, which skeleton includes multiple “joints” and “bones,” and determine that the user made a gesture using his hands or arms. The coordinates of every joint can be determined by the computing unit, and thus every user hand/arm motion can be tracked, and corresponding motion data can be generated, which may include a velocity, acceleration, orientation, and so forth.

Further, the computing unit compares motion data associated with the user’s hand/arm gesture and motion data (and optionally orientation data) associated with movement of the handheld pointing device. When both of these motion data coincide or correspond to each other, the computing unit determines that the handheld pointing device is held by a corresponding arm or hand of the user. Since coordinates of the user’s arm/hand are known and tracked, the same coordinates are then assigned to the handheld pointing device. Therefore, the handheld pointing device can be tied to the virtual skeleton of the user so that the current location of the handheld pointing device can be determined and further monitored. In other words, the handheld pointing device is mapped on the depth map.

Once the handheld pointing device is tied to the user, movements of the handheld pointing device may be further tracked in real time to identify particular user gestures causing the computing unit to generate corresponding control commands. This approach can be used in various gaming and simulation/teaching software without a necessity to use inordinate computational software, high resolution depth sensing cameras, or auxiliary devices (e.g., a fighting sphere) attached to the handheld pointing device to facilitate its identification on the depth map. The technology described herein provides an easy and effective method for locating the handheld pointing device on the scene and tracking its motions.

Provided below is a detailed description of various embodiments related to methods and systems for determining a position of a handheld pointing device.

With reference now to the drawings, FIG. 1 shows an example system environment 100 for providing a real-time human-computer interface. The system environment 100 includes a gesture recognition control system 110, a display device 120, and an entertainment system 130.

The gesture recognition control system 110 is configured to capture various user gestures and user inputs, interpret them, and generate corresponding control commands, which are further transmitted to the entertainment system 130. Once the entertainment system 130 receives commands generated by the gesture recognition control system 110, the entertainment system performs certain actions depending on which software application is running. For example, the user may control a pointer on the display screen by making certain gestures.

The entertainment system 130 may refer to any electronic device such as a computer (e.g., a laptop computer, desktop computer, tablet computer, workstation, server), game console, television (TV) set, TV adapter, STB, smart television system, audio system, video system, cellular phone, smart phone, PDA, and so forth. Although the figure shows that the gesture recognition control system 110 and the entertainment system 130 are separate and stand-alone devices, in some alternative embodiments, these systems can be integrated within a single device.

FIG. 2 is a general illustration of a scene 200 suitable for controlling an electronic device by recognition of gestures made by a user. In particular, this figure shows a user 210 interacting with the gesture recognition control system 110 with the help of a handheld pointing device 220.

The gesture recognition control system 110 may include a depth sensing camera, a computing unit, and a communication unit, which can be stand-alone devices or embedded within a single housing (as shown). Generally speaking, the user and a corresponding environment, such as a living room, are located, at least in part, within the field of view of the depth sensing camera.

More specifically, the gesture recognition control system 110 may be configured to capture a depth map of the scene in real time and further process the depth map to identify the user, determine one or more user gestures, determine one or more user body parts, and generate corresponding control commands. The gesture recognition control system 110 may also determine specific motion data associated with user gestures, wherein the motion data may include coordinates of the user’s hands or arms, and velocity and acceleration of the user’s hands/arms. For this purpose, the gesture recognition control system 110 may generate a virtual skeleton of the user as shown in FIG. 3 and described below in greater detail.

The handheld pointing device 220 may refer to a controller wand, remote control device (e.g., a gaming console remote controller), smart phone, cellular phone, PDA,
tablet computer, or any other electronic device enabling the user 210 to generate specific commands by pressing dedicated buttons arranged thereon. The handheld pointing device 220 is configured to determine its velocity, acceleration and/or orientation within the space with the help of embedded acceleration sensors, gyroscopes, or other motion sensors and/or orientation sensors. The velocity, acceleration and/or orientation data can be transmitted to the gesture recognition control system 110 over a wireless or wire network. Accordingly, a communication module, which is configured to receive motion data (and optionally orientation data) associated with movements of handheld pointing device 220, may be embedded in the gesture recognition control system 110.

[0046] The gesture recognition control system 110 is also configured to determine the location of handheld pointing device 220 on the depth map by matching motion data associated with the gestures of one or more user’s arms captured by the depth sensing camera and motion data (and optionally the orientation data) associated with movements of handheld pointing device 220 as received by the communication module. When the motions match each other, the gesture recognition control system 110 acknowledges that the handheld pointing device 220 is held in a particular hand of the user and then assigns coordinates of the user’s hand to the handheld pointing device 220. In various embodiments, this technology can be used for determining that the handheld pointing device 220 is in “active use,” which means that the handheld pointing device 220 is held by the user 210 who is located in the sensitive area of the depth sensing camera.

[0047] FIG. 3A shows a simplified view of an exemplary virtual skeleton 300 as can be generated by the gesture recognition control system 110 based upon the depth map. As shown in the figure, the virtual skeleton 300 comprises a plurality of “bones” and “joints” 310 interconnecting the bones. The bones and joints, in combination, represent the user 210 in real time so that every motion of the user’s limbs is represented by corresponding motions of the bones and joints.

[0048] According to various embodiments, each of the joints 310 may be associated with certain coordinates in a three-dimensional (3D) space defining its exact location. Hence, any motion of the user’s limbs, such as an arm, may be interpreted by a plurality of coordinates or coordinate vectors related to the corresponding joint(s) 310. By tracking user motions via the virtual skeleton model, motion data can be generated for every limb movement. This motion data may include exact coordinates per period of time, velocity, direction, acceleration, orientation, and so forth.

[0049] FIG. 3B shows a simplified view of exemplary virtual skeleton 300 associated with the user 210 holding the handheld pointing device 220. In particular, when the gesture recognition control system 110 determines that the user 210 holds and the handheld pointing device 220 and then determines the location (coordinates) of the handheld pointing device 220, a corresponding mark or label can be generated on the virtual skeleton 300.

[0050] According to various embodiments, the gesture recognition control system 110 can determine an orientation of the handheld pointing device 220 by analyzing the virtual skeleton 300 and/or by acquiring orientation data from the handheld pointing device 220. In this case, the orientation of handheld pointing device 220 may be represented as a vector 320 as shown in FIG. 3B.

[0051] FIG. 4 shows an environment 400 suitable for implementing methods for determining a position of a handheld pointing device 220. As shown in this figure, there is provided the gesture recognition control system 110, which may comprise at least one depth sensing camera 410 configured to capture a depth map. The term “depth map,” as used herein, refers to an image or image channel that contains information relating to the distance of the surfaces of scene objects from a depth sensing camera. In various embodiments, the depth sensing camera 410 may include an infrared (IR) projector to generate modulated light, and also an IR camera to capture 3D images. In yet more example embodiments, the gesture recognition control system 110 may optionally comprise a color video camera 420 to capture a series of 2D images in addition to 3D imagery created by the depth sensing camera 410. The series of 2D images captured by the color video camera 420 may be used to facilitate identification of the user on the depth map and/or various gestures of the user. It should be also noted that the depth sensing camera 410 and the color video camera 420 can be either stand alone devices or be encased within a single housing.

[0052] Furthermore, the gesture recognition control system 110 may also comprise a computing unit 430 for processing depth data and generating control commands for one or more electronic devices 460 (e.g., the entertainment system 130). The computing unit 430 is also configured to implement steps of methods for determining a position of the handheld pointing device 220 as described herein.

[0053] The gesture recognition control system 110 also includes a communication module 440 configured to communicate with the handheld pointing device 220 and one or more electronic devices 460. More specifically, the communication module 440 is configured to receive motion data and orientation data from the handheld pointing device 220 and transmit control commands to one or more electronic devices 460.

[0054] The gesture recognition control system 110 may also include a bus 450 interconnecting the depth sensing camera 410, color video camera 420, computing unit 430, and communication module 440.

[0055] The aforementioned one or more electronic devices 460 can refer, in general, to any electronic device configured to trigger one or more predefined actions upon receipt of a certain control command. Some examples of electronic devices 460 include, but are not limited to, computers (e.g., laptop computers, tablet computers), displays, audio systems, video systems, gaming consoles, entertainment systems, lighting devices, cellular phones, smart phones, TV’s, and so forth.

[0056] The communication between the communication module 440 and the handheld pointing device 220 and/or one or more electronic devices 460 can be performed via a network (not shown). The network can be a wireless or wired network, or a combination thereof. For example, the network may include the Internet, local intranet, PAN (Personal Area Network), LAN (Local Area Network), WAN (Wide Area Network), MAN (Metropolitan Area Network), virtual private network (VPN), storage area network (SAN), frame relay connection, Advanced Intelligent Network (AIN) connection, synchronous optical network (SONET) connection, digital T1, T3, E1 or E3 line, Digital Data Service (DDS) connection, DSL (Digital Subscriber Line) connection, Ethernet connection, ISDN (Integrated Services Digital Net-
work) line, dial-up port such as a V.90, V.34 or V.34bis analog modem connection, cable modem, ATM (Asynchronous Transfer Mode) connection, or an FDDI (Fiber Distributed Data Interface) or CDDI (Copper Distributed Data Interface) connection. Furthermore, communications may also include links to any of a variety of wireless networks including WAP (Wireless Application Protocol), GPRS (General Packet Radio Service), GSM (Global System for Mobile Communication), CDMA (Code Division Multiple Access) or TDMA (Time Division Multiple Access), cellular phone networks, Global Positioning System (GPS), CDPD (Cellular Digital Packet Data), RIM (Research in Motion, Limited) duplex paging network, Bluetooth radio, or an IEEE 802.11-based radio frequency network. The network can further include or interface with any one or more of the following: RS-232 serial connection, IEEE-1394 (Firewire) connection, Fiber Channel connection, IrDA (infrared) port, SCSI (Small Computer Systems Interface) connection, USB (Universal Serial Bus) connection, or other wired or wireless, digital or analog interface or connection, such as a daisy chaining system.

[0057] FIG. 5 shows a simplified diagram of the handheld pointing device 220, according to an example embodiment. As shown in the figure, the handheld pointing device 220 comprises one or more motion sensors 510, one or more orientation sensors 520, and a communication module 530. In various alternative embodiments, the handheld pointing device 220 may include additional modules (not shown), such as an input module, a computing module, a display, or any other modules, depending on the type of the handheld pointing device 220.

[0058] The motion sensors 510 and orientation sensors 520 may include gyros, accelerometers, sensors, and so forth. In general, the motion sensors 510 are configured to determine motion data which may include a velocity, momentum, and acceleration such as pitch, roll, and yaw (in other words, acceleration for X, Y, and Z movement in Cartesian axes) of the handheld pointing device 220. The orientation sensors 520 may determine a relative orientation of the handheld pointing device 220. In an example, the orientation sensors 520 are configured to generate orientation data including one or more of the following: pitch angle, roll angle, and yaw angle related to the handheld pointing device 220. In operation, motion data and optionally orientation data are then transmitted to the gesture recognition control system 110 with the help of communication module 530. The motion data and orientation data can be transmitted via the network as described above.

[0059] FIG. 6 is a process flow diagram showing a method 600 for determining a position of the handheld pointing device 220, according to an example embodiment. The method 600 may be performed by processing logic that may comprise hardware (e.g., dedicated logic, programmable logic, and microcode), software (such as software run on a general-purpose computer system or a dedicated machine), or a combination of both. In one example embodiment, the processing logic resides at the gesture recognition control system 110.

[0060] The method 600 can be performed by the units/devices discussed above with reference to FIG. 4. Each of these units or devices can comprise processing logic. It will be appreciated by one of ordinary skill in the art that examples of the foregoing units/devices may be virtual, and instructions said to be executed by a unit/device may in fact be retrieved and executed by a processor. The foregoing units/devices may also include memory cards, servers, and/or computer discs. Although various modules may be configured to perform some or all of the various steps described herein, fewer or more units may be provided and still fall within the scope of example embodiments.

[0061] As shown in FIG. 6, the method 600 may commence at operation 610, with the depth sensing camera 410 generating a depth map by capturing a plurality of depth values of the scene in real time.

[0062] At operation 620, the depth map can be analyzed by the computing unit 430 to identify the user 210 on the depth map. At operation 630, the computing unit 430 segments the depth data of the user 210 so as to generate a virtual skeleton of the user 210.

[0063] At operation 640, the computing unit 430 determines coordinates of at least one user’s hand (user’s arm or user’s limb). The coordinates of the at least one user’s hand can be associated with the virtual skeleton as discussed above.

[0064] At operation 650, the computing unit 430 determines a motion of the at least one user’s hand by processing a plurality of depth maps. At operation 660, the computing unit 430 generates motion data of the at least one user’s hand. At operation 670, the computing unit 430 acquires motion data and optionally orientation data of the handheld electronic device 220 via the communication module 440.

[0065] At operation 680, the computing unit 430 compares the motion data (and optionally orientation data) of handheld electronic device 220 as acquired at operation 670 and the motion data of the at least one user’s hand as generated at operation 660. If the motion data of handheld electronic device 220 correspond (or match or are relatively similar) to the motion data of the user’s hand, the computing unit 430 selectively assigns the coordinates of the user’s hand to the handheld pointing device 220 at operation 690. Thus, the location of handheld pointing device 220 is determined on the depth map. Further, the location of handheld pointing device 220 can be tracked in real time so that various gestures can be interpreted for generation of corresponding control commands for one or more electronic devices 460.

[0066] In various embodiments, the described technology can be used for determining that the handheld pointing device 220 is in active use by the user 210. As mentioned earlier, the term “active use” means that the user 210 is identified on the depth map (see operation 620) or, in other words, is located within the viewing area of depth sensing camera 410 when the handheld pointing device 220 is moved.

[0067] In addition, the method 600 may further include operations (not shown) when the computing unit 430 generates a vector defining the current orientation of the handheld pointing device 220. The orientation of handheld pointing device 220 may be represented as the vector 320 (see FIG. 3b). The computing unit 430 generates the vector 320 by processing the orientation data of the handheld electronic device 220 as acquired at operation 670 by transforming the orientation data tied to an axis system of the handheld electronic device 220 to orientation data tied to an axis system of the gesture recognition control system 110. In other words, the vector coordinates are calculated for the axis system associated with the gesture recognition control system 110 based upon the vector coordinates in the axis system of the handheld electronic device 220.

[0068] FIG. 7 shows a diagrammatic representation of a computing device for a machine in the example electronic form of a computer system 700, within which a set of instruc-
tions for causing the machine to perform any one or more of the methodologies discussed herein can be executed. In example embodiments, the machine operates as a standalone device, or can be connected (e.g., networked) to other machines. In a networked deployment, the machine can operate in the capacity of a server, a client machine in a server-client network environment, or as a peer machine in a peer-to-peer (or distributed) network environment. The machine can be a personal computer (PC), tablet PC, STB, PDA, cellular telephone, portable music player (e.g., a portable hard drive audio device, such as a Moving Picture Experts Group Audio Layer 3 (MP3) player), web appliance, network router, switch, bridge, or any machine capable of executing a set of instructions (sequential or otherwise) that specify actions to be taken by that machine. Further, while only a single machine is illustrated, the term “machine” shall also be taken to include any collection of machines that separately or jointly execute a set (or multiple sets) of instructions to perform any one or more of the methodologies discussed herein.

[0069] The example computer system 700 includes a processor or multiple processors 702 (e.g., a central processing unit (CPU), graphics processing unit (GPU), or both), main memory 704 and static memory 706, which communicate with each other via a bus 708. The computer system 700 can further include a video display unit 710 (e.g., a liquid crystal display (LCD) or cathode ray tube (CRT)). The computer system 700 also includes at least one input device 712, such as an alphanumeric input device (e.g., a keyboard), pointer control device (e.g., a mouse), microphone, digital camera, video camera, and so forth. The computer system 700 also includes a disk drive unit 714, signal generation device 716 (e.g., a speaker), and network interface device 718.

[0070] The disk drive unit 714 includes a computer-readable medium 720 that stores one or more sets of instructions and data structures (e.g., instructions 722) embodying or utilized by any one or more of the methodologies or functions described herein. The instructions 722 can also reside, completely or at least partially, within the main memory 704 and/or within the processors 702 during execution by the computer system 700. The main memory 704 and the processors 702 also constitute machine-readable media.

[0071] The instructions 722 can further be transmitted or received over the network 724 via the network interface device 718 utilizing any one of a number of well-known transfer protocols (e.g., Hyper Text Transfer Protocol (HTTP), CAN, Serial, and Modbus).

[0072] While the computer-readable medium 720 is shown in an example embodiment to be a single medium, the term “computer-readable medium” should be taken to include a single medium or multiple media (e.g., a centralized or distributed database, and/or associated caches and servers) that store the one or more sets of instructions. The term “computer-readable medium” shall also be taken to include any medium that is capable of storing, encoding, or carrying a set of instructions for execution by the machine, and that causes the machine to perform any one or more of the methodologies of the present application, or that is capable of storing, encoding, or carrying data structures utilized by or associated with such a set of instructions. The term “computer-readable medium” shall accordingly be taken to include, but not be limited to, solid-state memories, optical and magnetic media. Such media may also include, without limitation, hard disks, floppy disks, flash memory cards, digital video disks, random access memory (RAM), read only memory (ROM), and the like.

[0073] The example embodiments described herein may be implemented in an operating environment comprising computer-executable instructions (e.g., software) installed on a computer, in hardware, or in a combination of software and hardware. The computer-executable instructions may be written in a computer programming language or may be embodied in firmware logic. If written in a programming language conforming to a recognized standard, such instructions may be executed on a variety of hardware platforms and for interfaces associated with a variety of operating systems. Although not limited thereto, computer software programs for implementing the present method may be written in any number of suitable programming languages such as, for example, C, C++, Ch, Cobol, Eiffel, Haskell, Visual Basic, Java, JavaScript, Python, or other compilers, assemblers, interpreters, or other computer languages or platforms.

[0074] Thus, methods and systems for determining a position of a handheld pointing device have been described. Although embodiments have been described with reference to specific example embodiments, it will be evident that various modifications and changes can be made to these example embodiments without departing from the broader spirit and scope of the present application. Accordingly, the specification and drawings are to be regarded in an illustrative rather than a restrictive sense.

What is claimed is:

1. A computer-implemented method for determining a position of a handheld pointing device, the method comprising:
   - determining one or more motions of one or more user hands on a depth map;
   - generating motion data associated with the one or more motions of the one or more user hands;
   - acquiring motion data of a handheld pointing device;
   - determining that the motion of the handheld pointing device is associated with the one or more motions of the one or more user hands;
   - and determining a current position of the handheld pointing device on the depth map.

2. The method of claim 1, further comprising generating the depth map by capturing a series of images.

3. The method of claim 1, further comprising generating a virtual skeleton of the user, the virtual skeleton comprising at least one virtual limb of the user.

4. The method of claim 3, further comprising:
   - determining coordinates of the one or more user hands, the coordinates to be associated with the virtual skeleton;
   - and generating motion data of the one or more user hands.

5. The method of claim 4, wherein determining that the motion of the handheld pointing device is associated with the one or more motions of the one or more user hands comprises comparing motion data of the one or more user hands and motion data of the handheld pointing device.

6. The method of claim 5, further comprising determining which hand is holding the pointing device.

7. The method of claim 6, further comprising selectively assigning the coordinates of the hand holding the handheld pointing device to the handheld pointing device.

8. The method of claim 7, further comprising determining an orientation of the handheld pointing device based upon the
coordinates of various virtual skeleton joints related to the hand holding the handheld pointing device.

9. The method of claim 8, further comprising generating a vector associated with the orientation of the handheld pointing device.

10. The method of claim 1, further comprising acquiring orientation data of the handheld pointing device, wherein the orientation data comprises one or more of the following: a pitch angle, a roll angle, and a yaw angle.

11. The method of claim 1, wherein the motion data associated with a motion of the handheld pointing device comprises one or more of acceleration data, velocity data, and inertial data.

12. The method of claim 1, further comprising determining that the handheld pointing device is in active use by the user.

13. The method of claim 12, wherein the handheld pointing device is in active use by the user when the handheld pointing device is held and moved by the user and when the user is identified on the depth map.

14. The method of claim 1, further comprising identifying the user on the depth map.

15. The method of claim 1, further comprising tracking motions of the user hand.

16. A system for determining a position of a handheld pointing device, the system comprising:
   a depth sensing device configured to generate a depth map;
   a communication module configured to acquire motion data of a handheld pointing device; and
   a computing unit communicatively coupled to the depth sensing device, the computing unit configured to:
   determine one or more motions of one or more user hands on a depth map;
   generate motion data associated with the one or more motions of the one or more user hands;
   determine that the motion of the handheld pointing device is associated with the one or more motions of the one or more user hands; and
   determine a current position of the handheld pointing device on the depth map.

17. The system of claim 16, further comprising a video camera communicatively coupled to the computing unit, the video camera being configured to facilitate generation of the depth map.

18. The system of claim 16, wherein the computing unit is further configured to generate a virtual skeleton of the user, the virtual skeleton comprising at least one virtual hand of the user.

19. The system of claim 16, wherein the computing unit is further configured to:
   determine coordinates of the one or more user hands, the coordinates to be associated with the virtual skeleton;
   generate motion data of the one or more user hands; and
   compare motion data of the one or more user hands and motion data of the handheld pointing device.

20. A processor-readable nontransitory medium having instructions stored thereon, which when executed by one or more processors, cause the one or more processors to:
   determine one or more motions of one or more user hands on a depth map;
   generate motion data associated with the one or more motions of the one or more user hands;
   acquire motion data of a handheld pointing device;
   determine that the motion of the handheld pointing device is associated with the one or more motions of the one or more user hands; and
   determine a current position of the handheld pointing device on the depth map.