



(19) **United States**  
(12) **Patent Application Publication**  
**Boulkenafed et al.**

(10) **Pub. No.: US 2014/0358475 A1**  
(43) **Pub. Date: Dec. 4, 2014**

(54) **BODY POSTURE TRACKING**

**Publication Classification**

(71) Applicant: **Dassault Systemes**, Velizy Villacoublay (FR)  
(72) Inventors: **Malika Boulkenafed**, Courbevoie (FR); **Fabrice Francis Michel**, Montrouge (FR)  
(73) Assignee: **Dassault Systemes**, Velizy Villacoublay (FR)

(51) **Int. Cl.**  
**G01B 21/16** (2006.01)  
(52) **U.S. Cl.**  
CPC ..... **G01B 21/16** (2013.01)  
USPC ..... **702/152**

(21) Appl. No.: **14/288,286**

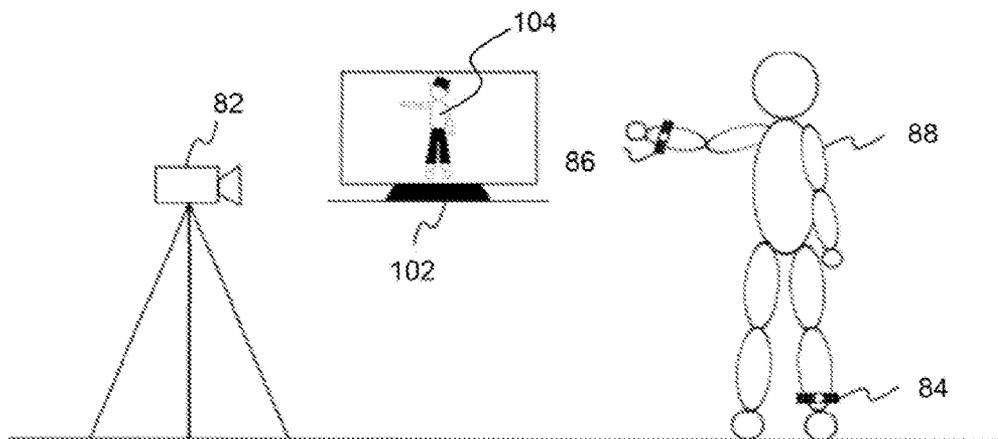
(22) Filed: **May 27, 2014**

(30) **Foreign Application Priority Data**

May 29, 2013 (EP) ..... 13305703.4

(57) **ABSTRACT**

It is provided a computer-implemented method for performing body posture tracking, comprising the steps of collecting (S10) depth measurements of a body with at least one depth sensor; collecting (S20) inertial measurements with at least one inertial sensor attached to the body; and determining (S30) at least one posture of the body as a function of the depth measurements and the inertial measurements. Such a method improves the field of body posture tracking.



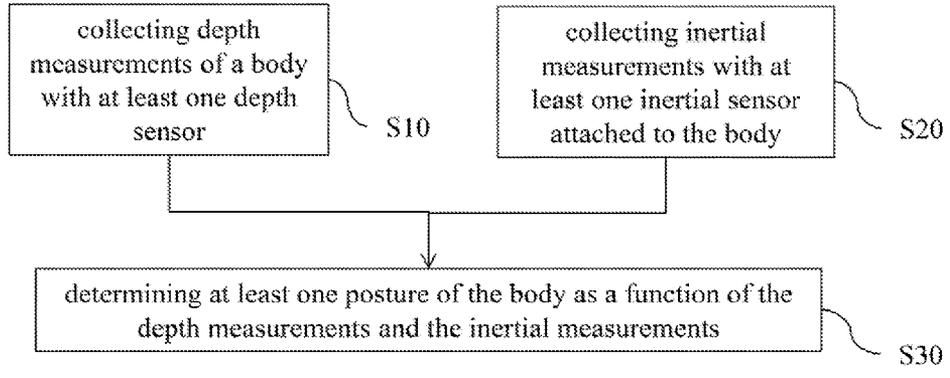


FIG. 1

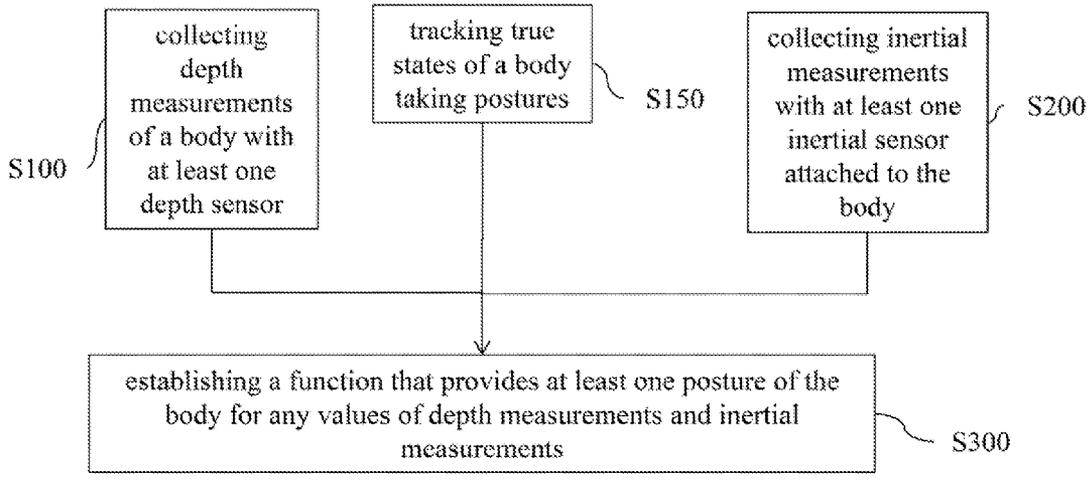


FIG. 2

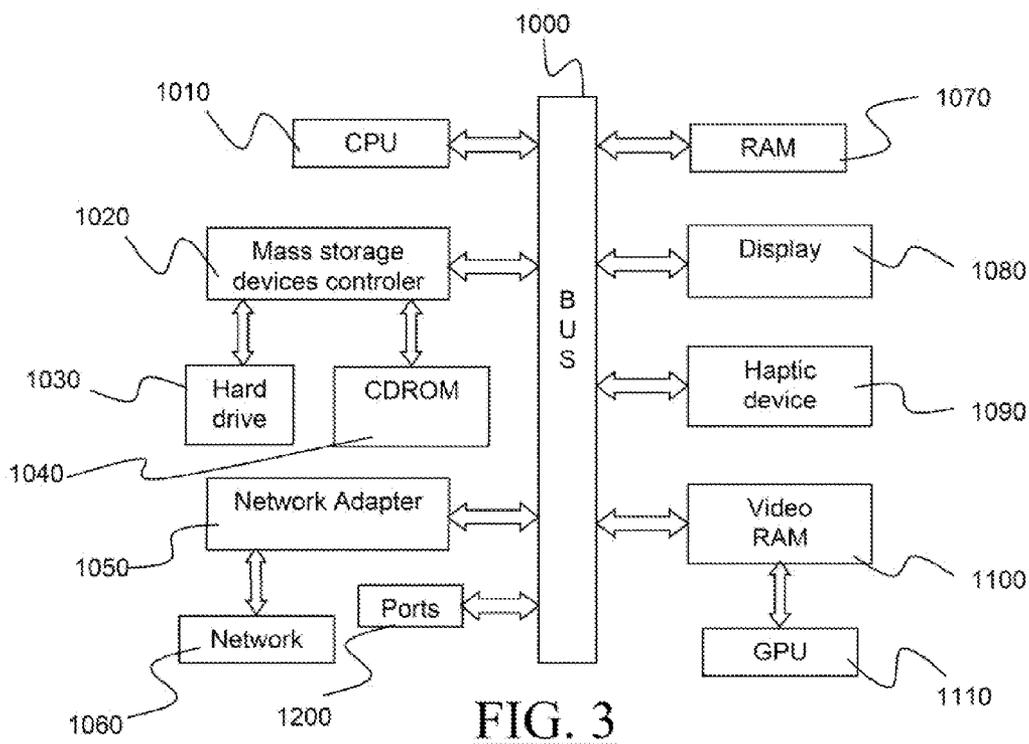


FIG. 3

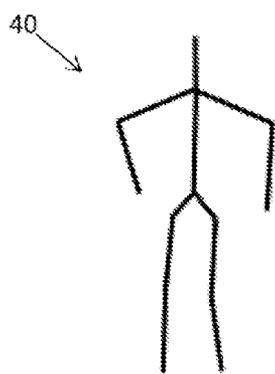


FIG. 4

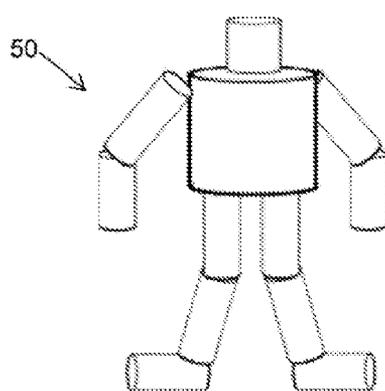


FIG. 5

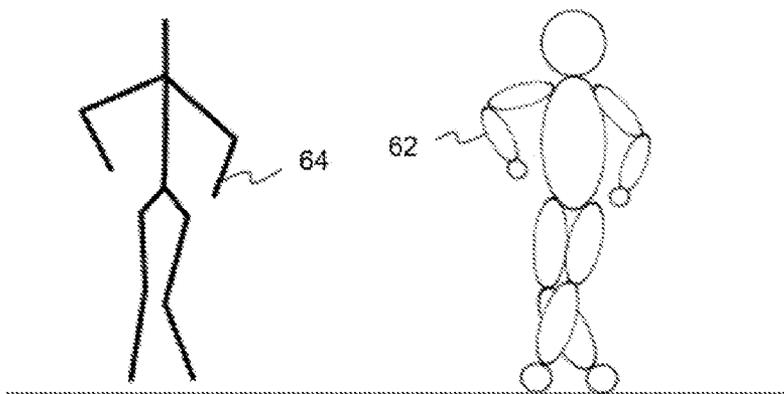


FIG. 6

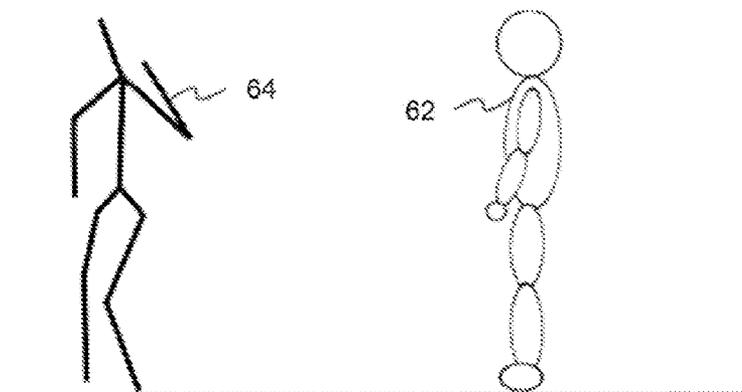


FIG. 7

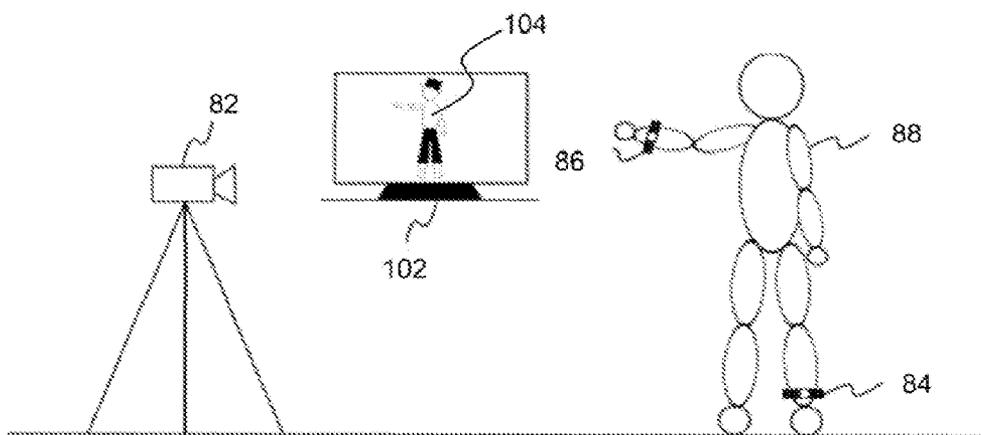


FIG. 8

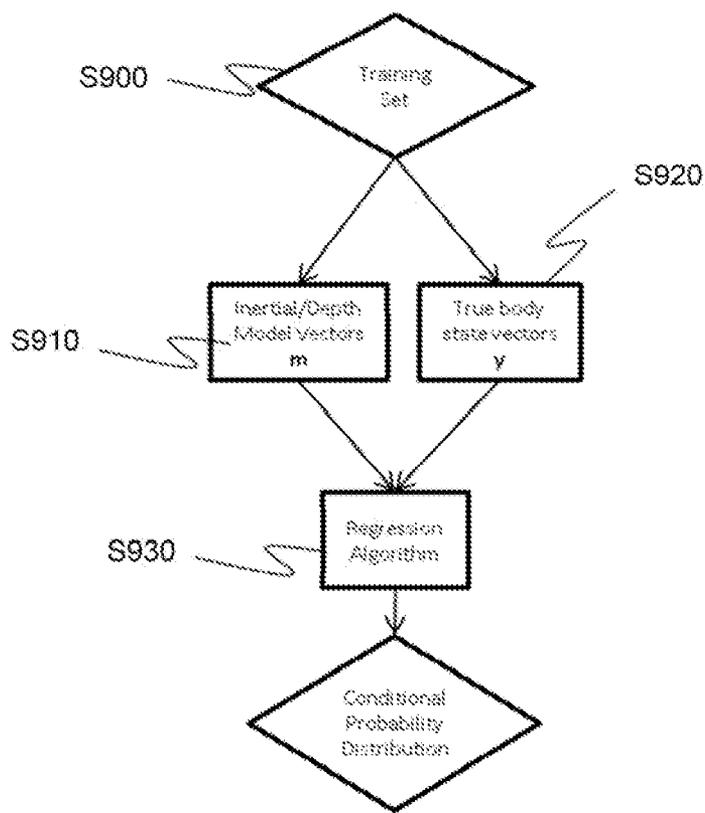


FIG. 9

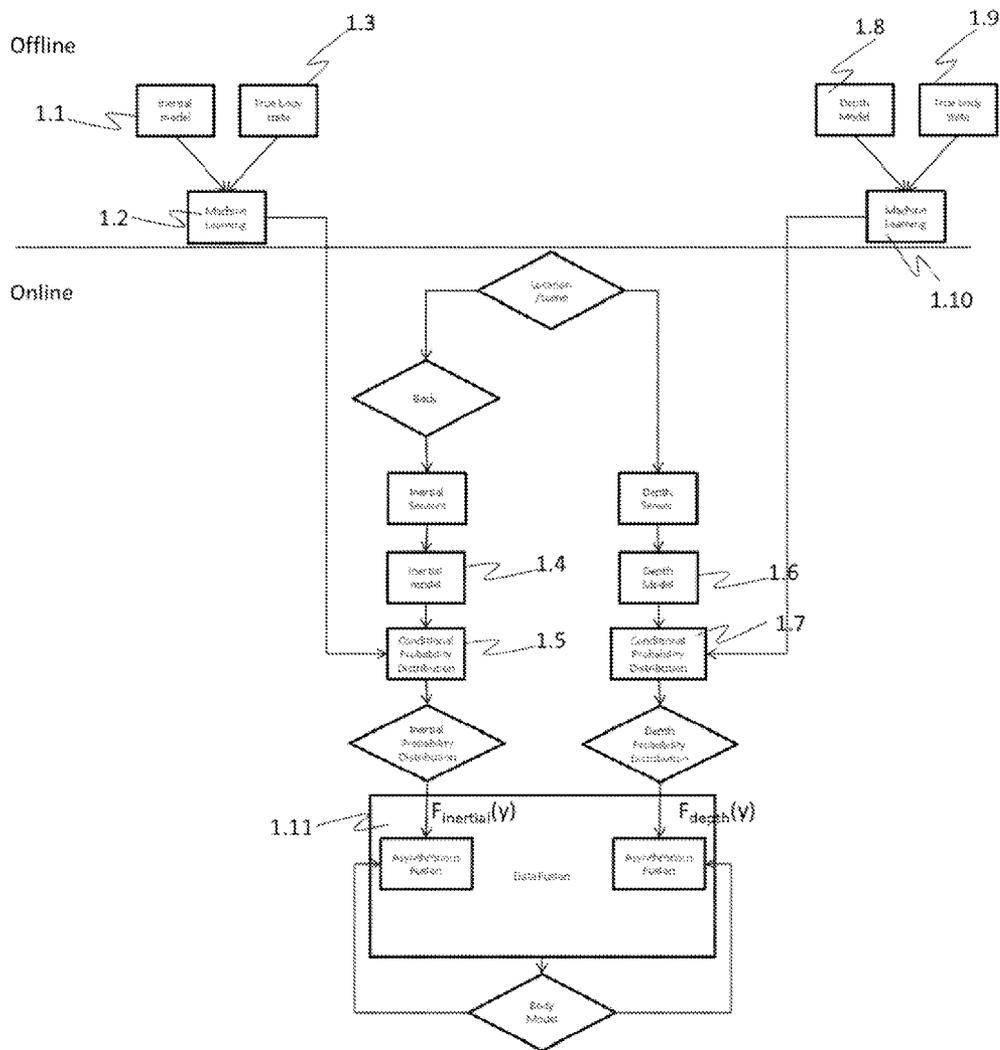


FIG. 10

**BODY POSTURE TRACKING**

## RELATED APPLICATION(S)

[0001] This application claims priority under 35 U.S.C. §119 or 365 to European Application No. 13305703.4, filed May 29, 2013.

[0002] The entire teachings of the above application(s) are incorporated herein by reference.

## FIELD OF THE INVENTION

[0003] The invention relates to the field of computer programs and systems, and more specifically to computer-implemented methods, programs, data storage media and systems for performing body posture tracking.

## BACKGROUND

[0004] Body posture tracking is related to body motion tracking. Systems offering such tracking capacities allow the recognition of a real body's posture or motion (i.e. a time-series of postures). The idea is to describe such posture or motion in a computerized form, for example so as to make use of such computerized data. This may be useful in many applications, including non-exhaustively virtual reality applications, video games, and surgical assistance.

[0005] Body posture tracking may be performed based on different technologies. Depth sensors may be used to provide depth measurements (e.g. a depth map) of a body evolving in a location covered by the sensors. Another technology involves inertial sensors that are attached to body parts to provide their 3D positions. Stereo vision cameras can also provide 3D positions. Besides, data fusion has also been used to combine different data in order to predict a posture with relatively high accuracy. So far, several data fusion approaches have been developed.

[0006] Data fusion involving depth sensors is a very active field, since current depth sensors, when used alone, fail at describing a three dimensional scene with high fidelity. The two major drawbacks of depth sensors are their limitations in the field of view, resulting in blind spots in the 3D scene, and the lack of precision of the depth map with low discriminative power in close by depths and high sensor noise around object edges.

[0007] Data fusion is used in this context to cope with these problems, but currently, accuracy is the subject that has attracted most attention, both in the industry and the academic context. In the industry, the most notable usage of data fusion for improving the accuracy of the depth sensors can be found at "3Gear Systems", where the fusion of two depth sensors and machine learning are associated in order to track accurately the movements of the fingers for a haptic interface. This application is not, to this day, realizable with only one depth sensor.

[0008] Usages of data fusion on depth sensors are also largely discussed in academic research. The most notable approaches include:

[0009] The fusion of a depth sensor, a stereo vision camera and visual cues extracted in luminance images ("Sensor Fusion for 3D Human Body Tracking with an Articulated 3D Body Model", Knoop et al.);

[0010] The fusion of a depth sensor and a stereo vision system, either infrared based or natural light based ("Fusion of Time-of-Flight Depth and Stereo for High Accu-

racy Depth Maps", Zhu et al., and "Improving the Kinect by Cross-Modal Stereo", Chiu et al.); and

[0011] The fusion of 4 depth sensors ("Markerless Motion Capture using multiple Color-Depth Sensors", Berger et al.);

[0012] Data fusion is also used to improve the accuracy of inertial sensors and circumvent their limitations. The most notable industrial systems featuring sensor fusion on inertial sensors include:

[0013] The Nintendo Wii Gaming Console (registered trademark) in which inertial sensors information is fused with Infrared sensors information; and

[0014] The Playstation 3 Gaming console system dubbed Playstation (registered trademarks) move featuring a RGB camera that tracks the movements of a glowing ball on a "wand" the user keeps in hand, the wand is also equipped with inertial sensors, the console system then performs the fusion.

[0015] Data fusion between RGB images and inertial sensor information has also been debated in the academia, for example in the following papers:

[0016] "Advanced tracking through efficient image processing and visual-inertial sensor fusion", Bleser et al.; and

[0017] "Visual-Inertial Sensor Fusion: Localization, Mapping and Sensor-to-Sensor Self-Calibration", Kelly et al.

[0018] In this context, there still is a need to improve the tracking of a body posture.

## SUMMARY OF THE INVENTION

[0019] According to one aspect, it is therefore provided a computer-implemented method (hereafter referred to as "tracking method") for performing body posture tracking. The tracking method comprises the step of collecting depth measurements of a body with at least one depth sensor. The tracking method also comprises collecting inertial measurements with at least one inertial sensor attached to the body. And the tracking method also comprises determining at least one posture of the body as a function of the depth measurements and the inertial measurements.

[0020] The tracking method may comprise one or more of the following:

[0021] the depth measurements and the inertial measurements constitute time-series, and determining a posture is repeated at each time the method collects depth measurements and/or inertial measurements, the method thereby tracking body motion;

[0022] determining a posture at each current time is performed based on predetermined data, on the depth measurements and/or inertial measurements collected at the current time, and on data determined at the previous time;

[0023] determining the posture comprises determining and maximizing a probability distribution that assigns probabilities to postures of the body as a function of the depth measurements and the inertial measurements;

[0024] at each current time, the probability distribution that assigns probabilities to postures of the body as a function of the depth measurements and the inertial measurements is recurrently obtained by multiplying the probability distribution that assigns probabilities to postures of the body as a function of the depth measurements and the inertial measurements determined at the

previous time, by a probability distribution that assigns probabilities to postures of the body as a function of the inertial measurements only and/or a probability distribution that assigns probabilities to postures of the body as a function of the depth measurements only, the probability distribution function of the inertial measurements only and/or the depth measurements only being obtained based on the predetermined data;

**[0025]** the method comprises an initial stage, including the steps of tracking true states of a body taking postures, collecting depth measurements of the body with said at least one depth sensor, and collecting inertial measurements with said at least one inertial sensor attached to the body; and then establishing the function that provides said at least one posture of the body for any values of depth measurements and inertial measurements;

**[0026]** the depth measurements and the inertial measurements constitute time-series and determining a posture is repeated at each time the method collects depth measurements and/or inertial measurements, and establishing the function comprises determining data based on which determining a posture at each current time is performed; and/or

**[0027]** the determined data comprise data based on which the method determines a probability distribution that assigns probabilities to postures of the body as a function of the inertial measurements only and a probability distribution that assigns probabilities to postures of the body as a function of the depth measurements only.

**[0028]** According to another aspect, it is provided a computer-implemented method (hereafter referred to as “initial stage method”) constituting an initial stage of the above method for performing body posture tracking. The initial stage method comprises the step of tracking true states of a body taking postures. The initial stage method also comprises the step of collecting depth measurements of the body with at least one depth sensor, and collecting inertial measurements with at least one inertial sensor attached to the body. And then the initial stage method comprises the step of establishing a function that provides at least one posture of the body for any values of depth measurements and inertial measurements.

**[0029]** Such initial stage method may comprise one or more of the following:

**[0030]** establishing the function comprises determining data based on which, if for the method for performing body posture tracking, the depth measurements and the inertial measurements constitute time-series and determining a posture is repeated at each time the method collects depth measurements and/or inertial measurements, determining a posture at each current time may be performed; and/or

**[0031]** the determined data comprise data based on which the method for performing body posture tracking may determine a probability distribution that assigns probabilities to postures of the body as a function of the inertial measurements only and/or a probability distribution that assigns probabilities to postures of the body as a function of the depth measurements only.

**[0032]** It is further proposed a computer program comprising instructions for performing any or both of the above methods.

**[0033]** It is further proposed a computer readable storage medium having recorded thereon the above computer program.

**[0034]** It is further proposed a system comprising a processor coupled with a memory having recorded thereon the computer program.

**[0035]** The system may further comprise at least one depth sensor and at least one at least one inertial sensor.

#### BRIEF DESCRIPTION OF THE DRAWINGS

**[0036]** Embodiments of the invention will now be described, by way of non-limiting example, and in reference to the accompanying drawings, where:

**[0037]** FIGS. 1-2 shows flowcharts of examples of the methods;

**[0038]** FIG. 3 shows an example of a system; and

**[0039]** FIGS. 4-10 illustrate the methods.

#### DETAILED DESCRIPTION OF THE INVENTION

**[0040]** FIG. 1 shows a flowchart of an example of the computer-implemented tracking method, for performing body posture tracking. The method comprises collecting S10 depth measurements of a body with at least one depth sensor. The method also comprises collecting S20 inertial measurements with at least one inertial sensor attached to the body. The method also comprises determining S30 at least one posture of the body as a function of the depth measurements and the inertial measurements. Such a method improves the tracking of a body posture.

**[0041]** Existing depth sensor based data fusion systems principally try to circumvent the problem of the sensor accuracy, disregarding almost completely the problem of the system’s blind spots, resulting in systems that are hardly robust to occlusions in the 3D scene. This is mainly due to the type of sensors that are used, since fusion is performed with field sensors that are not able to completely capture a 3D scene. On the other hand inertial sensor based data fusion systems try to improve on the performance of the existing sensors and not infer a whole body posture. In this endeavor, the basic knowledge provided by an RGB camera is sufficient. Inference on a whole body model requires more information than what is gathered by the sensors that are used in these applications.

**[0042]** On the contrary, the tracking method of FIG. 1 collects (i.e. gathers/takes) both depth and inertial measurements of the body, and uses both these specific types of measurements to estimate the body posture at S30, since the function contemplated at S30 is a function of both types of measurements (i.e. functions that never depend on one of the depth measurements or inertial measurements, e.g. for all possible values of such measurements, are thus excluded by the method). In other words, the present tracking method performs data fusion between both depth and inertial data. By performing data fusion, it is meant that the determining S30 combines both depth measurements and inertial measurements to output an estimation of the true body posture on which the measurements were collected. The use of data fusion allows to limit sensor accuracy issues (e.g. by diminishing sensing noise and/or errors). The specific data fusion used by the tracking method (i.e. fusing specifically depth and inertial data) allows to do this in a particularly interesting way, with the possibility to infer a whole body model while avoiding blind spots, all this with limited hardware (e.g. the number of sensors at use).

**[0043]** In an example, less than five depth sensors (e.g. only one or two) depth sensors are involved at S10, and/or less than ten, preferably less than five (e.g. only two, three or four) inertial sensors are involved at S20. In such an example, the method involves relatively little hardware, while providing an accurate result for most real postures taken by the body on which the measurements are collected.

**[0044]** The tracking method is advantageous on two grounds. First, the data fusion of depth measurements (e.g. depth maps) and inertial measurements (i.e. inertial sensor information) improves on the detection accuracy of the body model position, because even if both sensors bear signals with noise, the quantity of relevant information from both sensors helps them to cooperate and output a more accurate assessment of the ground truth. Second, using inertial sensors helps to deal with the intractable problem of occlusions, a problem that is encountered when part of the scene is not observed by the depth sensor. Inertial sensors on the other hand are not sensitive to occlusions but are cumbersome to place on the user. The method allows using a very limited amount of inertial sensors that will help recover information that is not accessible to the depth sensor and infer an overall accurate body model posture.

**[0045]** The method is for body posture tracking. A body is any set of articulated limbs, which may be substantially rigid for at least part of them. The bodies that may be tracked by the method may be human bodies or animal bodies. Other types of bodies, such as mechanical bodies, may be tracked as well. A posture of a body is any set of positions of its limbs, allowed by the articulations of the body. Tracking a body posture amounts to inferring the true state of the body posture (i.e. describe the body in terms of positions taken by the limbs), e.g. real-time, as widely known in the art. Indeed, the body may be in motion in a location, or scene. The location may be covered by the at least one depth sensor (i.e. the location is the collecting field of said depth sensor), thus allowing the collecting S10. The method may notably recognize the occurrence of a specific predetermined body posture type (in other words, different types of body postures may be predetermined, and the method may determine that the body has taken an instance of one of them).

**[0046]** The depth measurements and the inertial measurements may constitute time-series. In other words, the collecting S10 and the collecting S20 are each repeated over time, such that the body may take different postures due its motion, resulting in a series of postures. Such a series of postures may be called “motion” or “gesture”. The repetitions of the collecting S10 and the collecting S20 may be synchronous (i.e. simultaneous), for at least a part of the repetitions, or asynchronous, for at least a part of the repetitions. Examples are discussed later. In such a case of depth and inertial measurements constituting time-series, the determining S30 of a posture may be repeated at each time the method collects (S10, S20) depth measurements and/or inertial measurements, so as to determine the series of postures of the body. In other words, each time new depth and/or inertial measurements are collected, the method determines S30 a new corresponding posture of the body. This way, the method tracks body motion with relatively high hardware efficiency, with relatively high accuracy and avoiding possible occlusions relatively well. Such method may notably recognize the occurrence of a specific predetermined type body gesture. Gesture types may be predetermined and the method may recognize that the body has taken an instance of a type.

**[0047]** A depth sensor involved at S10 may be any known type of three-dimensional (3D) depth sensor. Such a depth sensor is a sensor exterior to (i.e. detached from) the body (the body is in free motion relative to the sensor) that provides field measurements (many measures for each time step) of the body and the surrounding scene (the “depth measurements”) that allow the determination of 3D positions of the body. The depth sensor may for example output a depth model of the body, as explained later.

**[0048]** Examples of depth sensors include:

**[0049]** Microsoft’s Kinect (registered trademark).

**[0050]** Stereo-vision Cameras (such as registered trademark PointGrey BumbleBee2);

**[0051]** Structured Light sensors (such as registered trademark Primesense Carmine);

**[0052]** Time of flight sensors (such as registered trademark Mesa Swissranger);

**[0053]** Multi camera rig (such as registered trademark Organic Motion openstage2);

**[0054]** Light Field Camera (such as registered trademark Lytro);

**[0055]** Laser telemeter;

**[0056]** Sonar;

**[0057]** Radar;

**[0058]** Lidar; and/or

**[0059]** Every monocular camera equipped with 3D reconstruction and depth map computation from structure from motion techniques.

**[0060]** Similarly, an inertial sensor involved at S20 may be any known type of inertial sensor, that provides physical measurements (the “inertial measurements”) related to the 3D position of a location attached to the body, such location depending on where on the body and how the inertial sensor is attached/linked. An inertial sensor is thus a sensor that senses physical data from which 3D positions can be deduced (the deduced positions being fixed relative to the sensor, e.g. the deduced positions being the positions of the sensor itself).

**[0061]** Examples of inertial sensors include any micro-electromechanical system combining one or more of the following sensors:

**[0062]** Gyrometer;

**[0063]** Accelerometer;

**[0064]** Magnetometer;

**[0065]** Pressure Sensor; and/or

**[0066]** MHD Sensors.

**[0067]** Typically, inertial sensors do not output 3D position measures per se but they output first (speed) and second (acceleration) derivatives. The knowledge of the initial position allows the integration to get back the new position.

**[0068]** Furthermore, physical measurements outputted by inertial sensors also depend on the movements of other body parts from the ones to which the inertial sensors are attached, as known in the art.

**[0069]** The method determines at S30 a posture of the body. In other words, the method provides an estimation of the true state, i.e. the real limb positions, of the body taking a posture. The method performs such determination at S30 as a function of the depth measurements (collected at S10) and the inertial measurements (collected at S20). In other words, the method applies a scheme (the “function”) that takes as inputs the measurements and outputs a posture (an estimation of the real posture of the body, possibly the occurrence of a predeter-

mined type, as mentioned above). Such function may itself be composed of sub-functions, as shown in the examples provided later.

**[0070]** The function may be predetermined, and the method may comprise a step of providing the function before the determining S30. Notably, the function may be predetermined in the “initial stage method”, which is a method constituting an initial stage of the tracking method. In other words, the initial stage method is performed prior to the tracking method, and constitutes a “calibration” that provides the function used at S30. The initial stage method may also be referred to as “Offline stage”, whereas the tracking method may be referred to as “Online stage”, such appellations being known in the field.

**[0071]** FIG. 2 shows a flowchart of an example of the computer-implemented initial stage method, constituting an initial stage of the method for performing body posture tracking.

**[0072]** The initial stage method comprises tracking S150 true states of a body taking postures. In other words, a body takes different postures (i.e. the body is in motion), e.g. in a location or scene that may be the same or similar (in terms of physical characteristics) to the location or scene of the tracking method. The initial stage method tracks such postures at S150 with any known tracking method providing a result with relatively high accuracy (higher than the tracking method of FIG. 1), thereby providing true states of the body. The true states are substantially the real postures taken by the body. For example, the body may take predetermined postures that are known in advance, the tracking S150 amounting to retrieve such predetermined postures. In a practical example, the tracking S150 is achieved with known methods for providing high accuracy results (involving a high number of depth sensors relative to the method of FIG. 1, for example strictly more than five, and/or involving a high number of inertial sensors relative to the method of FIG. 1, for example strictly more than ten). The initial stage method is typically performed once, possibly at the manufacturer side or at an initial calibration phase of any system implementing the method, while the tracking method based on the function established by said initial stage method may be performed any number of times (as long as the material still functions). For this reason, the costs involved by the high accuracy required by the tracking S150 are not an issue.

**[0073]** In parallel to the tracking S150, the initial stage method comprises collecting S100 depth measurements of the body with at least one depth sensor, and collecting S200 inertial measurements with at least one inertial sensor attached to the body. Any depth sensor involved at S100 and/or inertial sensor involved at S200 may be the same or similar (i.e. of the same type, e.g. the same model) as the sensors involved in the tracking method at S10 and S20. Also, the collecting S100 and/or S200 are performed in the same way or similarly to the collecting S10 and/or S20. The idea is that the initial stage method compares what these sensors output to the true states (that are known thanks to S150), in order to establish the function that relates both data. Thus, the (experimental) conditions of the initial stage method may be the same or similar to the (real) conditions of the tracking method. These machine learning concepts are known from the field of data fusion and are thus not further discussed.

**[0074]** The initial stage method then comprises establishing S300 a function that provides at least one posture of the body for any values of depth measurements and inertial measurements. Such a function may then be used in the tracking

method, and for any values collected at S10 and/or S20, the tracking method will be able to perform the determining S30. Examples of the establishing S300 are provided later.

**[0075]** In the case where, in the tracking method, the depth measurements and the inertial measurements constitute time-series and determining S30 a posture is repeated at each time the method collects (S10, S20) depth measurements and/or inertial measurements, establishing S300 the function correspondingly comprises determining data based on which determining a posture at each current time may be performed. The initial stage is indeed associated to the tracking method, such that any example of the tracking method is feasible.

**[0076]** In the specific case that time-series are considered, the initial stage may consider this specificity and the function may actually depend on the current time. This way, time information is used and the tracking is improved. Indeed, the determining S30 at each current time (noted e.g.  $t+1$ ) may be performed based on predetermined data (e.g. established at S300 of the initial stage), on the depth measurements and/or inertial measurements collected at the current time (depth measurements are collected at repetition of S10 at  $t+1$  and/or inertial measurements are collected at repetition of S10 at  $t+1$ ), and on data determined at the previous time ( $t$ ) (e.g. data determined from depth and/or inertial measurements not later than the previous time  $t$ , and/or data determined from a repetition of determining S30 not later than the previous time  $t$ ). This allows real-time body tracking, in an improved way. Examples are provided later.

**[0077]** The methods are computer-implemented. This means that the methods are executed on at least one computer, or any system alike. Thus, steps of the methods are performed by the computer, possibly fully automatically, or, semi-automatically (e.g. steps may be triggered by the user). A typical example of computer-implementation of the method is to perform any or both of the methods with a system suitable for this purpose. The system may comprise a memory having recorded thereon instructions for performing any or both of the methods. In other words, software is already ready on the memory for immediate use. The system is thus suitable for performing any or both of the methods without installing any other software. Such a system may also comprise at least one processor coupled with the memory for executing the instructions. In other words, the system comprises instructions coded on a memory coupled to the processor, the instructions providing means for performing any or both of the methods. Such a system is an efficient tool for tracking body postures.

**[0078]** To cause the system to perform any or both of the methods, it is provided a computer program comprising instructions for execution by a computer, the instructions comprising means for this purpose. The program may be recorded on a data storage medium. The program may for example be implemented in digital electronic circuitry, or in computer hardware, firmware, software, or in combinations of them. Apparatus of the invention may be implemented in a computer program product tangibly embodied in a machine-readable storage device for execution by a programmable processor; and method steps of the invention may be performed by a programmable processor executing a program of instructions to perform functions of the invention by operating on input data and generating output. The instructions may advantageously be implemented in one or more computer programs that are executable on a programmable system including at least one programmable processor coupled to receive data and instructions from, and to transmit data and

instructions to, a data storage system, at least one input device, and at least one output device. The application program may be implemented in a high-level procedural or object-oriented programming language, or in assembly or machine language if desired; and in any case, the language may be a compiled or interpreted language. The program may be a full installation program, or an update program. In the latter case, the program updates an existing system to a state wherein the system is suitable for performing the method.

**[0079]** FIG. 3 shows an example of a system adapted for performing any or both of the initial stage and tracking methods.

**[0080]** The system of the example comprises a central processing unit (CPU) **1010** connected to an internal communication BUS **1000**, a random access memory (RAM) **1070** also connected to the BUS. The system is further provided with a graphics processing unit (GPU) **1110** which is associated with a video random access memory **1100** connected to the BUS. Video RAM **1100** is also known in the art as frame buffer. A mass storage device controller **1020** manages accesses to a mass memory device, such as hard drive **1030**. Mass memory devices suitable for tangibly embodying computer program instructions and data include all forms of non-volatile memory, including by way of example semiconductor memory devices, such as EPROM, EEPROM, and flash memory devices; magnetic disks such as internal hard disks and removable disks, magneto-optical disks, and CD-ROM disks **1040**. Any of the foregoing may be supplemented by, or incorporated in, specially designed ASICs (application-specific integrated circuits). A network adapter **1050** manages accesses to a network **1060**. The system may also include a haptic device **1090** such as a cursor control device, a keyboard or the like. A cursor control device is used in the system to permit the user to selectively position a cursor at any desired location on screen **1080**, as mentioned with reference to FIG. 2. By screen, it is meant any support on which displaying may be performed, such as a computer monitor. In addition, the cursor control device allows the user to select various commands, and input control signals. The cursor control device includes a number of signal generation devices for input control signals to system. Typically, a cursor control device may be a mouse, the button of the mouse being used to generate the signals.

**[0081]** The system further comprises ports **1200** that may allow the connection of at least one depth sensor and/or at least one inertial sensor, involved in the tracking method and/or the initial stage method. Alternatively or additionally, the system may comprise a wireless station for wireless communication with said sensors. The system may in any case comprise any known required hardware for performing collecting steps (S10, S20, S100, S200) or the tracking **5150** of any or both the methods.

**[0082]** The tracking method may be used in a variety of applications ranging from video gaming to surgical assistance. In most of these applications, “gesture/motion recognition” and “tracking” are useful. The tracking method may thus enhance such technology fields by improving body posture or motion tracking.

**[0083]** Examples of the tracking method and of the initial stage method are now discussed.

**[0084]** The methods may collect the depth measurements at S10 and/or S100 and organize them as a depth map (i.e. a 3D map of depth measurements). Then, before the determining S30 or the establishing S300, the methods may process the

depth map in such a way that the body limbs are detected and fitted to a comprehensive model. Body motion/posture tracking is in the prior art typically performed through time, and through detection and/or prediction, the tracking method may update the position of a limb to its actual spatial position. Tracking is advantageous for a smooth experience since it enforces a correct match between body limbs and also ensures that each body part remains detected in each and every frame.

**[0085]** A body model may be used by the methods of the examples to translate the recognized posture or gestures in an integrated fashion. Specifically, the methods may translate the depth measurements to compute parameters of a body model. Examples of body models include wireframe skeletons, such as skeleton **40** of FIG. 4, or cylindrical body shapes such as cylindrical body shape **50** of FIG. 5. These models are parameterized (e.g. each wire position in the wireframe skeleton or the cylinders’ lengths and widths in the cylindrical model). Some of the parameters may be given by the user, but most or all of them may be directly brought by the tracking method. The completed body model can then be used to animate an avatar or any representation of the body for representation on a display device, such as avatar **104** on display device **102** exemplified on FIG. 8. The exact management of body models is however out of the scope of the present discussion, as this is known from the prior art.

**[0086]** There exist many solutions that aim at extracting a body model from a depth map and that may be implemented by the methods, following S10 and/or S100. Reviewing them would be out of the scope of this discussion. A known example is to use visual cues and depth information to generate a hypothetical body model. This model is then confronted to a set of constraints as well as its previous state to infer the actual body model position.

**[0087]** However sophisticated the body model tracking is, tracking artifacts always subsist. These artifacts are due to the inherent inaccuracy of the depth motion sensors as well as a purely physical concern: since the depth sensor analyzes the scene from a vantage point, all blind spots from this vantage point are also blind spots in the depth sensor perception, resulting in incomplete scene detection. The situations where there are blind spots in the scene are also called occlusions.

**[0088]** In FIGS. 6-7, two such situations are depicted in which only self-occlusions are shown. On the right of each figure can be seen a schematic representation of the actual user position **62**, and on the left is the resulting body model **64** (a skeleton view is used as a practical example). In the first case (FIG. 6) the user is only crossing his legs, and this situation is problematic to most body model solvers and usually dealt with using additional constraints or model priors, which may be cumbersome. However, there is no amount of additional constraint or model prior that can help solve the problem encountered in FIG. 7. FIG. 7 shows the case where the user only shows his side to the depth sensor, in this case, half his body is occluded, this type of occlusion is simply intractable to a depth sensing unit alone. In this case, the body model ends up in a garbled state in which the solver mismatches the occluded body parts with parts present in the scene.

**[0089]** The tracking method of the examples brings a solution to this technical problem as it allows the creation of a body model that is free of occlusions.

**[0090]** Recently, inertial sensors have gained a lot of attention, notably through their use in video game consoles. Unfor-

tunately, recovering a whole body model from inertial sensors requires a large amount of them (between 10 and 20 inertial sensors placed in different locations on the body). This large amount is prohibitive for most applications since recovering the body model would require the user to be outfitted with these sensors every time.

**[0091]** As a viable alternative, the tracking method proposes to make use of both worlds. Indeed using a limited amount of body attached inertial sensors cannot provide the system with a full body posture but instead only a set of probable body postures at best. However the complementary usage of a depth sensor brings back the constraints lost by removing most body attached inertial sensors. Wearing few, for example two or three, sensors is minimally intrusive for the user, but at the same time reduces mismatches due to occlusions in the depth map, in a surprising extent.

**[0092]** As depicted in FIG. 8, the system of the examples may consist in a depth sensor **82** and a small amount of wearable inertial sensors (two: **84** and **86**) attached to the user's body **88**. The purpose of the methods is to accurately track the person's body movement through time and apply it to a computer generated body model, that is then shown on a display device **102** in the example.

**[0093]** Several technical terms used in the following are now defined and discussed.

**[0094]** A state or body state is a description of the motion or the absence of motion of the body. A true state is a description that matches with high accuracy the real position of the body. A true body state is a time evolving body model that measures with the highest possible precision the real position such that the true state and the real position can be virtually identical. For example, the highest distance error between a 3D position provided for any point of any limb of the body and the real position of said point is lower than a predetermined threshold, e.g. equal to a percentage of the height of the body, e.g. 1%, 0.1% or 0.01%.

**[0095]** A body model is a parametric representation of the body. As discussed earlier, several types of body models can be considered: a wireframe skeleton or a cylindrical model are just two examples. Mathematically a body model can be fully represented by a vector of the model parameters (e.g. nodes positions and limbs length in the wireframe skeleton case).

**[0096]** The depth model is the set of depth measurements (represented mathematically as a vector of measurements for each and every body part) acquired at **S10** or **S100** with the depth sensor(s) that incompletely represents the body state. The following examples use the depth model to infer the true body state. The depth model in itself can be used to create a body model but could also be a body model (in which case the depth model is represented mathematically by a vector of parameter measurements for each and every part of the body model). The term of depth model is used here for more generality. The examples use a body model extracted from the depth measurements as a depth model.

**[0097]** The inertial model is the set of inertial measurements (represented mathematically as a vector of measurements for each and every sensor location) acquired at **S20** or **S200** with the inertial sensor(s) that incompletely represents the body state. The following examples use the inertial model to infer the true body state. The inertial model in itself can be used to create a body model but could also be a body model (in which case the inertial model is represented mathematically by a vector of body model parameter measurements collected

on sensors placed at fixed and predefined places on the body). The term of inertial model is used here for more generality. The examples use the inertial sensor measurements as an inertial model.

**[0098]** The tracking method is based on the observations that both the depth sensor and the motion sensors only give an incomplete assessment of the body parts position, but this incomplete information can be fused to reconstitute an accurate representation of the body motion.

**[0099]** As explained earlier, the methods for in fine performing the tracking may be separated in two stages.

**[0100]** The first stage, denoted as initial or offline stage, corresponds to the stage where all the parameters of the algorithm are evaluated, and this is done once and for all. The term offline refers to the fact that this stage is transparent to the user of the tracking method and even if it has large computational needs, time to do computations can be taken. This stage in the following examples relies heavily on machine learning algorithms and especially on regression machine learning algorithms.

**[0101]** The second stage, denoted as online stage or tracking method, gathers all the steps of the method that are done in real-time during the body model tracking. This stage in the following examples decomposes in three technologies:

**[0102]** Extraction through time of the depth model and inference of the probability of the true body state.

**[0103]** Extraction through time of the inertial model and inference of the probability of the true body state.

**[0104]** Possibly asynchronous sensor fusion of both true body state probabilities in order to augment both the accuracy of the body model and the completeness of the model.

**[0105]** FIG. 10 illustrates the technological workflow and is described later.

**[0106]** An example of the offline initial stage is now discussed with reference to FIG. 9, which presents an example of a flowchart representing actions included in the establishing **S300**. This example of the initial stage corresponds to an online tracking method performing body motion tracking as explained earlier, i.e. wherein the depth measurements and the inertial measurements constitute time-series, and determining a posture is repeated at each time.

**[0107]** Furthermore, the tracking method corresponding to this example further presents the specificity that determining a posture at **S30** at each current time is performed based on predetermined data, on the depth measurements and/or inertial measurements collected at the current time, and on data determined at the previous time, namely by determining and maximizing a probability distribution that assigns probabilities to postures of the body as a function of the depth measurements and the inertial measurements. In other words, the method determines a probability distribution that assigns (i.e. provides) a probability that the body took a specific posture (for all postures) when collecting at **S10** and **S20** the depth and inertial measurements, given the values of said measurements. Then the method may simply maximize (i.e. determine the maximum or the maximum's argument) such probability distribution, for example by outputting the argument of the maximum of the probability distribution. Said argument is considered to be the most probable posture in the example.

**[0108]** In such a case, the predetermined data, based on which determining the posture is performed at **S30** and that the initial stage determines at **S300**, may comprise data based

on which the method for performing body posture tracking may determine a probability distribution that assigns probabilities to postures of the body as a function of the inertial measurements only and/or a probability distribution that assigns probabilities to postures of the body as a function of the depth measurements only. In other words, two probability distributions are contemplated: both assign probabilities to postures based on only one of the two types of data collected (depth or inertial). This keeps computational costs low, as illustrated in the following.

**[0109]** In order to infer the true body state given a set of incomplete and/or inaccurate measurements, the tracking method of the example first computes the probability distribution of the true body state being in state A given that the depth or inertial model is in state A'. Computing the probability distribution of the true body state given either the depth or inertial model is done with a classical Machine Learning algorithm for regression (step S930 on FIG. 6), that may implement any or a combination of:

**[0110]** Mixture of Experts for Regression (e.g. as described in "Adaptive mixtures of local experts", Jacobs et al.);

**[0111]** Boosted Regression (e.g. as described in "Greedy Function Approximation: A Gradient Boosting Machine", Friedman et al.);

**[0112]** Kernel Ridge Regression (e.g. as described in "Learning with Kernels", Smola et al.);

**[0113]** Markov Chain Monte Carlo (e.g. as described in "Bayesian Data Analysis", Gelman et al.); or

**[0114]** Approximate Bayesian Computation (e.g. as described in "Bayesian Data Analysis", Gelman et al.).

**[0115]** Machine Learning algorithms use a training dataset that is constructed as explained in the following.

**[0116]** Let us consider  $y$  as a true body state observation of the body.  $y$  is characterized as a vector of input (provided at step S920 on FIG. 9, and collected at S150) in space Y. Concurrently let us consider  $m$  as either the inertial or depth model vector (provided at step S920 on FIG. 9 and collected at S100 or S200), in space M. Each  $m$  and  $y$  are acquired at the same time so that they are perfectly matching. The training data set consists in a collection of N pairs  $m$  and  $y$ :  $D = \{(m, y), m \in M, y \in Y, i \in [1, N]\}$  (provided at step S900 on FIG. 9), using this data set we are interested in the construction of the conditional probability  $P(Y=y|M=m)$ . In all practicality N may be fairly large to be able to cover most adequately all dimensions of the problem. N may for example be as large as  $k$  times the number of dimensions in Y times (i.e. multiplying) the number of dimensions in  $m$ , with  $k$  superior to 100 or to 500, and/or inferior to 2000 or to 1500, preferably equal to 1000.

**[0117]** The creation of the training data set D is straightforward. The true body state observations  $y$  can be acquired at S150 using any known motion capture paradigm as explained earlier. An example includes the algorithm described in the paper entitled "A survey of computer vision-based human motion capture", by Moeslund et al. The body can also be fitted with many inertial sensors, much more than what is required for the tracking method, to be able to capture the movement of each body part. This may be done as described in the paper entitled "A Real-Time Articulated Human Motion Tracking Using Tri-Axis Inertial/Magnetic Sensors Package", by Zhu et al.

**[0118]** The corresponding  $m$  model vectors are acquired at the same time and are the same as the ones that would be

acquired during the online phase (after learning, during usage), and correspond to the depth model in one case and to the inertial model in the other.

**[0119]** Then, given a new model vector  $m_{new}$ , we get the distribution of possible true body states as

$$P(y | m_{new}) \\ \forall y \in Y$$

**[0120]** Known regression algorithms (step S930 on FIG. 9) are often parametric and model the conditional probability profile  $P(Y=y|M=m)$  using a mixture of nonlinear functions. The learning stage of the algorithm may consist in finding the parameters of the model.

**[0121]** Let us consider linear regression as an example of an algorithm that can be applied at S930, such as the algorithm described in the reference entitled "Bayesian Inference in Statistical Analysis", Box., or "Murphy, Kevin P. *Machine learning: a probabilistic perspective*. The MIT Press, 2012".

**[0122]** Each ground truth observation  $y$  is modeled as a linear expression of the measurement  $m$  as:

**[0123]**  $y_i = Am_i + \epsilon$  where A is a matrix of size the dimension of M times the dimension of Y, and  $\epsilon$  is an independent and identical normally distributed random noise:  $\epsilon_i \sim N(0, \Sigma)$  where  $\Sigma$  is the noise covariance matrix. Using this last equation, the conditional probability of  $y$  given  $m$  can be written:

$$P(y | m) = N(y | Am, \sigma) = \frac{1}{\det(\Sigma)\sqrt{2\pi}} \exp\left(-\frac{1}{2}(y - Am)^T \Sigma^{-1} (y - Am)\right)$$

**[0124]** A and E can be computed analytically maximizing the fit on D, we get:

$$A = (\underline{m}^T \underline{m})^{-1} \underline{m}^T \underline{y} \\ \Sigma = \text{cov}(y - Am)$$

**[0125]** where  $\underline{y}$  and  $\underline{m}$  are the matrices collecting all the training samples.

**[0126]** We see here that with the knowledge of D the method can completely define the probability  $P(Y=y|M=m)$ .

**[0127]** For a full profile of the conditional probability, the initial stage may preferably use (instead of linear regression) non-parametric methods such as MCMC (Markov Chain Monte Carlo) or ABC (Approximate Bayesian Computation).

**[0128]** An example of the online tracking stage corresponding to the initial stage of FIG. 9 is now discussed with reference to FIG. 10.

**[0129]** As explained earlier, in this example the depth measurements and the inertial measurements constitute time-series, and determining a posture is repeated at each time the method collects depth measurements and/or inertial measurements. Moreover, determining a posture at each current time is performed based on predetermined data (during the initial stage), on the depth measurements and/or inertial measurements collected at the current time, and on data determined at the previous time. Furthermore, determining the posture comprises determining and maximizing a probability distribution that assigns probabilities to postures of the body as a function of the depth measurements and the inertial measurements.

**[0130]** In this example, at each current time (noted t+1), the probability distribution  $P(y_{t+1}|X_{t+1}, Z_{t+1})$  that assigns probabilities to postures  $y$  of the body as a function of the depth measurements and the inertial measurements is recurrently obtained (i.e. a recurrence is predefined to determine the probability distribution as the determining S30 is repeated) by multiplying the probability distribution  $P(y_t|X_t, Z_t)$  that assigns probabilities to postures  $y$  of the body as a function of the depth measurements and the inertial measurements determined at the previous time (noted t), by the probability distribution  $P_{inertial}(y_{t+1}|x_{t+1})$  (e.g. determined during the initial stage) that assigns probabilities to postures  $y$  of the body as a function of the inertial measurements only; and/or by the probability distribution  $P_{depth}(y_{t+1}|z_{t+1})$  (e.g. determined during the initial stage) that assigns probabilities to postures  $y$  of the body as a function of the depth measurements only. As explained earlier, the probability distribution function of the inertial measurements only and/or the depth measurements only may be obtained based on the predetermined data (e.g. determined during the initial stage). This is detailed below.

**[0131]** The method may comprise after collecting S10 depth measurements the extraction of a body model from a depth map, as discussed earlier. This corresponds to step 1.6 on FIG. 10.

**[0132]** Technologies allowing the extraction of a body model from a depth map are out of the scope of the present discussion. Current technologies track specific landmarks places on the body joints and recreate a wireframe skeleton of the body or track body parts position and orientations to recreate a cylinder based body model. An adequate method for body model extraction can be found in the paper entitled “Real-Time Human Pose Recognition in Parts from Single Depth Images”, by Shotton et al.

**[0133]** We denote the depth model vector as  $z$  (provided at step 1.8 of FIG. 10), in the space of depth model vector observations  $Z$  and  $y$  as a true body state observation of the body.  $y$  is characterized as a vector of input (provided at step 1.9 of FIG. 10) in space  $Y$ . The conditional probability profile  $P_{depth}(y|Z=z)$  is estimated using the procedure described in the discussion regarding the offline initial stage (step 1.10 of FIG. 10). The obtained distribution is used in conjunction with the inertial sensor conditional probability distribution to extract the most probable body state.

**[0134]** Given a new and never before seen depth model vector  $z$ , the most probable body model state that could be inferred by the depth sensor is obtained by maximization of the probability:

$$y = \underset{y}{\operatorname{argmax}} P_{depth}(Y = y | Z = z).$$

This is step 1.7. We are however interested in the whole probability profile expressed as a function of  $y$ :  $f_{depth}(y) = P_{depth}(Y=y|Z=z)$ .

**[0135]** The method may comprise after collecting S20 inertial measurements the extraction of a body model from a few inertial sensors placed onto the body, as discussed earlier. This corresponds to step 1.4 on FIG. 10.

**[0136]** The inertial sensors output a signal time series that bears some information about the state of the body part it is mounted on but also subtle variations associated with the movement of other body parts. The inertial sensor’s signal may be highly corrupted by noise, a preprocessing step involving signal filtering and signal processing being thereby

advantageous. Describing these known technologies is outside the scope of this discussion. Examples of such technologies that work can be for instance found in the paper entitled “A Real-Time Articulated Human Motion Tracking Using Tri-Axis Inertial/Magnetic Sensors Package”, by Zhu et al.) and the references therein.

**[0137]** As before, let us consider  $y$  as a true body state observation of the body.  $y$  is characterized as a vector of input (step 1.3) in space  $Y$ . Concurrently let us consider  $x$  as the inertial model vector (step 1.1), in space  $X$ . The conditional probability profile  $P_{inertial}(y|X=x)$  is estimated using the procedure (step 1.2) described in the Machine Learning Technology section.

**[0138]** Given a new and never before seen inertial model vector  $x$ , the most probable body model state that could be inferred with the inertial sensor is obtained by maximization of the probability:

$$y = \underset{y}{\operatorname{argmax}} P_{inertial}(Y = y | X = x).$$

We are however interested in the whole probability profile expressed as a function of  $y$ :  $f_{inertial}(y) = P_{inertial}(Y=y|X=x)$ .

**[0139]** Now, the method may implement synchronous or asynchronous data fusion at S30 (such general types of data fusion being discussed in the paper entitled “Multi Sensor Data Fusion”, by Durrant-Whyte) of the depth model and the inertial model.

**[0140]** Let us consider a time governed process. For each time step  $t$  we want to know the value  $y$  that maximizes the probability  $P(y_{t+1}|X_{t+1}, Z_{t+1})$  where  $X_{t+1}$  is  $\{x_{t+1}, x_t, x_{t-1}, \dots, x_0\}$  and  $Z_{t+1}$   $\{z_{t+1}, z_t, z_{t-1}, \dots, z_0\}$  is the set of depth and inertial models acquired so far. This maximizing value  $y^*$  is evaluated at time t+1 and corresponds to the (t+1)<sup>th</sup> value of  $y$ . In order to get the function of  $y$ ,  $P(y_{t+1}|X_{t+1}, Z_{t+1})$ , the method uses data fusion techniques that will correlate this quantity with  $P_{depth}(y_{t+1}|Z=Z_{t+1})$  and  $P_{inertial}(y_{t+1}|X=X_{t+1})$ .

**[0141]** Using Bayes rule (such as described in the paper entitled “Bayesian Data Analysis”, by Gelman et al.), we can rewrite  $P(y|X_{t+1}, Z_{t+1})$  as:

$$\begin{aligned} P(y_{t+1} | X_{t+1}, Z_{t+1}) &= P(y_{t+1} | x_{t+1}, z_{t+1}, X_t, Z_t) \\ &= \frac{P(x_{t+1}, z_{t+1} | y_{t+1}, X_t, Z_t) P(y_{t+1} | X_t, Z_t)}{P(y_{t+1} | X_t, Z_t)} \end{aligned}$$

**[0142]** using the independence of the measurement of different sensors given the  $y$ , we can rewrite this as

$$P(y|X_{t+1}, Z_{t+1}) \propto P(x_{t+1}, z_{t+1} | y_{t+1}, X_t, Z_t) P(y|X_t, Z_t) \propto P(x_{t+1} | y_{t+1}, X_t) P(z_{t+1} | y_{t+1}, Z_t) P(y_{t+1} | X_t, Z_t)$$

**[0143]** where  $\propto$  is the proportionality symbol removing the terms independent of  $y$ .

**[0144]** Assuming that measurement noise is uncorrelated with time, we have)

$$P(x_{t+1} | y_{t+1}, Z_t) = P(x_{t+1} | y_{t+1})$$

**[0145]** using Bayes rule again:

$$P(z_{t+1} | y_{t+1}) \propto \frac{P(y_{t+1} | z_{t+1})}{P(y_{t+1})}$$

[0146] P(y) can be inferred through a probabilistic or physical model of the true body state, considering the high dimensionality of the state space we used a uniform density for P(y).

[0147] This final model leads to:

$$P(y_{t+1}|X_t, Z_t) = \int P_{depth}(y_{t+1}|z_{t+1}) P_{inertial}(y_{t+1}|x_{t+1}) P(y_t|X_t, Z_t)$$

[0148] Marginalization of the joint probability yields:

$$P(y_{t+1}|X_t, Z_t) = \int dy_t P(y_{t+1}|y_t, X_t, Z_t) P(y_t|X_t, Z_t).$$

[0149] However the true state is not dependent on the observations so we have

$$P(y_{t+1}|y_t, X_t, Z_t) = P(y_{t+1}|y_t).$$

[0150] This last quantity (the state transition probability) can be obtained through the use of a physical model of the true body state or through extensive learning in the parameter space. An example of a physical body model may be found in the paper entitled “Spacetime Constraints”, by Witkin et al.

[0151] The method uses a Dirac (centered in  $y_t$ ) distribution for the state transition probability, leaving out the state prediction to the sensors only. It can be noted however that using the Dirac distribution is part of the general model where we could have used a physical model of the true body state, namely the case of the resting body model.

[0152] We get the recurrence used by the method as mentioned above:

$$P(y_0|X_0, Z_0) = P(y_0|x_0, z_0) = P_0$$

$$P(y_{t+1}|X_{t+1}, Z_{t+1}) \propto P_{depth}(y_{t+1}|z_{t+1}) P_{inertial}(y_{t+1}|x_{t+1}) P(y_t|X_t, Z_t)$$

[0153]  $P_0$  may be initialized either with a normal law with identity variance and zero mean or a prior body model P(y) computed beforehand on the ground truth data with Monte Carlo Sampling.

[0154] Even if normalization of  $P(y_t|X_t, Z_t)$  is not required, it may be conducted at regular intervals by the method to prevent the quantity to reach virtually zero in machine precision.

[0155] As illustrated by step 1.11, in the case of asynchronous data streams, techniques such as track-to-track filtering may be used to process the data stream as the data comes. Track-to-track filtering updates the probabilities of the body model motion as each of the probabilities  $P_{depth}(y|Z=z)$  and  $P_{inertial}(y|X=x)$  are given, which in turn relates to the sensors frequency.

[0156] The recursion then becomes:

$$P(y_0|X_0, Z_0) = P_0$$

$$P(y_{t+1}|X_{t+1}, Z_{t+1}) \propto P_{inertial}(y_{t+1}|x_{t+1}) P(y_t|X_t, Z_t)$$

or

$$P(y_{t+1}|X_{t+1}, Z_{t+1}) \propto P_{depth}(y_{t+1}|z_{t+1}) P(y_t|X_t, Z_t).$$

[0157] Once the probability of the body model motion given each sensor output is obtained, the most probable body model motion is obtained through maximization of the probability:

$$y_{t+1}^* = \underset{y_{t+1}}{\operatorname{argmax}} P(y_{t+1} | x_{t+1}, z_{t+1}).$$

[0158] While this invention has been particularly shown and described with references to example embodiments

thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the scope of the invention encompassed by the appended claims.

1. A computer-implemented method for performing body posture tracking, comprising the steps of:

collecting (S10) depth measurements ( $Z_{t+1}$ ) of a body with at least one depth sensor;

collecting (S20) inertial measurements ( $X_{t+1}$ ) with at least one inertial sensor attached to the body; and

determining (S30) at least one posture ( $y^*$ ) of the body as a function of the depth measurements and the inertial measurements.

2. The method of claim 1, wherein the depth measurements and the inertial measurements constitute time-series, and determining a posture is repeated at each time the method collects depth measurements and/or inertial measurements, the method thereby tracking body motion.

3. The method of claim 2, wherein determining a posture at each current time (t+1) is performed based on predetermined data, on the depth measurements ( $z_{t+1}$ ) and/or inertial measurements ( $x_{t+1}$ ) collected at the current time, and on data determined at the previous time.

4. The method of claim 3, wherein determining the posture comprises determining and maximizing a probability distribution ( $P(y_{t+1}|X_{t+1}, Z_{t+1})$ ) that assigns probabilities to postures (y) of the body as a function of the depth measurements and the inertial measurements.

5. The method of claim 4 wherein, at each current time (t+1), the probability distribution ( $P(y_{t+1}|X_{t+1}, Z_{t+1})$ ) that assigns probabilities to postures (y) of the body as a function of the depth measurements and the inertial measurements is recurrently obtained by multiplying the probability distribution ( $P(y_t|X_t, Z_t)$ ) that assigns probabilities to postures (y) of the body as a function of the depth measurements and the inertial measurements determined at the previous time (t), by a probability distribution ( $P_{inertial}(y_{t+1}|x_{t+1})$ ) that assigns probabilities to postures (y) of the body as a function of the inertial measurements only and/or a probability distribution ( $P_{depth}(y_{t+1}|z_{t+1})$ ) that assigns probabilities to postures (y) of the body as a function of the depth measurements only, the probability distribution function of the inertial measurements only and/or the depth measurements only being obtained based on the predetermined data.

6. The method of claim 1, wherein the method comprises an initial stage, including the steps of:

tracking (S150) true states (y) of a body taking postures;

collecting (S100) depth measurements (z) of the body with said at least one depth sensor, and collecting (S200)

inertial measurements (x) with said at least one inertial sensor attached to the body; and then establishing (S300) the function that provides said at least one posture ( $y^*$ ) of the body for any values of depth measurements (z) and inertial measurements (x).

7. The method of claim 6, wherein the depth measurements and the inertial measurements constitute time-series and determining a posture is repeated at each time the method collects depth measurements and/or inertial measurements, and establishing the function comprises determining data based on which determining a posture at each current time (t+1) is performed.

8. The method of claim 7, wherein the determined data comprise data based on which the method determines a probability distribution ( $P_{inertial}(y_{t+1}|X_{t+1})$ ) that assigns probabili-

ties to postures (y) of the body as a function of the inertial measurements only and a probability distribution ( $P_{depth}(y_{t+1}|z_{t+1})$ ) that assigns probabilities to postures (y) of the body as a function of the depth measurements only.

9. A computer-implemented method constituting an initial stage of a method for performing body posture tracking, comprising the steps of:

- tracking true states (y) of a body taking postures;
- collecting depth measurements (z) of the body with at least one depth sensor, and collecting inertial measurements (x) with at least one inertial sensor attached to the body; and then
- establishing a function that provides at least one posture (y\*) of the body for any values of depth measurements (z) and inertial measurements (x).

10. The method of claim 9, wherein establishing the function comprises determining data based on which, if for the method for performing body posture tracking, the depth measurements and the inertial measurements constitute time-series and determining a posture is repeated at each time the method collects depth measurements and/or inertial measurements, determining a posture at each current time (t+1) may be performed.

11. The method of claim 10, wherein the determined data comprise data based on which the method for performing body posture tracking may determine a probability distribution ( $P_{inertial}(y_{t+1}|x_{t+1})$ ) that assigns probabilities to postures (y) of the body as a function of the inertial measurements only and/or a probability distribution ( $P_{depth}(y_{t+1}|z_{t+1})$ ) that assigns probabilities to postures (y) of the body as a function of the depth measurements only.

- 12. A computer program product comprising:
  - a data storage medium embodying code instructions for performing body posture tracking; and
  - the code instructions including instructions configuring a computer to:
    - collect (S10) depth measurements ( $Z_{t+1}$ ) of a body with at least one depth sensor;
    - collect (S20) inertial measurements ( $X_{t+1}$ ) with at least one inertial sensor attached to the body; and
    - determine (S30) at least one posture (y\*) of the body as a function of the depth measurements and the inertial measurements.

- 13. A system comprising:
  - a processor; and
  - a memory operatively coupled to the processor and having recorded thereon program code, wherein the processor executing the program code:
    - collects (S10) depth measurements ( $Z_{t+1}$ ) of a body with at least one depth sensor;

collects (S20) inertial measurements ( $X_{t+1}$ ) with at least one inertial sensor attached to the body; and

determines (S30) at least one posture (y\*) of the body as a function of the depth measurements and the inertial measurements.

14. The system of claim 13, further comprising at least one depth sensor and at least one inertial sensor.

15. The system of claim 13, wherein the depth measurements and the inertial measurements constitute time-series, and the processor determines a posture by repeating at each time that the processor collects depth measurements and/or inertial measurements, the system thereby tracking body motion.

16. The system of claim 15, wherein the processor determines a posture at each current time (t+1) based on predetermined data, on the depth measurements ( $Z_{t+1}$ ) and/or inertial measurements ( $x_{t+1}$ ) collected at the current time, and on data determined at the previous time.

17. The system of claim 16, wherein the processor determines the posture by: determining and maximizing a probability distribution ( $P(y_{t+1}|X_{t+1}, Z_{t+1})$ ) that assigns probabilities to postures (y) of the body as a function of the depth measurements and the inertial measurements.

18. The system of claim 13, wherein the processor executing the program code further performs an initial stage to:

- track (S150) true states (y) of a body taking postures;
- collect (S100) depth measurements (z) of the body with said at least one depth sensor, and collect (S200) inertial measurements (x) with said at least one inertial sensor attached to the body; and then
- establish (S300) the function that provides said at least one posture (y\*) of the body for any values of depth measurements (z) and inertial measurements (x).

19. The system of claim 18, wherein the depth measurements and the inertial measurements constitute time-series and determining a posture is repeated at each time the method collects depth measurements and/or inertial measurements, and establishing the function comprises determining data based on which determining a posture at each current time (t+1) is performed.

20. The system of claim 19, wherein the determined data comprise data based on which the method determines a probability distribution ( $P_{inertial}(y_{t+1}|x_{t+1})$ ) that assigns probabilities to postures (y) of the body as a function of the inertial measurements only and a probability distribution ( $P_{depth}(y_{t+1}|z_{t+1})$ ) that assigns probabilities to postures (y) of the body as a function of the depth measurements only.

\* \* \* \* \*