



(11) **EP 4 261 793 A1**

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
18.10.2023 Bulletin 2023/42

(21) Application number: **23167060.5**

(22) Date of filing: **06.04.2023**

(51) International Patent Classification (IPC):
G06V 20/58 ^(2022.01) **G06V 20/10** ^(2022.01)
G06F 18/25 ^(2023.01) **G06V 10/764** ^(2022.01)
G06V 10/82 ^(2022.01)

(52) Cooperative Patent Classification (CPC):
G06V 20/58; G06F 18/251; G06V 10/764;
G06V 10/82; G06V 20/10

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB
GR HR HU IE IS IT LI LT LU LV MC ME MK MT NL
NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA
Designated Validation States:
KH MA MD TN

(30) Priority: **11.04.2022 US 202217718192**

(71) Applicant: **Vorwerk & Co. Interholding GmbH**
42270 Wuppertal (DE)

(72) Inventors:
• **ORZECZOWSKI, Pawel**
San Jose, 95134 (US)
• **WILSON, Mark**
San Jose, 95134 (US)
• **DOMINGO, Ron**
San Jose, 95134 (US)
• **BLANKLEY, Stephanie**
San Jose, 95134 (US)

(74) Representative: **Dr. Solf & Zapf**
Patent- und Rechtsanwalts PartG mbB
Schlossbleiche 20
42103 Wuppertal (DE)

(54) **OBSTACLE AVOIDANCE USING FUSED DEPTH AND INTENSITY FOR NNT TRAINING**

(57) Embodiments provide a method for obstacle avoidance in a mobile robot. Images of objects are captured with mobile robot image sensors with a floor level perspective. Data corresponding to the images is used to train a neural network. The identified objects are classified by indicating whether the objects are a potential

hazard. In one embodiment, an image sensor on a robot provides intensity and depth data for each of a plurality of pixels of the images. The intensity and depth data are fused to produce fused data. The fused data is then used to train the neural network.

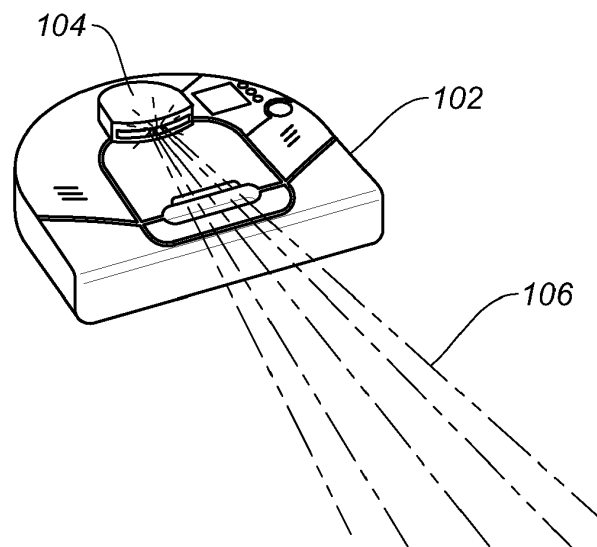


FIG. 1

EP 4 261 793 A1

Description

BACKGROUND OF THE INVENTION

[0001] The present invention relates to robots which collect images, and in particular to low profile cleaning robots with image detection capabilities.

[0002] Image detection and object classification has been used in many fields. There have been proposals to provide image recognition in household robots, such as cleaning robots. For example, US Pub. 20150289743 describes image recognition to determine a kind of dirt (e.g., hairs, spilled food) and select the appropriate cleaning capability. US Pub. 20160167226 describes loading training data into the memory of a cleaning robot for use in machine learning for object recognition and classification, and also describes machine learning alternately implemented on remote servers over the Internet. Machine learning to detect objects typically involves training a neural network using libraries of object images available online.

[0003] In addition to object detection, it is desirable to avoid obstacles by a cleaning robot. Neato Robotics uses infrared scanning lasers to detect the distance to obstacles. The distance to an obstacle can be determined using triangulation with a returned beam, using Time of Flight (TOF), or other methods. The laser can be on a rotating mount to detect obstacles, and map an environment, in all directions. Alternately, a non-moving laser emitter and detector can be used to give a 360 degree view, such as the PMD Technologies 3D sensor for TOF. In addition to image intensity, depth information can be provided by the PMD sensor, or by stereo cameras.

BRIEF SUMMARY OF THE INVENTION

[0004] Embodiments provide a method for obstacle avoidance in a mobile robot. Images of objects are captured with image sensors in mobile robots. The image sensors have a perspective within one foot of a surface upon which the mobile robots travel. Data corresponding to the images is transmitted to a remote neural network. The remote neural network is trained using machine learning by identifying and adding tags to the objects. The identified objects are classified by indicating whether the objects are a potential hazard. A library of identified and classified objects is created. A new image from an operating robot is compared to the library of identified and classified objects. A match of the new image to the library of identified and classified objects is identified. A hazard classification is provided to the operating robot. The operation of the operating robot is altered if the hazard classification is received.

[0005] Embodiments provide improved accuracy of obstacle detection by a robot, such as a cleaning robot. Object recognition is used to determine whether an object is an obstacle that should be avoided, or is something that can be run over and picked up by a cleaning robot,

such as a dust ball or spider web. Cleaning robots are typically short, and have a unique, floor-level viewpoint of objects, such as seeing chairs and legs from the bottom, not the top. This can cause misidentification when using a neural network trained by online images from other perspectives. Thus, embodiments use images captured by cleaning robots to train a neural network for machine learning. The objects in these images can be identified by humans and/or comparison to images in online libraries with different viewpoints.

[0006] In one embodiment, an image sensor on a robot provides intensity and depth data for each of a plurality of pixels of the images. The intensity and depth data are presented together for each pixel, providing "fused" data. The fused data is then used to train a neural network. This provides more differentiation and measurement richness for the neural network.

[0007] In one embodiment, the neural network is a multi-layer neural network. A high-fidelity model is trained on simultaneous depth and intensity images collected by a PMD time-of-flight camera. The fused depth and intensity data is used to train the multi-layer neural network.

[0008] In one embodiment of the method, the fusing comprises providing the intensity and depth data for each pixel in an image. Preferably, intensity data and depth data for each pixel in an image captured by a robot is provided by a PMD sensor. The PMD sensor is for example mounted near the top of a cleaning robot in one embodiment, in particular less than 304,8 mm (a foot) above the surface on which the robot rests, and/or less than 152,4 mm (6 inches) in one embodiment. The pixels in the image data correspond to a wide field of view around the robot, ranging from 90 - 360 degrees.

[0009] In one embodiment, the fusing further comprises providing color information for each pixel.

[0010] It has proven advantageous, if according to an embodiment the image sensor is a Photonic Mixer Device and the neural network is a multi-layer neural network.

[0011] In one embodiment, the operating mobile robot is a cleaning robot and turns off a brush motor when the operating mobile robot passes over one of the objects before the object is classified to indicate whether the object is a potential hazard.

[0012] In one embodiment, the method further contains the step of slowing down a drive motor when the operating mobile robot passes over one of the objects classified as a potential hazard.

[0013] According to one embodiment, the indicating whether the object is a potential hazard comprises a confidence rating. Preferably, confirmed jamming hazard database stores images taken just before a robot became jammed or otherwise rendered inoperable. If a submitted image is a near match to something identified as a hazard in database, a response to the submitting robot is preferably a probability (90%, 80%, 70%, 50%, etc.) that the object in the image is a hazard. The probability indicates the degree of confidence that the object in the submitted image is the same or similar to a confirmed hazard object

in database. The robot will then preferably avoid the hazard, unless overruled by the user.

[0014] One embodiment of the method further comprises the step of obtaining at least one additional image of the object from a different viewpoint. Preferably, the cleaning robot returns to the object when object classification is indefinite. The additional information can be additional image views of the object from different directions or angles. Preferably, multiple cameras on the cleaning robot can capture different angles, or the cleaning robot can be maneuvered around the object for different views.

[0015] One embodiment of the method further comprises the steps of performing image processing to recognize objects in images captured by the image sensors in the mobile robots; transmitting the images using a wireless transceiver in the mobile cleaning robot; tagging locations of the images on a map; receiving an object classification of the images from a remote object classifier, with the object classification indicating whether the objects are a potential hazard; and returning to and cleaning tagged locations of objects not indicated as a potential hazard.

[0016] The invention further relates to a method for obstacle avoidance in a mobile robot comprising: capturing images of objects with image sensors in mobile robots, the images comprising data corresponding to intensity and depth for each of a plurality of pixels of the images; transmitting the data corresponding to the images to a remote neural network; training the remote neural network using machine learning by identifying and adding tags to the objects; classifying identified objects by indicating whether the objects are a potential hazard; creating a library of identified and classified objects; comparing a new image from an operating mobile robot to the library of identified and classified objects; identifying a match of the new image to the library of identified and classified objects; providing a hazard classification to the operating mobile robot; and altering the operation of the operating mobile robot if the hazard classification is received.

[0017] One embodiment of the aforementioned method further comprises the steps of fusing the intensity and depth data to produce fused data; and providing the fused data to the remote neural network.

[0018] The invention further relates to a system including at least a cleaning robot and a computing system, wherein the system is designed and equipped in such a way that it can perform the method according to one of embodiments described. The cleaning robot and/or the computing system are designed according to one of the respective embodiments described.

[0019] A cleaning robot preferably has a LIDAR (Light Detection and Ranging) turret, which emits a light beam. The light beam in one embodiment is directed in a line or array with a wide field of view, such as 90, 180, 270 or 360 degrees. The light beam can come from a single or multiple light sources, such as a laser diode or LED. The field of view can be generated with optics in a fixed

turret, such as with a PMD sensor (Photonic Mixer Device), or by rotating the turret. Detected reflections of the light beam off objects are preferably used to calculate both the distance to objects and the location of the cleaning robot.

[0020] According to an embodiment, wheels move the cleaning robot, and in particular a brush helps free dirt to be vacuumed into the dirt bag. A cleaning robot preferably includes a processor that operates a program downloaded to memory. The processor communicates with other components using a bus or other electrical connections. In a cleaning mode, wheel motors control the wheels independently to move and steer the robot. Brush and vacuum motors clean the floor, and can preferably be operated in different modes, such as a higher power intensive cleaning mode or a normal power mode.

[0021] In one embodiment PMD (Photonic Mixer Device) module of the cleaning robot includes a laser, a detector and optics for directing a laser beam across a wide field of view. Alternately, instead of a stationary PMD, a turret motor could move a laser and detector to detect objects up to 360 degrees around the cleaning robot. The cleaning robot has one or more sensors, such as a bump sensor indicating contact with an object, proximity sensor indicating closeness to an object, and accelerometer and tilt sensors, which indicate a drop-off (e.g., stairs) or a tilting of the cleaning robot (e.g., upon climbing over an obstacle). Other sensors may preferably be included in other embodiments, such as a dirt sensor for detecting the amount of dirt being vacuumed, a motor current sensor for detecting when the motor is overloaded, such as due to being entangled in something, a floor sensor for detecting the type of floor, and an image sensor (camera) for providing images of the environment and objects.

[0022] Preferably, through the Internet and/or other network(s), the cleaning robot can be controlled, and can send information back to a remote user. A remote server can preferably provide commands, and can process data uploaded from the cleaning robot. In particular, a handheld smartphone or watch can be operated by a user to send commands either directly to cleaning robot (e.g. through Bluetooth, direct RF, a WiFi LAN, etc.) or can send commands through a connection to the Internet. The commands could for example be sent to a server for further processing, then forwarded in modified form to cleaning robot over the Internet.

[0023] Computing system may interact with various user owned or user operated devices via a wide area network such as the Internet. A Client computing system may be implemented, for example, as a consumer device such as a smart phone, other mobile phone, tablet computer, wearable computing device (e.g., smart watch, eyeglasses), desktop computer, laptop computer, and so on.

BRIEF DESCRIPTION OF THE DRAWINGS

[0024]

FIG. 1 is a diagram of a cleaning robot with a LIDAR turret according to an embodiment.

FIG. 2 is a diagram of a cleaning robot and charging station according to an embodiment.

FIG. 3 is a diagram of the underside of a cleaning robot according to an embodiment.

FIG. 4 is a diagram of a smartphone control application display for a cleaning robot according to an embodiment.

FIG. 5 is a diagram of a smart watch control application display for a cleaning robot according to an embodiment.

FIG. 6 is a diagram of a the electronic system for a cleaning robot according to an embodiment.

FIG. 7 is a simplified block diagram of a representative computing system and client computing system usable to implement certain embodiments of the present invention.

FIG. 8 is a diagram of a training mechanism using fused intensity and depth images according to an embodiment.

FIG. 9 is a diagram of an embodiment of a cleaning map indicating the locations of detected objects.

FIG. 10 is a diagram of an embodiment of a system for detecting hazards and classifying objects.

FIG. 11 is a flowchart of an embodiment of a method for detecting hazards and taking corrective action.

DETAILED DESCRIPTION OF THE INVENTION

Overall Architecture

[0025] FIG. 1 is a diagram of a cleaning robot with a LIDAR turret according to an embodiment. A cleaning robot 102 has a LIDAR (Light Detection and Ranging) turret 104 which emits a light beam 106. The light beam in one embodiment is directed in a line or array with a wide field of view, such as 90, 180, 270 or 360 degrees. The light beam can come from a single or multiple light sources, such as a laser diode or LED. The field of view can be generated with optics in a fixed turret 104, such as with a PMD sensor (Photonic Mixer Device), or by rotating the turret 104. Detected reflections of the light beam off objects are used to calculate both the distance

to objects and the location of the cleaning robot. One embodiment of the distance calculation is set forth in US Pat. No. 8,996,172, "Distance sensor system and method," the disclosure of which is incorporated herein by reference. The collected data is also used to create a map, using a SLAM (Simultaneous Location and Mapping) algorithm. One embodiment of a SLAM algorithm is described in US Pat. No. 8,903,589, "Method and apparatus for simultaneous localization and mapping of mobile robot environment," the disclosure of which is incorporated herein by reference.

[0026] FIG. 2 is a diagram of a cleaning robot and charging station according to an embodiment. Cleaning robot 102 with turret 104 is shown. Also shown is a cover 204 which can be opened to access a dirt collection bag and the top side of a brush. Buttons 202 allow basic operations of the robot cleaner, such as starting a cleaning operation. A display 205 provides information to the user. Cleaning robot 102 can dock with a charging station 206, and receive electricity through charging contacts 208.

[0027] FIG. 3 is a diagram of the underside of a cleaning robot according to an embodiment. Wheels 302 move the cleaning robot, and a brush 304 helps free dirt to be vacuumed into the dirt bag.

[0028] FIG. 4 is a diagram of a smartphone control application display for a cleaning robot according to an embodiment. A smartphone 402 has an application that is downloaded to control the cleaning robot. An easy to use interface has a start button 404 to initiate cleaning.

[0029] FIG. 5 is a diagram of a smart watch control application display for a cleaning robot according to an embodiment. Example displays are shown. A display 502 provides an easy to use start button. A display 504 provides the ability to control multiple cleaning robots. A display 506 provides feedback to the user, such as a message that the cleaning robot has finished.

[0030] FIG. 6 is a high level diagram of a the electronic system for a cleaning robot according to an embodiment. A cleaning robot 602 includes a processor 604 that operates a program downloaded to memory 606. The processor communicates with other components using a bus 634 or other electrical connections. In a cleaning mode, wheel motors 608 control the wheels independently to move and steer the robot. Brush and vacuum motors 610 clean the floor, and can be operated in different modes, such as a higher power intensive cleaning mode or a normal power mode.

[0031] PMD (Photonic Mixer Device) module 616 includes a laser 620, a detector 616 and optics 622 for directing a laser beam across a wide field of view. Alternately, instead of a stationary PMD, a turret motor could move a laser and detector to detect objects up to 360 degrees around the cleaning robot. Various sensors provide inputs to processor 604, such as a bump sensor 624 indicating contact with an object, proximity sensor 626 indicating closeness to an object, and accelerometer and tilt sensors 628, which indicate a drop-off (e.g., stairs) or a tilting of the cleaning robot (e.g., upon climbing over an

obstacle). Examples of the usage of such sensors for navigation and other controls of the cleaning robot are set forth in US Pat. No. 8,855,914, "Method and apparatus for traversing corners of a floored area with a robotic surface treatment apparatus," the disclosure of which is incorporated herein by reference. Other sensors may be included in other embodiments, such as a dirt sensor for detecting the amount of dirt being vacuumed, a motor current sensor for detecting when the motor is overloaded, such as due to being entangled in something, a floor sensor for detecting the type of floor, and an image sensor (camera) for providing images of the environment and objects.

[0032] A battery 614 provides power to the rest of the electronics through power connections (not shown). A battery charging circuit 612 provides charging current to battery 614 when the cleaning robot is docked with charging station 206 of FIG. 2. Input buttons 623 allow control of robot cleaner 602 directly, in conjunction with a display 630. Alternately, cleaning robot 602 may be controlled remotely, and send data to remote locations, through transceivers 632.

[0033] Through the Internet 636, and/or other network(s), the cleaning robot can be controlled, and can send information back to a remote user. A remote server 638 can provide commands, and can process data uploaded from the cleaning robot. A handheld smartphone or watch 640 can be operated by a user to send commands either directly to cleaning robot 602 (through Bluetooth, direct RF, a WiFi LAN, etc.) or can send commands through a connection to the Internet 636. The commands could be sent to server 638 for further processing, then forwarded in modified form to cleaning robot 602 over the Internet 636.

[0034] A camera or cameras 642 captures images of objects near the robot cleaner. In one embodiment, at least one camera is positioned to obtain images in front of the robot, showing where the robot is heading. The images are buffered in an image buffer memory 644. The images may be video, or a series of still images. These images are stored for a certain period of time, such as 15 seconds - 2 minutes, or up to 10 minutes, or for an entire cleaning operation between leaving a charging station and returning to the charging station. The images may subsequently be written over.

Computer Systems for Media Platform and Client System

[0035] Various operations described herein may be implemented on computer systems. FIG. 7 shows a simplified block diagram of a representative computing system 702 and client computing system 704 usable to implement certain embodiments of the present invention. In various embodiments, computing system 702 or similar systems may implement the cleaning robot processor system, remote server, or any other computing system described herein or portions thereof. Client computing system 704 or similar systems may implement user de-

vices such as a smartphone or watch with a robot cleaner application.

[0036] Computing system 702 may be one of various types, including processor and memory, a handheld portable device (e.g., an iPhone® cellular phone, an iPad® computing tablet, a PDA), a wearable device (e.g., a Google Glass® head mounted display), a personal computer, a workstation, a mainframe, a kiosk, a server rack, or any other data processing system.

[0037] Computing system 702 may include processing subsystem 710. Processing subsystem 710 may communicate with a number of peripheral systems via bus subsystem 770. These peripheral systems may include I/O subsystem 730, storage subsystem 768, and communications subsystem 740.

[0038] Bus subsystem 770 provides a mechanism for letting the various components and subsystems of server computing system 704 communicate with each other as intended. Although bus subsystem 770 is shown schematically as a single bus, alternative embodiments of the bus subsystem may utilize multiple buses. Bus subsystem 770 may form a local area network that supports communication in processing subsystem 710 and other components of server computing system 702. Bus subsystem 770 may be implemented using various technologies including server racks, hubs, routers, etc. Bus subsystem 770 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. For example, such architectures may include an Industry Standard Architecture (ISA) bus, Micro Channel Architecture (MCA) bus, Enhanced ISA (EISA) bus, Video Electronics Standards Association (VESA) local bus, and Peripheral Component Interconnect (PCI) bus, which may be implemented as a Mezzanine bus manufactured to the IEEE P1386.1 standard, and the like.

[0039] I/O subsystem 730 may include devices and mechanisms for inputting information to computing system 702 and/or for outputting information from or via computing system 702. In general, use of the term "input device" is intended to include all possible types of devices and mechanisms for inputting information to computing system 702. User interface input devices may include, for example, a keyboard, pointing devices such as a mouse or trackball, a touchpad or touch screen incorporated into a display, a scroll wheel, a click wheel, a dial, a button, a switch, a keypad, audio input devices with voice command recognition systems, microphones, and other types of input devices. User interface input devices may also include motion sensing and/or gesture recognition devices such as the Microsoft Kinect® motion sensor that enables users to control and interact with an input device, the Microsoft Xbox® 360 game controller, devices that provide an interface for receiving input using gestures and spoken commands. User interface input devices may also include eye gesture recognition devices such as the Google Glass® blink detector that detects eye ac-

tivity (e.g., "blinking" while taking pictures and/or making a menu selection) from users and transforms the eye gestures as input into an input device (e.g., Google Glass®). Additionally, user interface input devices may include voice recognition sensing devices that enable users to interact with voice recognition systems (e.g., Siri® navigator), through voice commands.

[0040] Other examples of user interface input devices include, without limitation, three dimensional (3D) mice, joysticks or pointing sticks, gamepads and graphic tablets, and audio/visual devices such as speakers, digital cameras, digital camcorders, portable media players, webcams, image scanners, fingerprint scanners, barcode reader 3D scanners, 3D printers, laser rangefinders, and eye gaze tracking devices. Additionally, user interface input devices may include, for example, medical imaging input devices such as computed tomography, magnetic resonance imaging, position emission tomography, medical ultrasonography devices. User interface input devices may also include, for example, audio input devices such as MIDI keyboards, digital musical instruments and the like.

[0041] User interface output devices may include a display subsystem, indicator lights, or non-visual displays such as audio output devices, etc. The display subsystem may be a cathode ray tube (CRT), a flat-panel device, such as that using a liquid crystal display (LCD) or plasma display, a projection device, a touch screen, and the like. In general, use of the term "output device" is intended to include all possible types of devices and mechanisms for outputting information from computing system 702 to a user or other computer. For example, user interface output devices may include, without limitation, a variety of display devices that visually convey text, graphics and audio/video information such as monitors, printers, speakers, headphones, automotive navigation systems, plotters, voice output devices, and modems.

[0042] Processing subsystem 710 controls the operation of computing system 702 and may comprise one or more processing units 712, 714, etc. A processing unit may include one or more processors, including single core processor or multicore processors, one or more cores of processors, or combinations thereof. In some embodiments, processing subsystem 710 may include one or more special purpose co-processors such as graphics processors, digital signal processors (DSPs), or the like. In some embodiments, some or all of the processing units of processing subsystem 710 may be implemented using customized circuits, such as application specific integrated circuits (ASICs), or field programmable gate arrays (FPGAs). In some embodiments, such integrated circuits execute instructions that are stored on the circuit itself. In other embodiments, processing unit(s) may execute instructions stored in local storage, e.g., local storage 722, 724. Any type of processors in any combination may be included in processing unit(s) 712, 714.

[0043] In some embodiments, processing subsystem

710 may be implemented in a modular design that incorporates any number of modules (e.g., blades in a blade server implementation). Each module may include processing unit(s) and local storage. For example, processing subsystem 710 may include processing unit 712 and corresponding local storage 722, and processing unit 714 and corresponding local storage 724.

[0044] Local storage 722, 724 may include volatile storage media (e.g., conventional DRAM, SRAM, SDRAM, or the like) and/or non-volatile storage media (e.g., magnetic or optical disk, flash memory, or the like). Storage media incorporated in local storage 722, 724 may be fixed, removable or upgradeable as desired. Local storage 722, 724 may be physically or logically divided into various subunits such as a system memory, a ROM, and a permanent storage device. The system memory may be a read-and-write memory device or a volatile read-and-write memory, such as dynamic random access memory. The system memory may store some or all of the instructions and data that processing unit(s) 712, 714 need at runtime. The ROM may store static data and instructions that are needed by processing unit(s) 712, 714. The permanent storage device may be a non-volatile read-and-write memory device that may store instructions and data even when a module including one or more processing units 712, 714 and local storage 722, 724 is powered down. The term "storage medium" as used herein includes any medium in which data may be stored indefinitely (subject to overwriting, electrical disturbance, power loss, or the like) and does not include carrier waves and transitory electronic signals propagating wirelessly or over wired connections.

[0045] In some embodiments, local storage 722, 724 may store one or more software programs to be executed by processing unit(s) 712, 714, such as an operating system and/or programs implementing various server functions such as functions of UPP system 102, or any other server(s) associated with UPP system 102. "Software" refers generally to sequences of instructions that, when executed by processing unit(s) 712, 714 cause computing system 702 (or portions thereof) to perform various operations, thus defining one or more specific machine implementations that execute and perform the operations of the software programs. The instructions may be stored as firmware residing in read-only memory and/or program code stored in non-volatile storage media that may be read into volatile working memory for execution by processing unit(s) 712, 714. In some embodiments the instructions may be stored by storage subsystem 768 (e.g., computer readable storage media). In various embodiments, the processing units may execute a variety of programs or code instructions and may maintain multiple concurrently executing programs or processes. At any given time, some or all of the program code to be executed may be resident in local storage 722, 724 and/or in storage subsystem including potentially on one or more storage devices. Software may be implemented as a single program or a collection of separate programs

or program modules that interact as desired. From local storage 722, 724 (or non-local storage described below), processing unit(s) 712, 714 may retrieve program instructions to execute and data to process in order to execute various operations described above.

[0046] Storage subsystem 768 provides a repository or data store for storing information that is used by computing system 702. Storage subsystem 768 provides a tangible non-transitory computer-readable storage medium for storing the basic programming and data constructs that provide the functionality of some embodiments. Software (programs, code modules, instructions) that when executed by processing subsystem 710 provide the functionality described above may be stored in storage subsystem 768. The software may be executed by one or more processing units of processing subsystem 710. Storage subsystem 768 may also provide a repository for storing data used in accordance with the present invention.

[0047] Storage subsystem 768 may include one or more non-transitory memory devices, including volatile and non-volatile memory devices. As shown in FIG. 7, storage subsystem 768 includes a system memory 760 and a computer-readable storage media 752. System memory 760 may include a number of memories including a volatile main RAM for storage of instructions and data during program execution and a non-volatile ROM or flash memory in which fixed instructions are stored. In some implementations, a basic input/output system (BIOS), containing the basic routines that help to transfer information between elements within computing system 702, such as during start-up, may typically be stored in the ROM. The RAM typically contains data and/or program modules that are presently being operated and executed by processing subsystem 710. In some implementations, system memory 760 may include multiple different types of memory, such as static random access memory (SRAM) or dynamic random access memory (DRAM). Storage subsystem 768 may be based on magnetic, optical, semiconductor, or other data storage media. Direct attached storage, storage area networks, network-attached storage, and the like may be used. Any data stores or other collections of data described herein as being produced, consumed, or maintained by a service or server may be stored in storage subsystem 768.

[0048] By way of example, and not limitation, as depicted in FIG. 7, system memory 760 may store application programs 762, which may include client applications, Web browsers, mid-tier applications, relational database management systems (RDBMS), etc., program data 764, and one or more operating systems 766. By way of example, an example operating systems may include various versions of Microsoft Windows®, Apple Macintosh®, and/or Linux operating systems, a variety of commercially-available UNIX® or UNIX-like operating systems (including without limitation the variety of GNU/Linux operating systems, the Google Chrome® OS, and the like) and/or mobile operating systems such as iOS, Windows®

Phone, Android® OS, BlackBerry® 10 OS, and Palm® OS operating systems.

[0049] Computer-readable storage media 752 may store programming and data constructs that provide the functionality of some embodiments. Software (programs, code modules, instructions) that when executed by processing subsystem 710 a processor provide the functionality described above may be stored in storage subsystem 768. By way of example, computer-readable storage media 752 may include non-volatile memory such as a hard disk drive, a magnetic disk drive, an optical disk drive such as a CD ROM, DVD, a Blu-Ray® disk, or other optical media. Computer-readable storage media 752 may include, but is not limited to, Zip® drives, flash memory cards, universal serial bus (USB) flash drives, secure digital (SD) cards, DVD disks, digital video tape, and the like. Computer-readable storage media 752 may also include, solid-state drives (SSD) based on non-volatile memory such as flash-memory based SSDs, enterprise flash drives, solid state ROM, and the like, SSDs based on volatile memory such as solid state RAM, dynamic RAM, static RAM, DRAM-based SSDs, magnetoresistive RAM (MRAM) SSDs, and hybrid SSDs that use a combination of DRAM and flash memory based SSDs. Computer-readable media 752 may provide storage of computer-readable instructions, data structures, program modules, and other data for computing system 702.

[0050] In certain embodiments, storage subsystem 768 may also include a computer-readable storage media reader 750 that may further be connected to computer-readable storage media 752. Together and, optionally, in combination with system memory 760, computer-readable storage media 752 may comprehensively represent remote, local, fixed, and/or removable storage devices plus storage media for storing computer-readable information.

[0051] In certain embodiments, computing system 702 may provide support for executing one or more virtual machines. Computing system 702 may execute a program such as a hypervisor for facilitating the configuring and managing of the virtual machines. Each virtual machine may be allocated memory, compute (e.g., processors, cores), I/O, and networking resources. Each virtual machine typically runs its own operating system, which may be the same as or different from the operating systems executed by other virtual machines executed by computing system 702. Accordingly, multiple operating systems may potentially be run concurrently by computing system 702. Each virtual machine generally runs independently of the other virtual machines.

[0052] Communication subsystem 740 provides an interface to other computer systems and networks. Communication subsystem 740 serves as an interface for receiving data from and transmitting data to other systems from computing system 702. For example, communication subsystem 740 may enable computing system 702 to establish a communication channel to one or more

client computing devices via the Internet for receiving and sending information from and to the client computing devices.

[0053] Communication subsystem 740 may support both wired and/or wireless communication protocols. For example, in certain embodiments, communication subsystem 740 may include radio frequency (RF) transceiver components for accessing wireless voice and/or data networks (e.g., using cellular telephone technology, advanced data network technology, such as 3G, 4G or EDGE (enhanced data rates for global evolution), WiFi (IEEE 802.11 family standards, or other mobile communication technologies, or any combination thereof), global positioning system (GPS) receiver components, and/or other components. In some embodiments communication subsystem 740 may provide wired network connectivity (e.g., Ethernet) in addition to or instead of a wireless interface.

[0054] Communication subsystem 740 may receive and transmit data in various forms. For example, in some embodiments, communication subsystem 740 may receive input communication in the form of structured and/or unstructured data feeds, event streams, event updates, and the like. For example, communication subsystem 740 may be configured to receive (or send) data feeds in real-time from users of social media networks and/or other communication services such as Twitter® feeds, Facebook® updates, web feeds such as Rich Site Summary (RSS) feeds, and/or real-time updates from one or more third party information sources.

[0055] In certain embodiments, communication subsystem 740 may be configured to receive data in the form of continuous data streams, which may include event streams of real-time events and/or event updates, that may be continuous or unbounded in nature with no explicit end. Examples of applications that generate continuous data may include, for example, sensor data applications, financial tickers, network performance measuring tools (e.g. network monitoring and traffic management applications), clickstream analysis tools, automobile traffic monitoring, and the like.

[0056] Communication subsystem 740 may also be configured to output the structured and/or unstructured data feeds, event streams, event updates, and the like to one or more databases that may be in communication with one or more streaming data source computers coupled to computing system 702.

[0057] Communication subsystem 740 may provide a communication interface 742, e.g., a WAN interface, which may provide data communication capability between the local area network (bus subsystem 770) and a larger network, such as the Internet. Conventional or other communications technologies may be used, including wired (e.g., Ethernet, IEEE 802.3 standards) and/or wireless technologies (e.g., Wi-Fi, IEEE 802.11 standards).

[0058] Computing system 702 may operate in response to requests received via communication interface

742. Further, in some embodiments, communication interface 742 may connect computing systems 702 to each other, providing scalable systems capable of managing high volumes of activity. Conventional or other techniques for managing server systems and server farms (collections of server systems that cooperate) may be used, including dynamic resource allocation and reallocation.

[0059] Computing system 702 may interact with various user-owned or user-operated devices via a wide-area network such as the Internet. An example of a user-operated device is shown in FIG. 7 as client computing system 702. Client computing system 704 may be implemented, for example, as a consumer device such as a smart phone, other mobile phone, tablet computer, wearable computing device (e.g., smart watch, eyeglasses), desktop computer, laptop computer, and so on.

[0060] For example, client computing system 704 may communicate with computing system 702 via communication interface 742. Client computing system 704 may include conventional computer components such as processing unit(s) 782, storage device 784, network interface 780, user input device 786, and user output device 788. Client computing system 704 may be a computing device implemented in a variety of form factors, such as a desktop computer, laptop computer, tablet computer, smart phone, other mobile computing device, wearable computing device, or the like.

[0061] Processing unit(s) 782 and storage device 784 may be similar to processing unit(s) 712, 714 and local storage 722, 724 described above. Suitable devices may be selected based on the demands to be placed on client computing system 704; for example, client computing system 704 may be implemented as a "thin" client with limited processing capability or as a high-powered computing device. Client computing system 704 may be provisioned with program code executable by processing unit(s) 782 to enable various interactions with computing system 702 of a message management service such as accessing messages, performing actions on messages, and other interactions described above. Some client computing systems 704 may also interact with a messaging service independently of the message management service.

[0062] Network interface 780 may provide a connection to a wide area network (e.g., the Internet) to which communication interface 740 of computing system 702 is also connected. In various embodiments, network interface 780 may include a wired interface (e.g., Ethernet) and/or a wireless interface implementing various RF data communication standards such as Wi-Fi, Bluetooth, or cellular data network standards (e.g., 3G, 4G, LTE, etc.).

[0063] User input device 786 may include any device (or devices) via which a user may provide signals to client computing system 704; client computing system 704 may interpret the signals as indicative of particular user requests or information. In various embodiments, user input device 786 may include any or all of a keyboard, touch

pad, touch screen, mouse or other pointing device, scroll wheel, click wheel, dial, button, switch, keypad, microphone, and so on.

[0064] User output device 788 may include any device via which client computing system 704 may provide information to a user. For example, user output device 788 may include a display to display images generated by or delivered to client computing system 704. The display may incorporate various image generation technologies, e.g., a liquid crystal display (LCD), light-emitting diode (LED) including organic light-emitting diodes (OLED), projection system, cathode ray tube (CRT), or the like, together with supporting electronics (e.g., digital-to-analog or analog-to-digital converters, signal processors, or the like). Some embodiments may include a device such as a touchscreen that function as both input and output device. In some embodiments, other user output devices 788 may be provided in addition to or instead of a display. Examples include indicator lights, speakers, tactile "display" devices, printers, and so on.

[0065] Some embodiments include electronic components, such as microprocessors, storage and memory that store computer program instructions in a computer readable storage medium. Many of the features described in this specification may be implemented as processes that are specified as a set of program instructions encoded on a computer readable storage medium. When these program instructions are executed by one or more processing units, they cause the processing unit(s) to perform various operation indicated in the program instructions. Examples of program instructions or computer code include machine code, such as is produced by a compiler, and files including higher-level code that are executed by a computer, an electronic component, or a microprocessor using an interpreter. Through suitable programming, processing unit(s) 712, 714 and 782 may provide various functionality for computing system 702 and client computing system 704, including any of the functionality described herein as being performed by a server or client, or other functionality associated with message management services.

[0066] It will be appreciated that computing system 702 and client computing system 704 are illustrative and that variations and modifications are possible. Computer systems used in connection with embodiments of the present invention may have other capabilities not specifically described here. Further, while computing system 702 and client computing system 704 are described with reference to particular blocks, it is to be understood that these blocks are defined for convenience of description and are not intended to imply a particular physical arrangement of component parts. For instance, different blocks may be but need not be located in the same facility, in the same server rack, or on the same motherboard. Further, the blocks need not correspond to physically distinct components. Blocks may be configured to perform various operations, e.g., by programming a processor or providing appropriate control circuitry, and various blocks

might or might not be reconfigurable depending on how the initial configuration is obtained. Embodiments of the present invention may be realized in a variety of apparatus including electronic devices implemented using any combination of circuitry and software.

Training Neural Network with robot images

[0067] FIG. 8 is a diagram of a training mechanism using fused intensity and depth images according to an embodiment. A PMD sensor 802 provides intensity data 804 and depth data 806 for each pixel in an image captured by a robot. The PMD sensor is mounted near the top of a low profile cleaning robot in one embodiment, less than 304,8 mm (a foot) above the surface on which the robot rests, or less than 152,4 mm (6 inches) in one embodiment. The pixels in the image data correspond to a wide field of view around the robot, ranging from 90-360 degrees. The images may appear somewhat curved due to the different locations of the optics for each pixel position. However, by comparing to similarly captured images, themselves distorted, any distortion actually aids in the identification of the object.

[0068] In one embodiment, an optional Fuser Module 808 receives the depth data 806 and intensity data 804 for each pixel in an image and modifies the data. Alternately, the raw data from PMD sensor 802 can simply be buffered or passed to a Machine Learning Module 810, such as a Neural Network (NNT). The Fuser Module 808 can reformat the data or assign each depth and intensity value to an X,Y pixel position corresponding to the pixel being read. In one embodiment, in addition to simple grayscale or intensity data, the sensor or an additional sensor can provide color images, such as RGB data for each pixel in the image.

[0069] In an alternate embodiment, rather than providing depth and intensity/color data separately, the fusion can be any mathematical combination of the depth and intensity data. This provides a single, combined number for each pixel corresponding to intensity and depth. The mathematical operation is multiplying the intensity data by the depth data in one embodiment. Alternately, the intensity and depth data are added. Alternately the intensity is divided by the depth, or vice-versa. Alternately, the depth and intensity are fed into a polynomial equation. In one embodiment, the depth and intensity data are numbers, such as a distance from a reference point for the depth. The intensity can be a number from 1-10, or 1-100. The mathematical relation, or the units used, or any normalization can be used to weight the depth and intensity the same or differently. For example, if it is determined that intensity is a better indicator of an object than depth, the intensity may be given a higher weighting to have a higher impact on the comparison. The weighting can also be done by multiplying by a weighting factor.

[0070] In one embodiment, a Machine Learning/NNT classifier 810 classifies the object and tags it with a clas-

sification. This can be done by comparison to identified objects in a training images library 812. Training images library 812 can initially be populated by humans identifying images, or from online libraries. In one embodiment, the online libraries are replaced by images taken by other robots with a similar low profile viewpoint. The objects can be tagged by a type descriptor (e.g., sock, yarn, ball, key, paper, tissue, etc.) and/or an indication of whether or not it is considered a hazard for the robot (together, tags 816). For example, a small paper may not be a hazard for a cleaning robot, but larger paper may be a hazard.

[0071] The depth information can be used by NNT classifier 810 to locate in the image objects to be identified. The contours of an object on the floor will be revealed by a large change in depth between the perimeter of the object and the wall behind it. Thus, when color and intensity of an object and the background are similar, the depth information allows isolation of the object. In addition, the NNT will learn that only objects on the floor are potential obstacles, not patterns on a wall, things supported or hanging above the robot height (e.g., the bottom of a chair), etc.

[0072] In an alternate embodiment, NNT 818 is trained without using fused depth and intensity data. Instead, NNT 818 is trained using object images from the perspective of low profile cleaning robots. This provides a viewpoint that more closely approximates the viewpoint encountered in the field, compared to online images of objects which are often from a higher perspective. Training with such objects will increase the accuracy of correct object identification and reduce the effect of incorrect identifications on the performance of the robot. The images provided may be only intensity, only depth, or both provided separately, and not fused. The images may be provided by any type of sensor, not just a PMD sensor. For example, images from a detector on a rotating turret may be used. Alternately, another type of depth sensor may be used, such as stereo cameras.

[0073] In one embodiment, the cleaning robot includes a camera, and can upload pictures through a Wireless Local Area Network (WLAN) and the Internet to a server or other computer that performs object recognition. PMD sensor 802 and fuser 808 are on the robot in one embodiment, while machine learning/NNT classifier 810 and training images library 812 are at a remote server. Upon determining whether there are any hazards in an image, a navigation map 818 is annotated with the obstacle location.

[0074] In one embodiment, the obstacle coordinates are downloaded to the robot for populating a navigation map on the robot.

[0075] Cleaning robots in the field can be used to provide the images to populate the training images library 812. As noted above, these can initially be classified using an online library or having a human identify the object. For example, the customer owner of the cleaning robot may be sent the image and be asked to identify the object. The customer can also be asked to provide a tag, indi-

cating whether the customer considers it a hazard to be avoided or not. Alternately, it can be inferred to be a hazard if the customer alters the operation of the robot (e.g., stops the robot and picks up the object). Note that a "hazard" may not simply be something that could damage the robot or adversely impact its performance. A "hazard" could also be something valuable (e.g., keys, coins, a pen, etc.) that the customer does not want lost in a vacuum bag.

[0076] Embodiments provide improved accuracy of obstacle detection by a robot, such as a cleaning robot. Object recognition is used to determine whether an object is an obstacle that should be avoided, or is something that can be run over and picked up by a cleaning robot, such as a dust ball or spider web. Cleaning robots are typically short, and have a unique, floor-level viewpoint of objects, such as seeing chairs and legs from the bottom, not the top. This can cause misidentification when using a neural network trained by online images from other perspectives. Thus, embodiments use images captured by cleaning robots to train a neural network for machine learning. The objects in these images can be identified by humans and/or comparison to images in online libraries with different viewpoints.

[0077] Embodiments provide practical and economic methods and apparatus for obstacle avoidance. The obstacle avoidance algorithm is based on machine deep-learning neural networks algorithms. The high-fidelity model is trained on simultaneous depth and intensity images collected by PMD's time-of-flight camera. The depth and intensity images are fused and then used to train the multi-layer neural network, this provides more differentiation and measurement richness for the neural nets.

[0078] The neural net training process itself is novel as it is based on images collected with the PMD sensor or another Time of Flight (ToF) camera sensor mounted on production like robot, rather than image databases published on online. This ensures physical appearance of the objects mimics true representation of objects as observed by robots running out in the field. In addition, virtual boundaries are used to block the robot from going over obstacles and allowing the sensor to see and capture the data in a highly predictable and controllable environment. The location and the characteristics of obstacles are known a priori and are used to train for both location and object classification.

Intensity and Depth measurements

[0079] There are multiple methods for distance measurement using a laser beam. These include TOF (time-of-flight) measurement, triangulation, and intensity measurement. The TOF measurement utilizes the time difference between emitted and detected laser pulses. This provides high accuracy for long distances, but can be complex. Triangulation provides good results if the detector is sufficiently far from the emitter to provide sufficient accuracy. Intensity measurement is very simple

and can be a good choice for short range distance measurement. This typically involves an intensity modulated laser beam (e.g., with sinusoidal modulation). The difference in the phase of the reflected beam from the phase of the transmitted beam can be mathematically related to the distance to the object causing the reflection. See, e.g., PMD Technologies US Pat. 10,436,883 for a description of the phase calculation. The phase is detected as a different intensity. Since the intensity of a laser beam is essentially constant over short distances, the variation in intensity is due to the change in phase.

PMD Sensor

[0080] There are multiple methods for generating three dimensional depth maps. Some require mechanical scanning, and others require stereo cameras. In one embodiment, an inexpensive PMD sensor (Photonic Mixer Device) is used. This sensor detects both the intensity and the distance in each pixel. Rather than using a single laser beam which is scanned over the environment, a PMD sensor illuminates the entire environment with modulated light from an array of laser diodes, or using optics to divide the light from a single or limited number of laser diodes. The optics could include a hologram or diffraction grating in one example. The distance is measured by using continuous modulation and measuring the phase delay in each pixel.

[0081] In one embodiment, the PMD sensor includes a modulation driver that defines a modulation frequency and signal characteristics, which are important for phase stability and accuracy. A programmable logic device provides the PMD with phase signals and can vary the modulation frequency.

Object classification

[0082] FIG. 9 is a diagram of an embodiment of a cleaning map created by a cleaning robot. A smartphone 901 (or tablet or other display device) shows a cleaning area or map 902 that has been mapped. A location of a robot charging station 904 is indicated. Also indicated are objects detected as the robot moved around, with the location on map 902 of objects indicated by icons 906, 908, 910 and 912. A user can touch an icon, such as icon 910, which is highlighted when touched. An image 914 of the object at the location of icon 910 is then displayed. The user can indicate the object is a potential hazard by swiping the image to the right, or indicate that it is not a hazard by swiping to the left. Alternately, other methods of displaying and selecting can be used, such as a list with yes and no buttons.

[0083] FIG. 10 is a diagram of an embodiment of an asynchronous object classification system. A cleaning robot 1002 uploads images, or portions of images, by WiFi to a router 1004, which is connected to Internet 1006. The images are provided to a robot management server 1008, which can communicate with an application

on a user device, such as a smartphone as shown in FIG. 9. Server 1012 also communicates the images to an Object Classification Server 1010. A machine learning module 1012 can be invoked on server 1010, or on a separate server. Three databases are shown, although they could be combined in a single database, or further segmented. An Object Classification Database 1014 is used to store images from robots and training images for machine learning module 1012. Database 1014 is invoked to classify and identify objects in submitted videos.

[0084] User identified hazards database 1016 stores images that have been manually identified as hazards by users, such as by the mechanism described in FIG. 9. This can either be used in conjunction with database 1014, or separately. In one embodiment, the images are not classified or identified at all. Rather, if a submitted image is a near match to something identified as a hazard in database 1016, a response to the submitting robot is a probability (90%, 80%, 70%, 50%, etc.) that the object in the image is a hazard. The robot will then avoid the hazard, unless overruled by the user as described further below.

[0085] In one embodiment, the robot can identify objects using the intensity/color and depth information from a PMD or similar sensor. The contours of the object can be determined, or simply the changes from a previously generated map can be noted. The object locations are noted on the navigation map, and the robot avoids them while the information is sent to a remote machine learning module for determination whether it is a hazard or not. The robot will asynchronously receive the hazard information later. For objects that are marked as not being hazards, the robot will return to clean those areas. Objects indicated as hazards will continue to be avoided, and the user may be notified of each hazard and its location.

[0086] Confirmed jamming hazard database 1018 stores images taken just before a robot became jammed or otherwise rendered inoperable. Again, if a submitted image is a near match to something identified as a hazard in database 1018, a response to the submitting robot is a probability (90%, 80%, 70%, 50%, etc.) that the object in the image is a hazard. The probability indicates the degree of confidence that the object in the submitted image is the same or similar to a confirmed hazard object in database 1018. The robot will then avoid the hazard, unless overruled by the user.

[0087] Object classification database 1018, in one embodiment, includes tags for each object indicating whether they are a hazard, and a degree of confidence that they are a hazard. If a submitted image is a near match to something identified as a hazard in object classification database 1014, a response to the submitting robot is a probability that the object in the image is a hazard. This response may be instead of, or in addition to, providing an object classification and/or object identification, along with a degree of confidence in the classification and/or identification.

[0088] In one embodiment, submitted messages are compared to images in all three databases, and the response is a weighted combination of the matches from the three databases. In one embodiment, matches from the confirmed jamming hazard database are weighted highest, then matches from the user identified hazard database, then matches from the object classification database.

[0089] In one embodiment, as described above, objects are identified as hazards with a percentage probability. Such hazards are then maintained as marked on a map of the environment and are avoided. The default setting for the probability that an object is a hazard is set to being above a default percentage, such as 10-30%. The default setting could be changed manually by a user, or could be part of a cleaning style, such as set forth in co-pending U.S. Patent Application No. 15/475,983, filed March 31, 2017, entitled "ROBOT WITH AUTOMATIC STYLES," the disclosure of which is hereby incorporated herein by reference. For example, a "fast" style would automatically set the threshold low, such as somewhere in the range 10-20%, while the "thorough" style would set the threshold higher, to try to clean more potential hazards, such as somewhere in the range 30-50%.

Turning off robot brush

[0090] In one embodiment, the robot has a cleaning brush and performs its normal cleaning operation, except that it proceeds with caution over (and around) unknown objects by turning off the brush or reducing the speed. This avoids the primary mode of entanglements. After the object is classified, compared to the user identified database, etc., the robot can return the those areas for a "touch up" cleaning with the brush turned on, if it is safe. In one embodiment, the robot will run over unidentified objects before classification, with the brush off, only if they are detected to be sufficiently small. Sufficiently small may indicated that the robot can pass over the object without contact. Alternately, if the object has been determined to be soft and compressible, an object that will partially contact the brush may be run over.

User Identification of hazards

[0091] In one embodiment, the hazards with a certainty less than a threshold, such as 80-90%, are presented to, or made available to, the user. The display can be the image, such as image 914 in FIG. 9, with or without the probability. The user can then indicate whether the object is indeed a hazard or not. Again, the user can change the default settings to require a higher or lower percentage probability of the object being a hazard before it is presented to the user. The user indications are then added to the user identified hazard database 1016. The images can be sent to robot management server 1008 and then relayed to the user. Alternately, the images can be stored for the use to view the next time the user accesses

the application on the user device. A text or other notification can be sent to the user to prompt review in real time. The user can adjust the settings to enable or disable such a notification. Optionally, the user can also input a description of the object (e.g., sock). The user indications of objects as hazards can be uploaded with the images of the hazards as a tag to a hazard database and also an object identification database. A user can elect whether to be prompted to identify hazards at all, or not to be bothered. Incentives may be offered to the user to identify hazards, such as a discount on future purchases. The user may simply indicate whether it is a hazard or not, such as by a right or left swipe, clicking a yes/no button, doing a tap or double tap or X or other gesture, etc. The user can also be prompted to type in an identification, or select from a list of potential matches identified by the remote object classification server.

[0092] In one embodiment, user identified hazard database 1016 contains not only images identified as hazards by a user, but also images identified as not being a hazard.

[0093] Thus, a submitted image can be compared to both. If the image is more similar to a non-hazard image than a hazard image, it can be indicated to have a low probability of being a hazard. Similarly, confirmed jamming hazard database 1018 may also contain images of objects that turned out not to be jamming hazards. This can be images that jammed a robot, but the robot was able to unjam through reversing the brush. This can also be images where an object is detected, but the robot moves over and cleans the object, with no jam occurring. Again, newly submitted images can be compared to both confirmed jamming hazards and confirmed non-hazards. It should be noted that although jamming is described as an example, any other action that renders the robot inoperable or partially inoperable is also covered by jamming, such as requiring increased power due to partial clogging of the robot or the robot getting stuck and unable to move, or trapped in a small area.

[0094] In practice, the same object may occur in different images as the robot approaches or goes by the object. In one embodiment, the object classification server does image matching, in combination with analyzing the location data tagged with the images, to determine if the same object is indicated in multiple images. The best image of the object is then returned to the robot. The image may also, or instead, be sent directly to a robot management server, and stored in the database section tagged for the user of that robot. The best image can then be accessed by the user, rather than multiple, duplicate images. The best image will typically be one where the object fills most of the image, but does not overflow it, and has a higher probability of matching an identified object or hazard than other images of the same object.

Corrective Action In Response to Object Detection.

[0095] FIG. 11 is a flowchart of an embodiment of a

method for detecting hazards and taking corrective action. Images are recorded in a buffer memory as the robot moves (1102), and are tagged with a timestamp and x, y location coordinates. As described above, the LIDAR can be used to determine the location of the object relative to the robot, and also to determine the robot location on a map. When a jam event occurs, it is recorded with a location (1104). A jam event includes anything which adversely affects the operability of the robot, such as clogging, requiring increased brush or movement motor power, trapping of the robot, immobilizing of the robot, etc. Such events can be detected with one or more sensors. The LIDAR can detect that the cleaning robot isn't moving. A current or voltage sensor can detect excessive power being required by the cleaning robot motor for translational movement, or the brush or other cleaning motor. The buffered images corresponding to the jammed location are then transmitted to the remote server (1106).

[0096] Corrective action can then be taken (1108), such as reversing the direction of rotation of the cleaning brush, allowing the brush to free spin and then backing up the robot, reversing the direction of the robot, increasing the robot brush or translational movement motor power, etc. If the jam event is not corrected, the user is notified (1110). The notification can be an indication on the robot app, a separate text message, or any other notification. The user can be directed to take appropriate action, such as clean the brush, remove, empty and replace the dirt container, pick up and move the robot to an open area, etc. The user can optionally be prompted to identify the object at the location of the jam event, and the user identification can be recorded and transmitted to the remote server (1112). Thus, the remote server may receive multiple types of tagged images: images tagged as causing a jam that was automatically overcome, images tagged as causing a jam that was not overcome, and images that caused a jam and have been labeled by a user.

[0097] Embodiments provide practical and economic methods and apparatus for asynchronously classifying images provided by a robot. Doing object identification using the processor in a robot in real time would make the robot more expensive. Since the robot takes a fair amount of time to do cleaning, and objects can be bypassed and returned to, real-time decisions are not needed such as would be needed in self-driving cars, for example. In a reconnaissance/exploratory or first cleaning pass, unidentified objects are simply avoided. Images of the object are uploaded over the Internet to a remote object detection and classification system, and the location is indicated by the cleaning robot. When the remote system subsequently returns an object identification or classification, the object can be indicated as something to be avoided, or the cleaning robot can return to the location and clean over the object if it is determined not to be a hazard. The classification of the object need not identify the object, but can simply be an indication that it is a potential hazard to the robot. New images and objects

are compared to the tagged hazards database to identify hazards. This eliminates the need for object recognition and classification - the robot will simply know that such an object is a hazard and has impacted other robots adversely.

[0098] In one embodiment, the cleaning robot can obtain additional information about the object. The additional information, or some additional information may be obtained upon the first encounter with the object. Alternately, the cleaning robot may return to the object when object classification is indefinite. The additional information can be additional image views of the object from different directions or angles. Multiple cameras on the cleaning robot can capture different angles, or the cleaning robot can be maneuvered around the object for different views. A bump or pressure sensor can be used with slight contact with the object to determine if it is hard or soft. For example, after detecting initial contact, the robot can continue to move for 12,7 mm ($\frac{1}{2}$ inch) to see if the object compresses or moves. The difference between the object moving (indicating it is hard) and compressing (indicating it is soft) can be determined by the amount of pressure detected on a bump sensor (with, in general, more pressure from a hard, moving object) and/or images or the LIDAR indicating that the object has moved after the robot initiates contact and then withdraws from contact.

[0099] Alternately, an identification of hardness or softness can come from identifying the type of material through machine learning, based on depth and intensity/color information.

Questionnaire

[0100] In one embodiment, a user completes a questionnaire and the answers are used to filter the potential object matches. For example, if the user does not have a pet, dog poop can be eliminated from the possible object classification. Conversely, if a user has a dog, dog poop can be added to the list of potential objects with a higher weighting of likelihood of a match. If a user has kids, toys can be weighted higher, or eliminated if a user doesn't have kids. Indicating birthdays can be used to increase the likelihood weighting of wrapping paper and ribbons around the time of the birthday. Other calendar dates can be used to increase the likelihood weighting, such as wrapping paper or ornaments around Christmas.

[0101] In one embodiment, the type of object detected may change the cleaning mode. For example, the detection of a throw rug on a wood or tile floor can change the brush mode for a vacuum cleaner robot. Different floor types may be stored as images indicating they are not a hazard, and also being tagged with the preferred cleaning mode.

[0102] In one embodiment, the robot may determine the image is too dark, or the remote server may indicate this with a request for a better illuminated image. The robot may have a light source that can be directed to the

object and can be turned on. The light source could be visible or IR. Alternately, the robot may communicate via WiFi over a home network with a lighting controller to have a light turned on in the room where the object is located.

Machine learning

[0103] In one embodiment, machine learning is used to determine image types and whether they are a hazard. A test environment may be set up with multiple known objects. These objects can both be tagged by a human tester, and also can be identified by test robots probing them, running over them, etc. The test objects are selected from a group typically found on the floor of a home, such as socks, wires, papers, dog poop, dog food, string, pencils, etc., etc.

[0104] In one embodiment, the type of room is identified, and the objects are weighted based on their likelihood of being in such a room. For example, a kitchen may be more likely to have food, utensils, etc. A bathroom is more likely to have towels, toothbrushes, etc. A closet is more likely to have socks and other clothing.

Conclusion

[0105] While the invention has been described with respect to specific embodiments, one skilled in the art will recognize that numerous modifications are possible. Embodiments of the invention may be realized using a variety of computer systems and communication technologies including but not limited to specific examples described herein.

[0106] Embodiments of the present invention may be realized using any combination of dedicated components and/or programmable processors and/or other programmable devices. The various processes described herein may be implemented on the same processor or different processors in any combination. Where components are described as being configured to perform certain operations, such configuration may be accomplished, e.g., by designing electronic circuits to perform the operation, by programming programmable electronic circuits (such as microprocessors) to perform the operation, or any combination thereof. Further, while the embodiments described above may make reference to specific hardware and software components, those skilled in the art will appreciate that different combinations of hardware and/or software components may also be used and that particular operations described as being implemented in hardware might also be implemented in software or vice versa.

[0107] Computer programs incorporating various features of the present invention may be encoded and stored on various computer readable storage media; suitable media include magnetic disk or tape, optical storage media such as compact disk (CD) or DVD (digital versatile disk), flash memory, and other non-transitory media.

Computer readable media encoded with the program code may be packaged with a compatible electronic device, or the program code may be provided separately from electronic devices (e.g., via Internet download or as a separately packaged computer-readable storage medium).

[0108] Thus, although the invention has been described with respect to specific embodiments, it will be appreciated that the invention is intended to cover all modifications and equivalents within the scope of the following claims.

Claims

1. A method for obstacle avoidance in a mobile robot (102) comprising:
 - capturing images of objects with image sensors in mobile robots (102), wherein the image sensors have a perspective within 304,8 mm (one foot) of a surface upon which the mobile robots (102) travel, or wherein the images comprising data corresponding to intensity and depth for each of a plurality of pixels of the images;
 - transmitting data corresponding to the images to a remote neural network (810);
 - training the neural network (810) using machine learning by identifying and adding tags to the objects;
 - classifying identified objects by indicating whether the objects are a potential hazard;
 - creating a library of identified and classified objects;
 - comparing a new image from an operating mobile robot (102) to the library of identified and classified objects;
 - identifying a match of the new image to the library of identified and classified objects;
 - providing a hazard classification to the operating mobile robot (102); and
 - altering the operation of the operating mobile robot (102) if the hazard classification is received.
2. The method of claim 1 wherein the data corresponding to the images comprises intensity and depth data for each of a plurality of pixels of the images.
3. The method of claim 2 further comprising: fusing the intensity and depth data to produce fused data; and providing the fused data to the remote neural network (810).
4. The method of claim 3 wherein fusing comprises providing the intensity and depth data for each pixel in an image.

5. The method of claim 3 or 4 wherein the fusing further comprises providing color information for each pixel.
6. The method of one of the claims 1 to 5 wherein the image sensor is a Photonic Mixer Device (616) and the neural network (810) is a multi-layer neural network. 5
7. The method of one of the claims 1 to 6 wherein the operating mobile robot (102) is a cleaning robot and turns off a brush motor when the operating mobile robot (102) passes over one of the objects before the object is classified to indicate whether the object is a potential hazard. 10
8. The method of one of the claims 1 to 7 further containing:
slowing down a drive motor when the operating mobile robot (102) passes over one of the objects classified as a potential hazard. 15 20
9. The method of one of the claims 1 to 8 wherein indicating whether the object is a potential hazard comprises a confidence rating. 25
10. The method of one of the claims 1 to 9 further comprising obtaining at least one additional image of the object from a different viewpoint.
11. The method of one of the claims 1 to 10 further comprising: 30
performing image processing to recognize objects in images captured by the image sensors in the mobile robot (102); 35
transmitting the images using a wireless transceiver in the mobile cleaning robot (102);
tagging locations of the images on a map (902);
receiving an object classification of the images from a remote object classifier, with the object classification indicating whether the objects are a potential hazard; and 40
returning to and cleaning tagged locations of objects not indicated as a potential hazard. 45
12. System including at least a mobile robot (102) and a computing system (702), wherein the system is designed and equipped in such a way that it can perform the method according to one of the claims 1 to 11. 50
13. System according to claim 12, wherein the system further comprises a client computing system (704), in particular that the client computing system (704) is implemented as a consumer device such as a smart phone, other mobile phone, tablet computer, wearable computing device, desktop computer or laptop computer. 55
14. Computer program product comprising instructions, which, when the program is executed by at least one computer, cause the computer to perform the steps of the method of one of the claims 1 to 11.

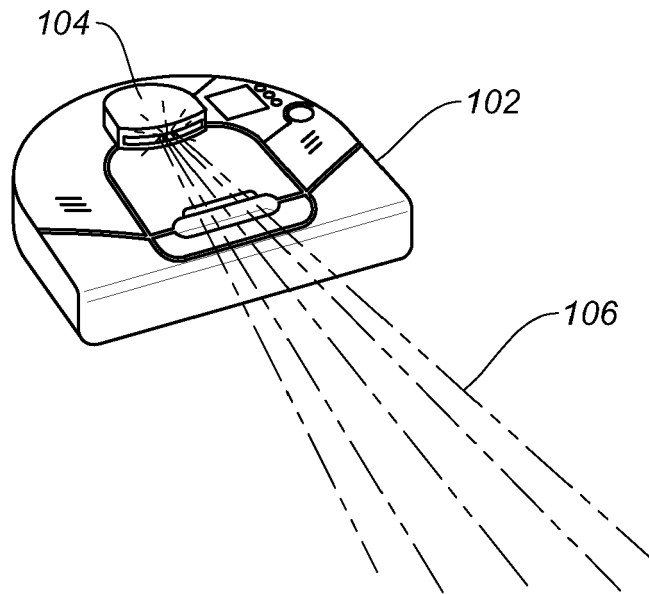


FIG. 1

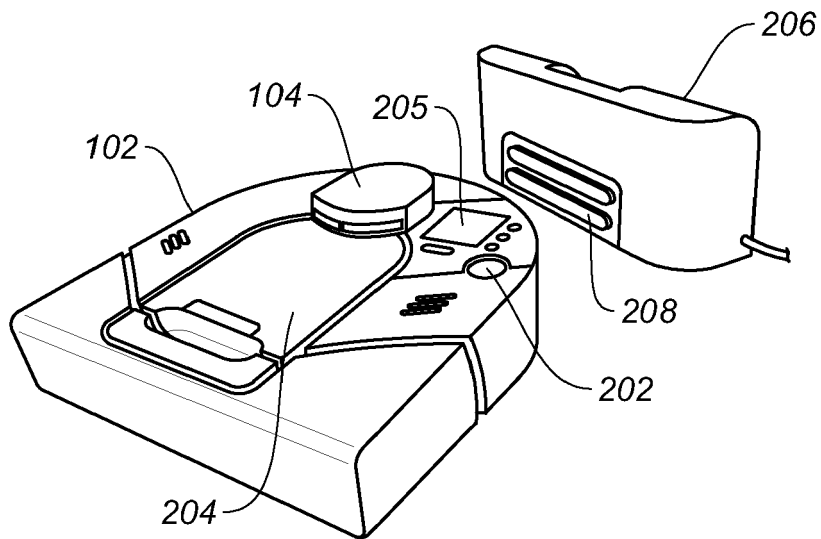


FIG. 2

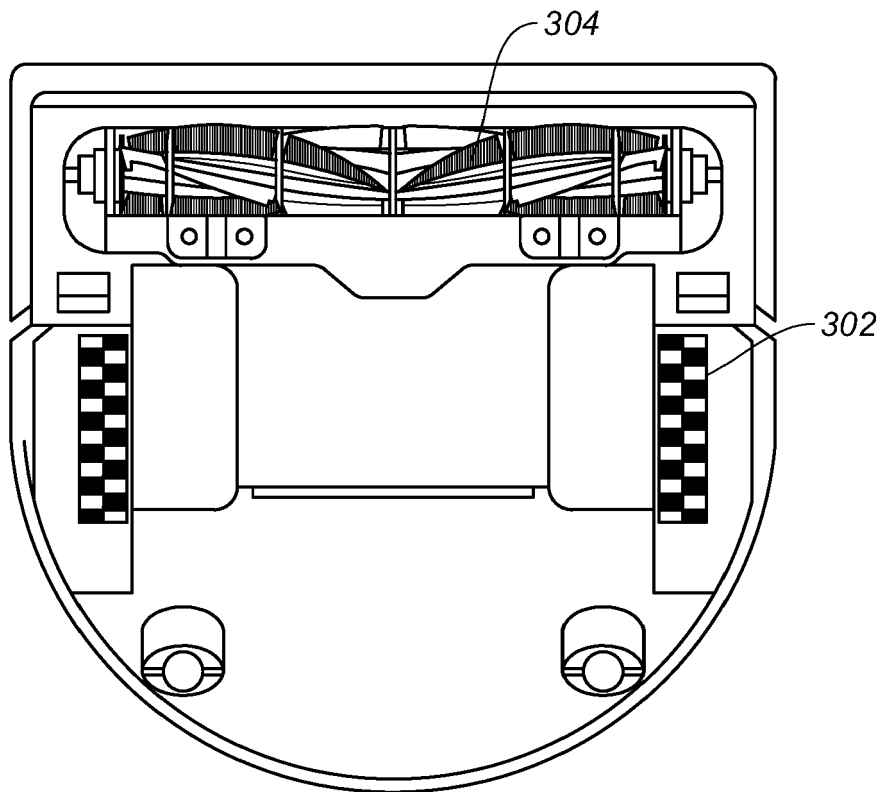


FIG. 3

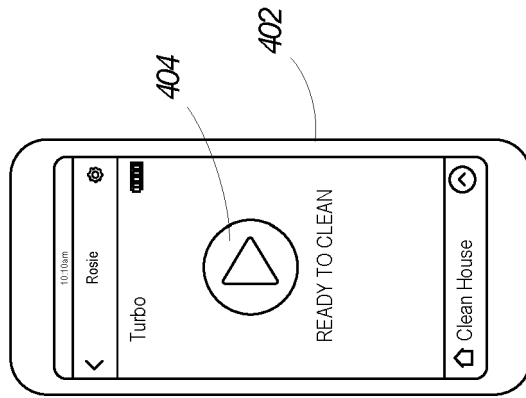


FIG. 4

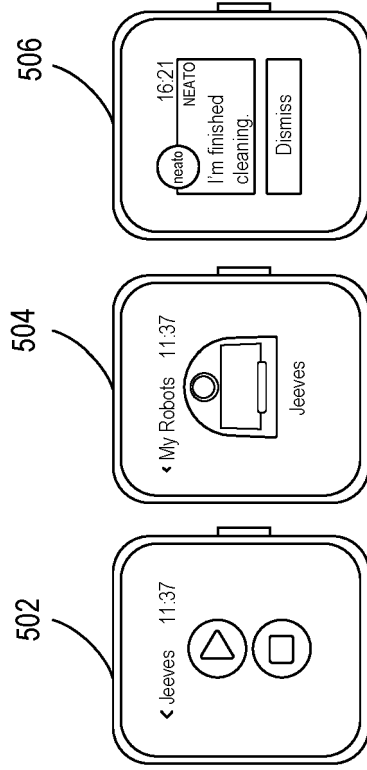


FIG. 5

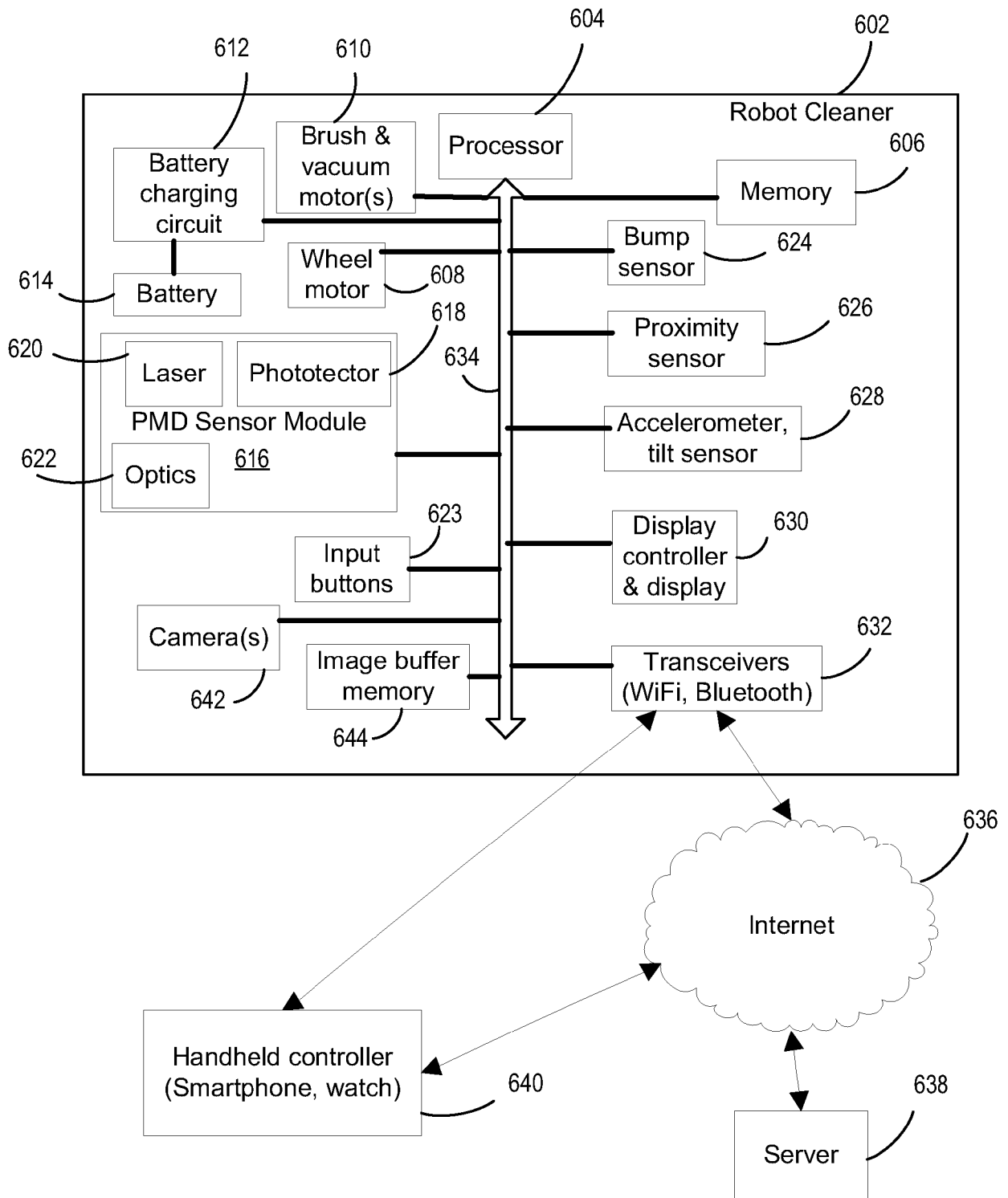


FIG. 6

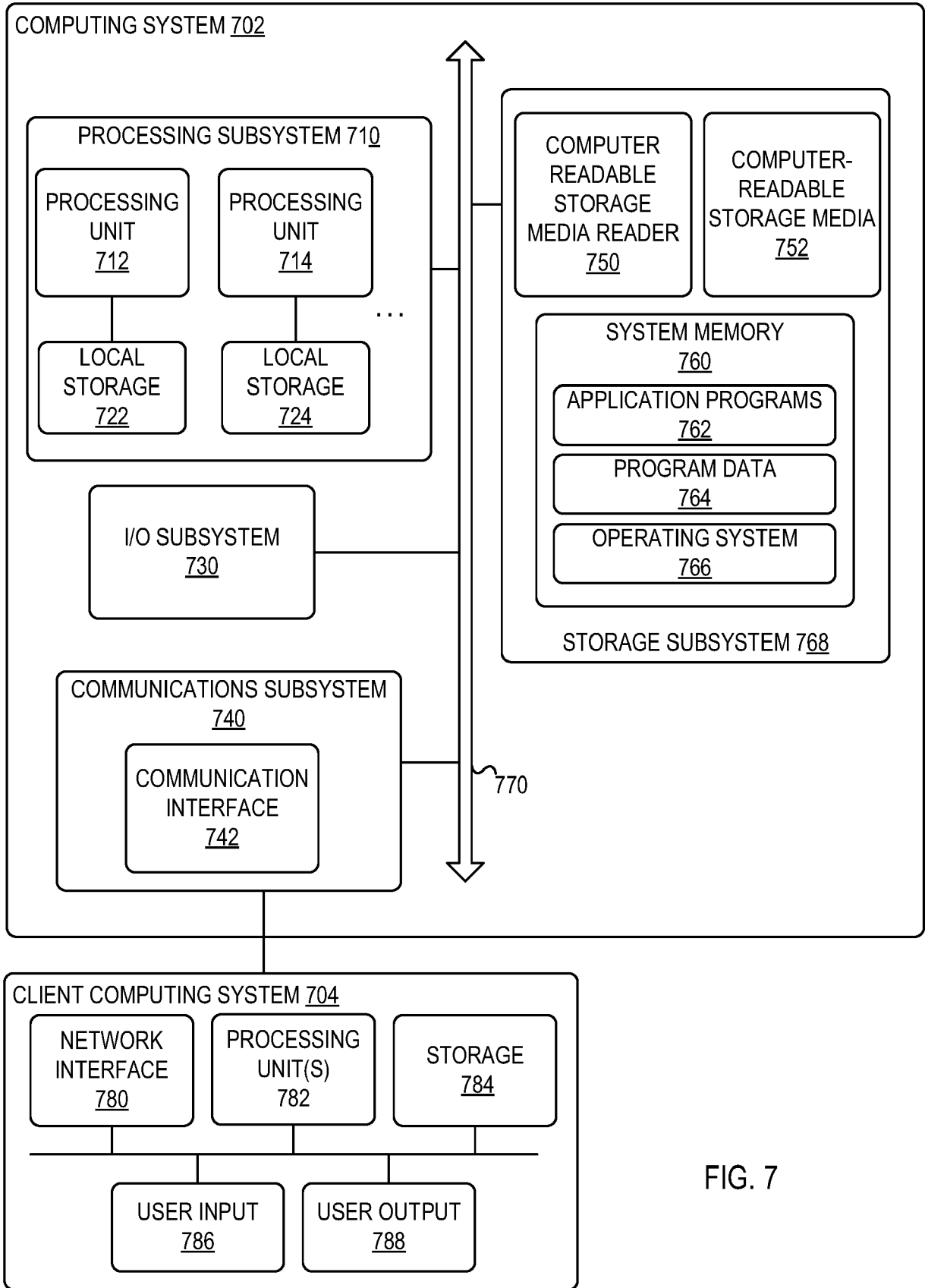


FIG. 7

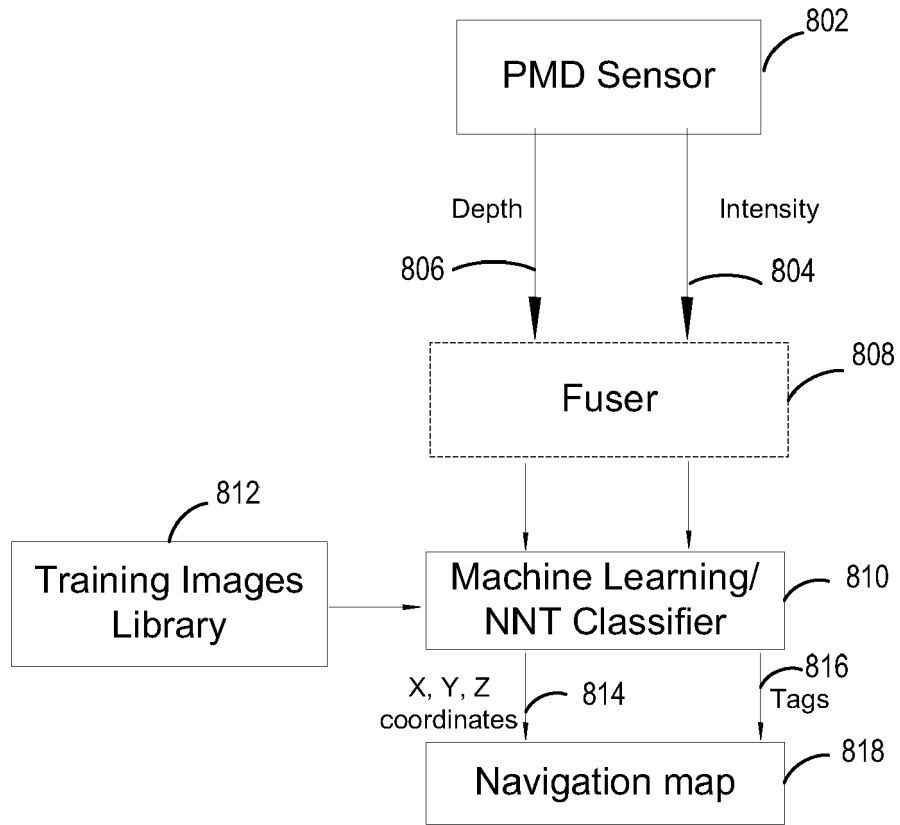


FIG. 8

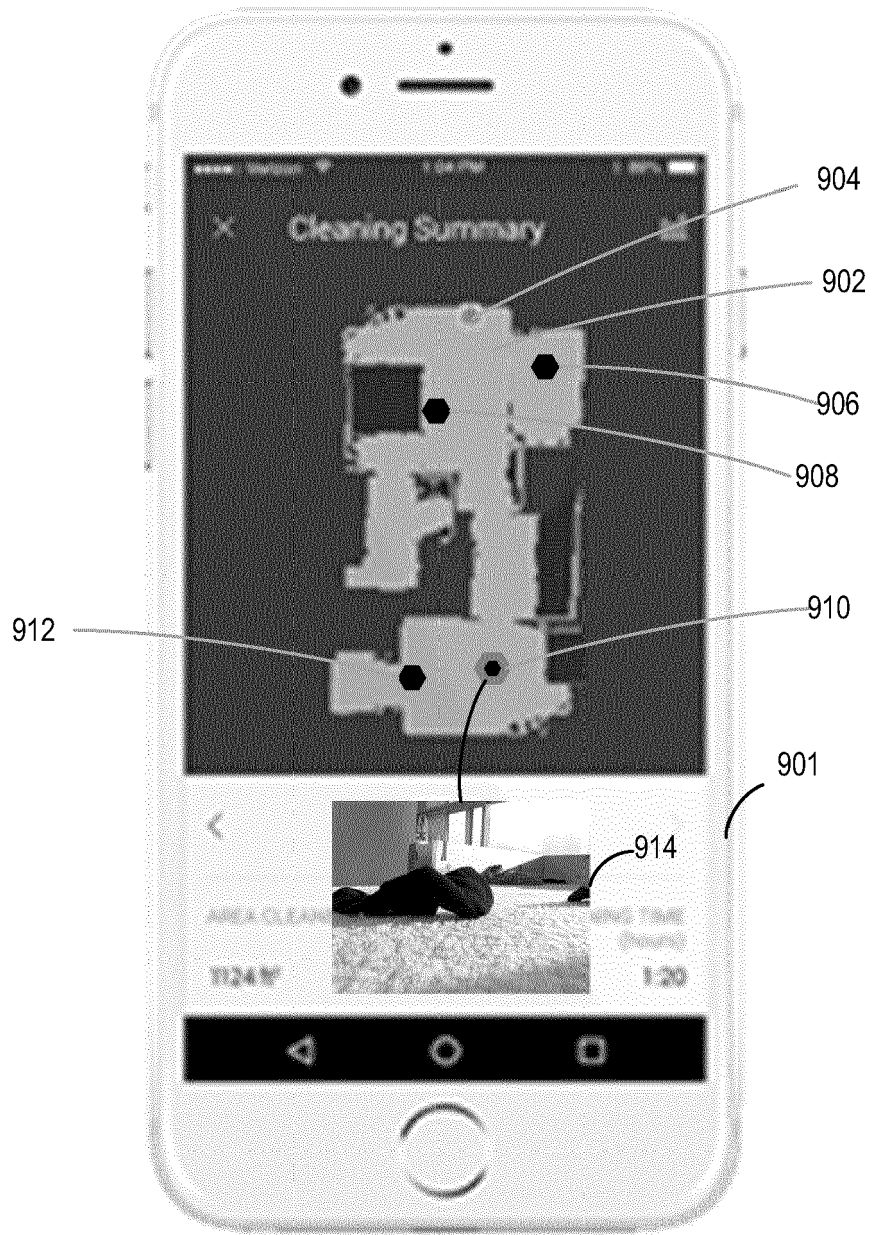


FIG. 9

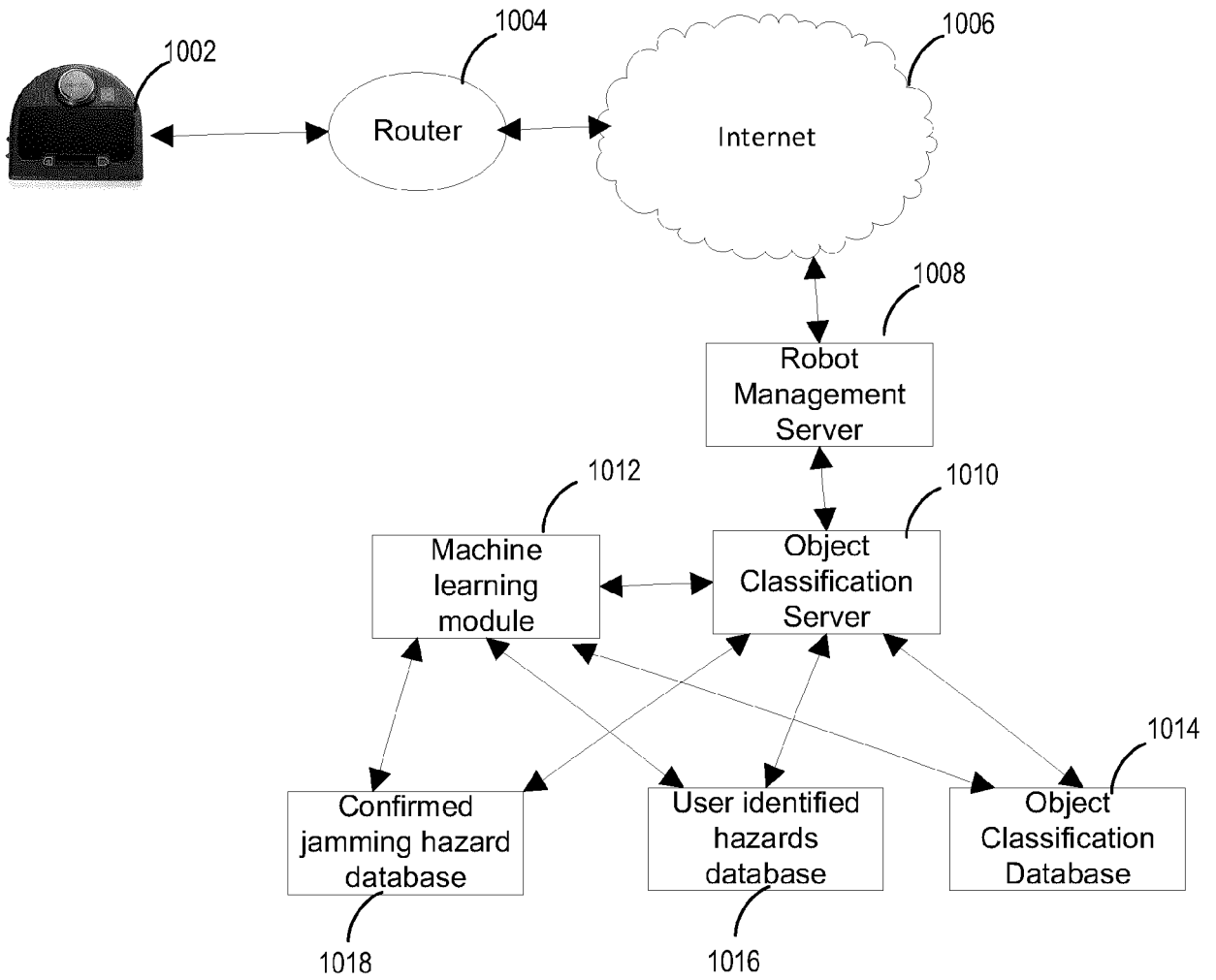


FIG. 10

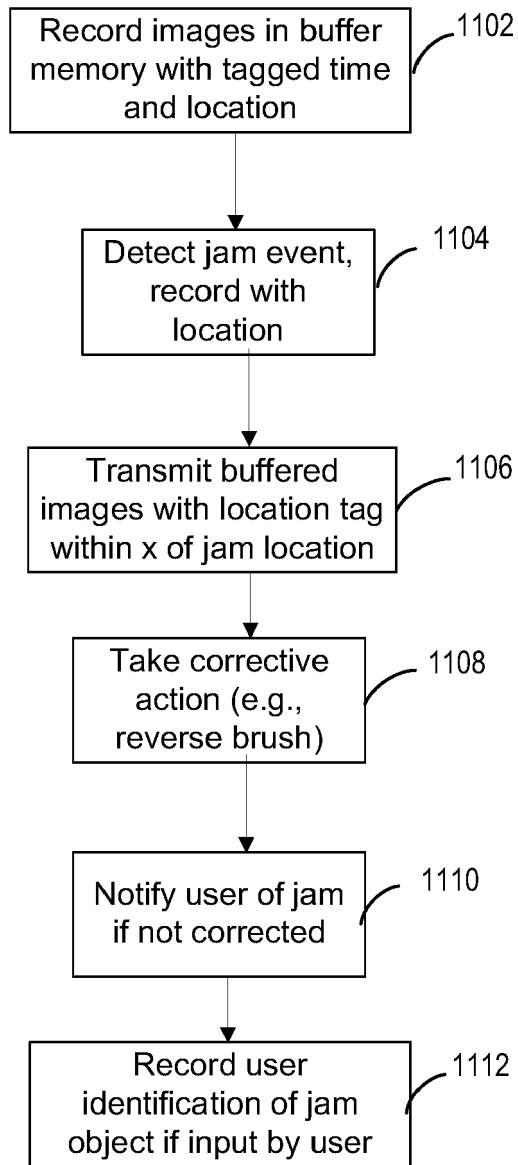


FIG. 11



EUROPEAN SEARCH REPORT

Application Number

EP 23 16 7060

5

10

15

20

25

30

35

40

45

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
X	US 2021/370511 A1 (HONG SOONHYUK [KR]) 2 December 2021 (2021-12-02) * paragraphs [0004], [0043], [0044], [0050], [0054] - [0056], [0059], [0063], [0067], [0071] * * paragraphs [0090], [0091], [0097], [0121], [0123], [0125], [0135] - [0138] * * paragraphs [0142], [0147], [0149], [0192], [0193], [0230], [0278] * -----	1-14	INV. G06V20/58 G06V20/10 G06F18/25 G06V10/764 G06V10/82
X	WO 2021/172936 A1 (LG ELECTRONICS INC [KR]) 2 September 2021 (2021-09-02) * paragraphs [0015], [0017], [0025], [0057], [0076], [0077], [0134], [0135], [0178], [0207] * -----	1-14	
X	US 2020/341479 A1 (KIM JUNGHWAN [KR] ET AL) 29 October 2020 (2020-10-29) * paragraphs [0039], [0079], [0085], [0092], [0093], [0158] - [0160], [0187], [0219], [0220], [0236], [0237], [0258] * -----	1-14	TECHNICAL FIELDS SEARCHED (IPC) G06V G06F
X	US 2019/320867 A1 (NOH DONGKI [KR] ET AL) 24 October 2019 (2019-10-24) * paragraphs [0082], [0180], [0218] - [0221], [0235] - [0239], [0276], [0277], [0286] * -----	1-14	

The present search report has been drawn up for all claims

2

50

Place of search The Hague	Date of completion of the search 20 August 2023	Examiner Androulidakis, Iosif
-------------------------------------	---	---

55

EPO FORM 1503 03:82 (P04C01)

CATEGORY OF CITED DOCUMENTS
 X : particularly relevant if taken alone
 Y : particularly relevant if combined with another document of the same category
 A : technological background
 O : non-written disclosure
 P : intermediate document

T : theory or principle underlying the invention
 E : earlier patent document, but published on, or after the filing date
 D : document cited in the application
 L : document cited for other reasons

 & : member of the same patent family, corresponding document

ANNEX TO THE EUROPEAN SEARCH REPORT
ON EUROPEAN PATENT APPLICATION NO.

EP 23 16 7060

5 This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

20-08-2023

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2021370511 A1	02-12-2021	CN 112714684 A	27-04-2021
		EP 3774202 A1	17-02-2021
		KR 20200036677 A	07-04-2020
		KR 20200036678 A	07-04-2020
		US 2021370511 A1	02-12-2021
WO 2021172936 A1	02-09-2021	EP 4110560 A1	04-01-2023
		KR 20210110064 A	07-09-2021
		US 2023091839 A1	23-03-2023
		WO 2021172936 A1	02-09-2021
US 2020341479 A1	29-10-2020	AU 2018356126 A1	11-06-2020
		CN 111479662 A	31-07-2020
		EP 3702111 A1	02-09-2020
		US 2020341479 A1	29-10-2020
		WO 2019083291 A1	02-05-2019
US 2019320867 A1	24-10-2019	AU 2017363769 A1	11-07-2019
		EP 3546139 A1	02-10-2019
		EP 3950234 A1	09-02-2022
		KR 20180058511 A	01-06-2018
		TW 201825037 A	16-07-2018
		US 2019320867 A1	24-10-2019
		US 2022265106 A1	25-08-2022
WO 2018097574 A1	31-05-2018		

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US 20150289743 A [0002]
- US 20160167226 A [0002]
- US 8996172 B [0025]
- US 8903589 B [0025]
- US 8855914 B [0031]
- US 10436883 B [0079]
- US 47598317 [0089]