

GAIT MONITORING METHOD AND ROBOT

The present application claims priority to the Singapore patent application no. 10202103579T which is incorporated herein in its entirety by reference.

TECHNICAL FIELD

[0001] The present disclosure relates generally to the field of rehabilitation technologies, and more particularly to a gait monitoring method and robot.

BACKGROUND

[0002] For many patients and the elderly, mobility impairment is a major factor contributing to rapid deterioration of overall health and well-being. Data gathered on the way a patient walks can be useful in many rehabilitative or general healthcare applications. In order for the healthcare professional to gather data, for example to assess a patient's recovery from a loss of locomotion, a marker-based motion capture system may be used. Conventionally, multiple sensors are attached to the patient's body or clothing to serve as markers. A typical hardware setup involves multiple cameras to record the patient's movements on a treadmill or on a defined straight track in a gait analysis or kinematics laboratory.

SUMMARY

[0003] In one aspect, the present disclosure provides a marker-less gait monitoring method for use with a subject, comprising: acquiring a point cloud of data points based on a depth image of at least one of the subject's lower limbs; allocating selected ones of the data points to one of a foot model and a shank model; and determining one or more pose parameters of the foot model based on optimizing parameters of a foot-shank model, wherein the foot-shank model includes the foot model and the shank model as adjoining parts that are rotatable relative to one another about a common origin.

[0004] The depth image comprises a back view image of the subject. The marker-less gait monitoring method may further comprise: acquiring the depth image, wherein the depth image is one of a plurality of depth images captured by at least one camera disposed on a robot, and wherein the robot in a tracking mode is configured to be positioned or to position

itself behind the subject. The marker-less gait monitoring method according to any described above, wherein the robot is configured to keep at least one camera within a range of trailing distances relative to the subject, and wherein the range of trailing distances is determined such that the depth image includes a foot and a shank of the at least one of the subject's lower limbs. The depth image comprises a back view image of one or both feet and shanks of the subject. The marker-less gait monitoring method may further comprise: from the point cloud, identifying the data points corresponding to at least one of a right lower limb and a left lower limb of the subject.

[0005] The marker-less gait monitoring method may comprise: sorting the data points into a plurality of clusters by respective depth values of the data points; responsive to finding two groups of clusters, allocating one of the two groups of clusters to a right lower limb of the subject and another of the two groups of clusters to a left lower limb of the subject; and responsive to finding only one group of clusters, allocating some of the data points to at least one lower limb of the subject. The marker-less gait monitoring method may further comprise: prior to allocating the selected ones of the data points to one of the foot model and the shank model, identifying the data points corresponding to each of a right lower limb and a left lower limb of the subject. The step of allocating selected ones of the data points to one of a foot model and a shank model may comprise: using an expansion-segmentation method to find a segmentation line separating the data points of the foot model and the data points of the shank model, wherein the segmentation line is defined relative to a lowest point in a cluster of data points of the point cloud. The allocating of some of the data points to the at least one lower limb of the subject may be based on vertices of a respective projected foot-shank model, in which the respective project foot-shank model is associated with an earlier depth image. The allocating of some of the data points may be based on a contour-segmentation method in which two convex hulls are defined by the respective vertices of a foot model part of the respective projected foot-shank model and a shank model part of the respective projected foot-shank model.

[0006] The marker-less gait monitoring method according to any described above, wherein optimizing parameters of the foot-shank model further comprises determining one or more shape parameters of the foot-shank model. The determining of the one or more pose parameters and/or one or more shape parameters comprises optimizing a foot-shank cost

function, and wherein the foot-shank cost function comprises a foot fitting cost and a shank fitting cost. The foot-shank cost function may further comprise a shape penalty to prevent oversizing of the foot model. The foot-shank cost function may further comprise an orientation penalty to penalize a non-alignment of the shank model relative to a vertical axis. The foot-shank cost function may further comprise a foot flexing term configured to penalize a flexing movement of the foot model, and wherein the flexing movement includes any one or both of a dorsiflexion-plantarflexion of the foot and an inversion-eversion of the foot. The foot-shank cost function may further comprise a temporal penalty, and wherein the temporal penalty is based on an assumption that the subject moves relatively slowly. The gait speed of the subject may be characteristic of a walking gait.

[0007] The marker-less gait monitoring method may further comprise: using the one or more pose parameters to determine an anterior-posterior distance; and using the anterior-posterior distance to identify a gait event. The marker-less gait monitoring method may further comprise: using the one or more pose parameters and/or one or more shape parameters to generate gait and pose information.

[0008] In another aspect, the present disclosure provides a method of gait rehabilitation assistance for a subject, comprising: tracking the subject over a length of time; and performing a gait and balance analysis for the subject using the gait and pose information of the subject, wherein the gait and pose information is generated according to any of the embodiments described above.

[0009] In yet another aspect, the present disclosure provides a robot for use with a subject, the robot comprising: a camera; a plurality of wheels, the plurality of wheels being coupled with the camera such that the robot in a tracking mode is configured to track the subject with the camera configured to concurrently capture back view depth images of the subject; and a controller operably coupled with the camera, the controller being configured to acquire image data from the camera, wherein the controller is configured to use the image data to perform the marker-less gait monitoring method according to any of the embodiments described above.

BRIEF DESCRIPTION OF DRAWINGS

[0010] Fig. 1 is a schematic block diagram of a robot according to an embodiment of the present disclosure;

[0011] Fig. 2A illustrates the robot of Fig. 1 positioned relative to a subject;

[0012] Fig. 2B is a partial front view of the robot of Fig. 2A;

[0013] Figs. 3A, 3B and 3C are schematic flow diagrams of a gait monitoring method according to embodiments of the present disclosure;

[0014] Fig. 4 is a schematic diagram of a foot-shank model according to one embodiment of the present disclosure;

[0015] Fig. 5 illustrates an example of a point cloud scene acquired by the robot of Fig. 1;

[0016] Fig. 6A illustrates an expansion-segmentation method of Fig. 3B;

[0017] Fig. 6B illustrates a contour-segmentation method of Fig. 3C;

[0018] Figs. 7A to 7D illustrate various types of gait events; and

[0019] Figs. 8A and 8B illustrate examples of the robot used in gait monitoring and/or gait rehabilitation, according to embodiments of the present disclosure.

DETAILED DESCRIPTION

[0020] Reference throughout this specification to “one embodiment”, “another embodiment” or “an embodiment” (or the like) means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment. Thus, the appearance of the phrases “in one embodiment” or “in an embodiment” or the like in various places throughout this specification are not necessarily all referring to the same embodiment. Furthermore, the described features, structures, or characteristics may be combined in any suitable manner in one or more embodiments. In the following description, numerous specific details are provided to give a thorough understanding of embodiments. One skilled in the relevant art will recognize, that the various embodiments be practiced without one or more of the specific details, or with other methods, components, materials, etc. In other instances, some or all known structures,

materials, or operations may not be shown or described in detail to avoid obfuscation.

[0021] The word “exemplary” is used herein to mean “serving as an example, instance, or illustration.” Any embodiment described herein as “exemplary” is not necessarily to be construed as preferred or advantageous over other embodiments. As used herein, the singular ‘a’ and ‘an’ may be construed as including the plural “one or more” unless apparent from the context to be otherwise.

[0022] Terms such as “first” and “second” are used in the description and claims only for the sake of brevity and clarity, and do not necessarily imply a priority or order, unless required by the context. The terms “about” and “approximately” as applied to a stated numeric value encompasses the exact value and a reasonable variance as will be understood by one of ordinary skill in the art, and the terms “generally” and “substantially” are to be understood in a similar manner, unless otherwise specified. The term “concurrent” or “concurrently” is used loosely to refer to two or more occurrences that at least partially overlap in time, the occurrences not necessarily starting and/or ending at the same time instant.

[0023] Fig. 1 is a schematic block diagram of a gait monitoring device **100** according to embodiments of the present disclosure. The gait monitoring device **100** may be configured as a standalone, independently operable device or it may be incorporated with another self-navigating vehicle or robot. The terms “gait monitoring device”, “robotic mobility assistant”, “gait rehabilitation robot”, “feet tracking apparatus”, or the like are thus used interchangeably in the present disclosure. For the sake of brevity, the term “robot” will be used hereafter. The robot **100** includes a controller **102** configured to perform a marker-less gait monitoring method **200** according to embodiments of the present disclosure. The robot **100** may include components **108** operably coupled to the controller **102**, including but not limited to one or any combination of the following: a battery, a communications device, data storage, etc. The robot **100** may include a plurality of motorized wheels or other motion-enabling devices **104** to enable the robot **100** to travel “overground”, i.e., on various floorings or the ground. The controller **102** is configured to enable self-navigation of the robot **100** such that the robot **100** in a tracking mode is configured to track a subject **90** as he/she walks around (i.e., to physically follow along or to move along a similar trajectory

after/behind the subject).

[0024] According to embodiments of the present disclosure, the robot **100** is configured to generally travel in substantially the same or a similar direction **110** as the direction **112** of travel of the subject **90**, as illustrated in Fig. 2. For the sake of brevity and ease of reference, the terms “forward direction” or “forward orientation” **110** are used in the present disclosure interchangeably to refer to the orientation of the robot **100**, the direction **110** of travel of the robot **100**, and/or the direction of travel **112** of the subject **90**. The terms “front”, “back”, “behind”, are to be understood relative to the forward orientation **110** and/or relative to the orientation of the robot **100**. The term “vertical” **114** is used for referring generally to a direction normal to the ground, simply for ease of reference and not to be limiting.

[0025] The robot **100** includes one or more camera **106** (or is operably coupled with one or more camera **106**) such that as the robot **100** tracks the subject **90**, the controller is configured to acquire at least one of a plurality of depth images captured by the at least one camera **106**. In other words, the robot in a tracking mode is configured to be positioned or to position itself behind the subject **90** while (concurrently or over a period of time) capturing a plurality of back view images of the subject **90** using the at least one of the one or more cameras **106** disposed on the robot **100**. In the following, for the sake of brevity, references to one camera **106** or at least one camera **106** will be understood to refer to one or more cameras **106** disposed on the robot **100**. The motion-enabling devices or wheels **104** may be coupled with the camera **106** such that the robot **100** in a tracking mode is configured to track the subject **90** with the camera **106** configured to concurrently capture back view images of the subject **90**. The robot **100** is configured to follow after a subject **90** even if the subject **90** is walking “over ground” and not confined to a treadmill or a special straight track. The special straight track refers to the typical tracks found in a kinematics laboratory for gait analysis or gait rehabilitation.

[0026] The camera **106** may be configured to continuously and/or intermittently capture images (e.g., still images, videos, or both) over a period of time, and to transmit the images/image data to the controller **102**. The camera **106** may be a depth camera, or an RGB-D (Red Green Blue Depth) camera in some cases. The camera **106** is configured to capture images in which each pixel of an object in the image is associated with depth value.

The camera **106** is preferably one configured to convert the image data into data points of a point cloud in a three-dimensional space (the terms “data points” and “points” in the context of a point cloud are used interchangeably in the present disclosure). The controller **102** is configured to acquire images/image data (e.g., point cloud, depth scene, etc.) from the camera **106**. It will be understood that the robot **100** can be powered by a battery **190** or other means to enable operation of the controller **102** and the camera **106**. At the same time, the robot **100** in a tracking mode is powered to follow in the wake of the subject **90**.

[0027] The camera **106** is preferably disposed on the robot **100** at a camera height **116** (Fig. 2B) above the ground **10** to capture images of one or both limbs **87** of the subject **90**. In some embodiments, the camera **106** is disposed at the front **113** of the robot **100** at a camera height **116**. In some embodiments, the camera **106** is supported by an adjustable bracket **105** so that the camera height **116** is correspondingly adjustable. In one non-limiting example, the camera height is about 10 centimeters.

[0028] In the present disclosure, the terms “limb” and “lower limb” are used interchangeably to refer collectively to a foot **80** and at least a part of a leg **60** including where the leg joins the foot at an ankle **75**. The term “shank” **70** as used herein refers to at least part of the leg **60** from the ankle **75** upwards, and does not necessarily refer to the whole part of the leg **60** from the ankle **75** to the knee **65**. The term “foot-shank” **87** refers collectively to the foot **80** and the connected shank **70** of the same leg **60**.

[0029] According to one embodiment of the present disclosure, the robot **100** is configured to be positioned or to position itself behind a (human) subject **90** at least for a majority of the time if the robot is in a tracking mode. The robot **100** may be configured to position itself such that the camera **106** is at least a trailing distance **120** apart from and at the back of the subject **90**. The trailing distance **120** required may vary owing to variations in the stature and body proportions of different subjects **90** relative to the camera height **116**. A back view of the subject **90** is within the field of view of the camera **106**, and back view images of the subject **90** are captured by the camera **106**.

[0030] The robot **100** may be configured to position itself at least the trailing distance **120** apart from and at the back of the subject such that at least a foot-shank **87** (and preferably both feet and shanks **87**) of the subject **90** is within the field of view of the camera **106**. That

is, the controller **102** may be configured to determine a minimum trailing distance **120** based on whether the back view images of the feet and shanks **87** of the subject **90** are captured by the camera **106**. As the subject **90** ambulates, the robot **100** is configured to follow behind the subject **90**. The robot **100** is configured to move substantially in the first direction **110**, keeping the subject **90** in front of the robot **100** and within a field of view of the camera **106**. The robot **100** may be configured to trail the subject **90**, spaced apart from the subject **90**, keeping the subject generally within a range between a minimum distance and a maximum distance in front of the camera **106**. In some embodiments, the robot includes a brace **130** extending from the robot **100** in the forward direction. The back and/or sides of the subject may abut or be supported by the brace **130**, in accordance with the needs or preference of the subject. The brace **130** may be adjustable such that the subject **90** can adjust his/her distance from the robot **100** as needed.

[0031] The robot **100** is configured to perform various embodiments of the marker-less gait monitoring method **200** disclosed herein. The gait monitoring method **200** is preferably based on a model **300** formed by geometric shapes. As shown in Fig. 4, one example of the model **300** according to the present disclosure includes a foot model **310** integrated with a shank model **320**. The foot model **310** corresponds to the foot **80** of the subject **90**. The foot model **310** is partially based on a trapezoidal prism **312** defining a foot length l and a foot height h . A proximal end **314** of the trapezoidal prism is fitted with a U-shape surface **316** characterized by a heel radius r to model a heel of the foot. The U-shape surface extends beyond the trapezoidal prism in a lengthwise orientation by an offset d to model a protrusion resulting from the calcaneus bone. A top face and a distal face of the trapezoidal prism are preferably omitted from the foot model **310**.

[0032] The model **300** further includes a shank model **320** corresponding to the shank **70** of the subject **90**. The shank model **320** is modeled as a partial tapered-cylinder **322** (also referred to as a conical frustrum or truncated cone) characterized by a first shank radius r_1 and a second shank radius r_2 , and a shank height s . The shank height s may be defined to be shorter than the actual distance from an ankle to a knee of the subject. A front part of the tapered-cylinder is preferably omitted from the shank model **320**.

[0033] The top extremity **318** of the foot model **310** near the proximal end **314** joins a

bottom extremity **328** of the shank model **320**. A foot **80** and the shank **70** connected to the foot **80** are referred to collectively as “foot-shank” **87** or interchangeably with the terms “lower limb” **87** or “limb” **87**. The top extremity **318** and the bottom extremity **328** share the same position in three-dimensional space but remain free to rotate relative to one another. The foot model **310** and the shank model **320** share the same position at a common origin (O). The foot model **310** and the shank model **320** may have the same or different orientations at various times. In other words, the foot model **310** and the shank model **320** are adjoining parts of one foot-shank model **300**, able to rotate relative to one another about the common origin (O). In some embodiments, the first shank radius $r1$ is preferably equal to the heel radius r . In some embodiments, the first shank radius $r1$ and the heel radius r are approximately the same.

[0034] One embodiment of the gait monitoring method **200** will now be described with reference to Figs. 3A, 3B, and 3C. The gait monitoring method **200** can be implemented using just one camera **106** to acquire (210) depth images of a subject **90**, in which the camera **106** is configured to follow the subject **90** if the subject **90** ambulates. The camera **106** is oriented to capture back view depth images of the lower limbs **87** of the subject **90**. The depth measurement or depth value associated with each pixel is measured along the forward direction **110**, relative to the camera **106**. The method **200** executed by the controller **102** is configured to assume that the limbs **87** and the ground **10** (ground plane **11**) are the only objects in each depth image. That is, it is assumed that there are no other large objects near the camera **106**. The depth images are preferably processed with morphological opening and closing to remove noise.

[0035] For the sake of clarity, the following will be described with respect to one depth image and with respect to one limb **87**. It will be understood that in actual implementation a plurality of depth images will be captured over the course of the subject’s ambulation, and that the plurality of depth images will be processed for both limbs **87** (if both limbs **87** are visible to the camera **106**).

[0036] The gait monitoring method **200** further includes acquiring (214) a point cloud **400** from the depth image. The controller **102** may be configured to reconstruct a point cloud **400** (e.g., Fig. 5) from the depth image or to acquire the point cloud **400** directly from the

camera **106**. For the sake of clarity, the point cloud is indicated by an outline **406** in Fig. 5. It will be understood that the point cloud **400** would appear as a plurality of data points distributed over a space/area.

[0037] The controller **102** is configured to identify **(220)** data points corresponding to the left lower limb and the right lower limb of the subject **90**. Data points corresponding to a depth further than a maximum depth or a maximum distance from the camera **106** are removed **(221)**. In some examples, the controller **102** is configured to discard or disregard data points that are further than a maximum depth of about 1.3 meters from the camera **106**. The maximum depth may be varied. The ground plane **11** is also identified. Data points at the ground plane **11** and below are discarded **(222)**. Connected data points with similar depth values are formed into clusters, which are then sorted into two groups of clusters using a K-Mean algorithm. Each of the two groups corresponds to one of the two limbs **87** and are identified as left limb and right limb according to the positions of the clusters in the image plane. Fig. 5 shows an example of a point cloud **400** including data points corresponding to two limbs **402**, **404** (data points corresponding to the ground plane **11** have been discarded). In this example, the group of clusters on the left **402** corresponds to the left limb (lower limb) of the subject **90**, and the group of clusters on the right **404** corresponds to the right limb (lower limb) of the subject **90**.

[0038] After the point clouds have been sorted into data points corresponding to the left limb and data points corresponding to the right limb, the next step is to distinguish between or to identify **(230)** for data points of the same limb, the data points corresponding to the foot **80** and the data points corresponding to the shank **70**.

[0039] In a preferred embodiment, an expansion-segmentation method **(230)** or algorithm is executed by the controller **102** to find a segmentation line separating the data points of the foot **80** from the data points of the shank **70**. As shown in Fig. 6A, at the beginning of an optimization process **230** (Fig. 3B), an initial segmentation line **410** may be defined at $r_{initial}$ rows, relative to a lowest data point **410** of the cluster **404** **(231)**. The data points (pixels) are allocated **(232)** to either the foot model **310** or the shank model **320**. All data points in the cluster **404** below the segmentation line **410** are labeled as foot pixels or foot points (i.e., allocated to the foot model **310**) and all data points in the cluster **404** above the

segmentation line **410** are labeled as shank pixels or shank points (i.e., allocated to the shank model **320**). In other words, data points above the segmentation line are selected and allocated to the shank model **320**, and data points below the segmentation line are selected and allocation to the foot model **310**. These labeled pixels (labeled data points) are loaded into an optimization algorithm (**240**) to localize the foot and the shank, and to also obtain an output cost indicative of a quality of the model fitting. Using a cost function optimization routine (**240**) based on the foot-shank model **300**, the cost may be iteratively determined (**243**, **244**) until the cost of a current iteration is not smaller than the cost of the previous (immediately preceding) iteration. The pose parameters (and in some cases, both the pose parameters and the shape parameters) corresponding to the previous iteration may be taken to be the optimization routine output associated with the current segmentation line (**246**). The segmentation line is re-defined (**236**), and the optimization routine output associated with the re-defined segmentation line is determined. In each re-definition of the segmentation line, the segmentation line is moved up by a predetermined number of rows ($\Delta r_{\text{expansion}}$) from the previous value of r . In other words, a new segmentation line **412** is defined (**236**), in which the number of rows between the segmentation line **412** from the lowest data point **420** increases with each succeeding iteration. For each segmentation line **412**, the data points are again differentiated into foot points and the shank points. That is, data points above the segmentation line **412** are labelled as shank points and data points below the segmentation line **412** are labelled as foot points. This process is reiterated until the cost associated with the current segmentation line is not smaller than the cost associated with the previous (immediately preceding) segmentation line (**234**). The optimized pose parameters (and in some cases both the pose parameters and the shape parameters) associated with the previous segmentation line are taken to be the generated pose parameters (or the generated pose parameters and the generated shape parameters) (**238**, **280**). For example, if the output cost associated with the segmentation line **414** is not smaller than the output cost associated with the segmentation line **412**, the segmentation line **412** will be used as the foot-shank segmentation line **430**. The optimized allocation of data points (pixels) to either the foot model or the shank model, the pose parameters, and/or the shape parameters will then be based on the foot-shank segmentation line **430**. The gait monitoring method **200** includes generating the pose parameters and/or the shape parameters (**238**). The gait monitoring method **200** may further include acquiring pose information and/or gait

information (280) and using the pose information and/or gait information in various gait assistive or mobility assistive applications (290).

[0040] Preferably, the robot 100 in a tracking mode is configured to keep the camera 106 within a range of trailing distances 120 (relative to the subject 90) such that most if not all of the images include both lower limbs 87 of the subject 90. Alternatively, the range of trailing distances 120 may be determined such that a majority or more than half of the images include one or both of the lower limbs 87 of the subject 90.

[0041] In some situations, only one of the lower limbs 87 is visible to the camera 106 and/or only one cluster of data points (or one group of clusters) is found in the point cloud. Such a situation may arise if one of the lower limbs 87 is occluded by the other limb 87, e.g., when the subject 90 is turning or changing the generation direction of travel 112. Such a situation may also arise if the two limbs 87 contact one another such that the algorithm identifies all data points as one large cluster. If only one cluster is found for an image in a frame (252) (e.g., Fig. 6B), the controller 102 may be configured to carry out contour-segmentation (250) in place of the expansion-segmentation method (230) (Fig. 6A) described above. In contour-segmentation, the vertices 440 of the foot model and the shank model (foot-shank model) from the previous frame (or another earlier frame) may be projected (254) (Fig. 3C) on to a two-dimensional image plane as shown in Fig. 6B. Two convex hulls 451, 452 are formed from the respective projected vertices 440. The pixels inside the hulls are labeled as foot pixels 461 if they lie near a surface of the (projected) foot model, and the pixels inside the hulls are labeled as shank pixels 462 if they lie near a surface of the (projected) shank model. That is, data points inside the hulls and near the surface of the projected foot model are selected and allocated to the foot model 310. Data points inside the hulls and near the surface of the projected shank model are selected and allocated to the shank model 320.

[0042] In other words, the data points are sorted into a plurality of clusters by respective depth values of the data points. Responsive to finding two groups of clusters, one of the two groups of clusters is associated with or allocated to a right lower limb of the subject, and another of the two groups of clusters is allocated to or associated with a left lower limb of the subject. Alternatively, responsive to finding only one cluster (or only one group of clusters), some of the data points are allocated (232) to or associated with at least one lower

limb **87** of the subject. That is, some of the data points may be allocated to or associated with at least one lower limb of the subject. This may be based on vertices of a respective projected foot-shank model, in which the respective projected foot-shank model is associated with an earlier depth image. The two convex hulls of the contour-segmentation method may be defined by the vertices of a foot model part of the respective projected foot-shank model and the vertices of a shank model part of the respective projected foot-shank model.

[0043] In other words, if only one cluster (or one group of clusters) is found, a contour segmentation process (**250**) is triggered. There are two possible scenarios. In a first scenario, only one convex hull has data points within its vertices **440**. The data points within the convex hull will be allocated (**232**) to the one lower limb **87** corresponding to the respective convex hull model (right lower limb or left lower limb). In a second scenario, both convex hulls have data points within their vertices **440**. The data points of the left convex hull will be allocated (**232**) to the left lower limb, and the data points of the right convex hull will be allocated (**232**) to the right lower limb.

[0044] After the selected data points have been allocated to the foot model part or the shank model part of the foot-shank model **300** using the expansion segmentation method (**230**) or the contour segmentation method (**250**), the selected data points allocated to the foot model **310** are input into a foot model optimizer to find the optimal foot shape parameters and/or foot pose, and the selected data points allocated to the shank model **320** are input into a shank model optimizer to find the optimal shank shape parameters and/or shank pose parameters (**240**). In some applications, only the pose parameters are generated. In some other applications, both the pose parameters and the shape parameters are generated.

[0045] According to some embodiments of the present disclosure, the optimization (**240**) is based on a cost function that is defined by at least two components, namely, a foot component and a shank component. In some embodiments, the foot-shank cost function (E) is defined by three or more components, namely, the foot fitting cost (E_{foot}), the shank fitting cost (E_{shank}), and other additional fitting costs ($E_{additional}$), as shown in Equation (1) below:

$$E = E_{foot} + E_{shank} + E_{additional} \quad (1)$$

where E is the foot-shank cost function corresponding to a total fitting cost of one foot-shank 87.

[0046] In some embodiments, the foot model optimizer and the shank model optimizer may be configured to update a change of the foot pose parameters and a change of the shank pose parameters respectively, instead of directly updating the foot pose parameters and the shank pose parameters. The final poses output from the optimizers in a frame t may be defined in terms of a rotation matrix R_t and a three-dimensional vector $\vec{t}_t$, respectively.

For example, in a first iteration of frame t , the current rotation and translation $\tilde{R}_t$ and $\tilde{t}_t$ are initialized to be R_{t-1} and $\vec{t}_{t-1}$ respectively. In a subsequent iteration, the optimizer updates the changes ΔR and $\Delta \vec{t}$, which are then integrated into the current pose parameters as shown in Equation (2) below. Changes in the pose parameters are reset to the identity a rotation and the zero vector for the next iteration.

$$\tilde{R}_{t,i+1} = \tilde{R}_{t,i} \Delta R^T, \quad \tilde{t}_{t,i+1} = \tilde{t}_{t,i} + \Delta \vec{t} \quad (2)$$

where i is the iteration count which is omitted in the subsequent text to avoid clutter.

[0047] The terms employed in the respective fitting costs are described below. The pose parameters of the foot at frame t may be defined as a rotation matrix $R_{foot,t}$ for orientation and a three-dimensional vector $\vec{t}_{model,t}$ for translation. The rotation may be transformed into a quaternion $\vec{q}_{foot,t}$. In the camera frame, C , the observed data points ${}_C\vec{p}_t$, are transformed into the model frame, M , as ${}_M\vec{p}_t$ in all computations below unless stated otherwise:

$${}_M\vec{p}_t = \Delta R_{foot,t} \tilde{R}_{foot,t}^T \left[{}_C\vec{p}_t - (\tilde{t}_{model,t}) + \Delta \vec{t}_{model,t} \right] \quad (3)$$

[0048] The subscript t is omitted in the following text to avoid clutter for the sake of clarity.

[0049] The foot fitting cost is given below in Equation (4):

$$E_{foot} = E_{dist,foot} + w_1 E_{silhouette,foot} + w_2 E_{ground} + w_3 E_{flat} \quad (4)$$

where the w terms represent weights of the respective costs.

[0050] The positions of the observed data points with respect to the current model are identified. The closed-form point-to-model distances are calculated. The term $E_{dist,foot}$, which refers to a sum of the distances of each sampled data point to the foot model can be computed as follows:

$$E_{dist,foot} = \sum_{\vec{p} \in S_{FT}} D(\vec{p}) \quad (5)$$

where S_{FT} is the set that contains the sampled foot data points, transformed into the model frame, and $D(\cdot)$ represents the point-to-model distance for that particular data point.

[0051] Shape penalty coefficients may be applied to $E_{dist,foot}$ to prevent the model from becoming oversized. This addresses oversizing that may have result from the shape parameters optimization process in which reduction of the point-to-model distance may be at the expense of increasing the model size.

$$r_{penalty} = \frac{1}{1 + e^{-a_r r}} \quad (6)$$

$$h_{penalty} = \frac{1}{1 + e^{-a_h h}} \quad (7)$$

where a_r and a_h are scalar coefficients.

[0052] The silhouette term $E_{silhouette,foot}$ penalizes the model when it does not fall within the silhouette of the observed data points when projected onto the two-dimensional image plane.

$$E_{silhouette,foot} = \sum_{\vec{u} \in S_{FM}} \min_{\vec{x} \in S'_{foot}} \|\vec{u} - \vec{x}\|_2 \quad (8)$$

where S_{FM} is a set in $\mathbb{R}^2$ containing the key points of the foot model, transformed into frame C and project into the image plane, S'_{foot} is the set of sampled foot pixels, and $\|\cdot\|_2$ denotes the l_2 norm. This term keeps the model within the observation point cloud, especially when the feet are substantially in the forward orientation and resulting in

ambiguity in the orientation of the foot.

[0053] A penalty is preferably imposed to have a more physically realistic pose in which the foot does not collide with other objects. More specifically, a ground penalty is preferably imposed so as to address the appearance of the front of the foot model dipping or penetrating into the ground, as may be the case when only one side of the foot is visible to the camera. E_{ground} may be imposed to penalize an extension of the foot model below the ground plane, as shown in equation 9:

$$E_{ground} = \sum_{\vec{u} \in S_{FM} \cap S_{below}} N\left(\tilde{R}_{foot} \Delta R_{foot}^T \vec{u} + \left(\tilde{t}_{model} + \Delta \vec{t}_{model}\right), \vec{n}_{ground}, \vec{c}_{ground}\right) \quad (9)$$

where S_{below} is the set of data points below the ground plant, $\vec{n}_{ground}$ is the normal of the ground plane, $\vec{c}_{ground}$ is a data point on the ground plane, and $N(\vec{p}, \vec{n}, \vec{c})$ represents the point-to-plane distance.

[0054] The controller **102** may be configured to run a shape optimization algorithm for a first plurality of $n_{shaping}$ frames in which a penalty E_{flat} is preferably imposed on the foot fitting cost. E_{flat} penalizes the cost when the sole of the foot model is not parallel to the ground plane. This is to take into account that at this stage the subject has not started walking and that the subject is essentially standing with both the feet flat on the ground (as opposed to being at least partially lifted up from the ground). The penalty E_{flat} may be expressed as shown in equation 10 below:

$$E_{flat} = \left| \left(\tilde{R}_{foot} \Delta R_{foot}^T (0, 1, 0)^T \right) \cdot \vec{n}_{ground} \right| + \left| \left(\tilde{R}_{foot} \Delta R_{foot}^T (1, 0, 0)^T \right) \cdot \vec{n}_{ground} \right| \quad (10)$$

where S_{below} is the set of data points below the ground plant, $\vec{n}_{ground}$ is the normal of the ground plane, $\vec{c}_{ground}$ is a data point on the ground plane, and $N(\vec{p}, \vec{n}, \vec{c})$ represents the point-to-plane distance.

[0055] The first term represents an angle between a forward axes of the foot model and the ground normal. The second term represents an angle between a side axis of the foot model and the ground normal. When the foot model is flat on the ground, both the first term and

the second term are zero.

[0056] An orientation of the shank may be represented by a quaternion. Preferably, the orientation of the shank is represented using Euler angles such that a rotation matrix of the shank is:

$$R_{shank} = R_x R_y R_z \quad (11)$$

with R_x , R_y , and R_z as the basic rotation matrices.

[0057] For this step, rotation about the z-axis may be ignored as only the data points on the shank model with negative y values (relative to the y-axis) are taken into consideration. In other words, the data points used are those visible to the camera **106** viewing from behind the subject. The orientation of the shank may be determined by the angles $\vec{\theta}_{shank} = \{\theta_x, \theta_y\}$. The shank model is configured to share the same translation vector $\vec{t}_{model}$ with the foot model.

[0058] The shank fitting cost is given below:

$$E_{shank} = E_{dist,shank} + w_4 E_{silhouette,shank} + w_5 E_{orientation} + w_6 E_{radii} \quad (12)$$

[0059] The terms $E_{dist,shank}$ and $E_{silhouette,shank}$ are defined similarly to their foot counterparts:

$$E_{dist,shank} = \sum_{\vec{p} \in S_{ST}} D(\vec{p}) \quad (13)$$

$$E_{silhouette,shank} = \sum_{\vec{u} \in S_{SM}} \min_{\vec{x} \in S'_{shank}} \|\vec{u} - \vec{x}\|_2 \quad (14)$$

where S_{ST} is the set that contains the sampled shank points that are transformed into the model frame, S_{SM} is a set in $\mathbb{R}^2$ containing the key points of the shank model that are transformed into frame C and projected onto the image plane, and S'_{shank} is the set of sampled shank pixels.

[0060] Similarly, oversizing of the shank model is penalized by applying (multiplying)

shape penalty coefficients to the $E_{dist,shank}$ during shape parameters optimization:

$$r_{1,penalty} = \frac{1}{1 + e^{-a_{r1}r1}} \quad (15)$$

$$r_{2,penalty} = \frac{1}{1 + e^{-a_{r2}r2}} \quad (16)$$

$$s_{penalty} = \frac{1}{1 + e^{-a_s s}} \quad (17)$$

where a_{r1} , a_{r2} , and a_s are scalar coefficients.

[0061] An orientation term $E_{orientation}$ is preferably applied to the shank fitting. $E_{orientation}$ penalizes the cost function when the leg is not aligned with a vertical axis **114**, i.e., a non-alignment of the shank model relative to the vertical axis **114**. The effect is to help “push” the shank model upright without significantly affecting a tilting of the shank model. This enables the model to avoid being trapped in a local minimum after oscillating between an anterior axis and a posterior axis in the course of walking, i.e., it gets around the problem of the model being unable to “straighten up” when the subject stops walking and stands. $E_{orientation}$ may be expressed as:

$$E_{orientation} = 1 - (\tilde{R}_{shank} \Delta R_{shank}^T * (0,0,1)^T) \cdot \vec{n}_{ground} \quad (18)$$

[0062] A term E_{radii} may be applied during the shape optimization process to better reflect that the upper part of the shank (second shank radius r_2) is typically thicker than the lower part of the shank (first shank radius r_1):

$$E_{radii} = \frac{1}{1 + e^{-r1/r2}} \quad (19)$$

[0063] Preferably, additional constraints are imposed on the foot-shank cost function, E . According to some embodiments of the present disclosure, the additional constraints include but are not limited to penalizing foot flexing movements. The typical ankle joint would allow the foot to flex in a wide range of movements relative to its standing orientation/position. In the course of walking, the typical gait of a healthy subject would involve a certain amount of foot flexing movements. Although it would appear counter-intuitive to penalize the foot-shank model in this manner, nonetheless it has been

experimentally verified that more realistic optimization outcomes can be obtained by incorporating any one or more of the additional fitting costs given in Equation (20) below:

$$E_{additional} = w_7 E_{limit} + w_8 E_{temporal,rot} + w_9 E_{temporal,trans} \quad (20)$$

[0064] In one aspect, the controller **102** is configured to take into account a foot flexing term E_{limit} which penalizes dorsiflexion, plantar flexion, inversion, and/or eversion of the foot. These flexing movements of the foot **80** may be taken with respect to the limb **87** in a standing pose in which (i) the shank **70** is substantially aligned along a vertical axis **114** (normal to the ground **10**), (ii) the foot **80** is pointing in the forward direction **110**, and (iii) the model surface corresponding to the sole being substantially flat and parallel to the ground. E_{limit} may be defined as shown in Equation (21) below:

$$E_{limit} = (\tilde{R}_{shank} \Delta R_{shank}^T (0,0,1)^T) \cdot (\tilde{R}_{foot} \Delta R_{foot}^T (0,1,0)^T) + (\tilde{R}_{shank} \Delta R_{shank}^T (0,0,1)^T) \cdot (\tilde{R}_{foot} \Delta R_{foot}^T (1,0,0)^T) \quad (21)$$

where the first term relates to a dorsiflexion-plantarflexion of the foot, and the second term relates to the inversion-eversion of the foot.

[0065] In another aspect, the robot 100 is configurable with a walking mode in which the robot 100 is able to track the movements of the subject even if the subject is moving relatively slowly. Such a walking mode is particularly useful when the robot 100 is used with stroke patients or post-surgery patients who are unable to make the typical striding motion. In one example, the foot-shank cost function, E , is configured to further take into account temporal term $E_{temporal,rot}$ and/or $E_{temporal,trans}$ as shown in Equations (22) and (23) below:

$$E_{temporal,rot} = Q(R_{foot,t-1}, \tilde{R}_{foot} \Delta R_{foot}^T) + Q(R_{shank,t-1}, \tilde{R}_{shank} \Delta R_{shank}^T) \quad (22)$$

$$E_{temporal,trans} = \| \vec{t}_{model,t-1} - (\vec{t}_{model,t} + \Delta \vec{t}_{model,t}) \|_2 \quad (23)$$

where S_{ST} is the set that contains the sampled shank points that are transformed into the model frame, S_{SM} is a set in $\mathbb{R}^2$ containing the key points of the shank model that are transformed into frame C and projected onto the image plane, and S'_{shank} is the set of

sampled shank pixels.

[0066] In Equation (22), $Q(\cdot)$ finds the smallest angular distance between two rotations. The temporal terms impose a temporal penalty on a pose that is significantly different from a pose of the previous frame. In other words, the controller **102** is configured to assume that the subject moves relatively slowly, and that the change of pose from one frame to the next frame is not significant.

[0067] In some embodiments, the controller **102** is configured to optimize the cost function using the Levenberg-Marquardt (LM) algorithm.

[0068] For a first $n_{shaping}$ number of frames, both the shape parameters (model geometry) and the pose parameters (orientations and positions of the foot-shank) are optimized. In the shape optimization part of the optimization, the camera **106** may be configured to acquire a number of frames of the subject **90** with one foot **80** facing the forward direction **110/112** and the other foot facing a lateral direction (as shown in Fig. 5). A lateral foot length l may be optimized at this stage.

[0069] The controller **102** may be configured to start a tracking process upon completion of the shape optimization or upon acquiring the shape parameters of the subject **90**. During the tracking process, the controller **102** is configured to optimize the pose parameters in each iteration (pose optimization), without the need to perform shape optimization.

[0070] In some embodiments, the pose optimization process is iterated until the foot-shank cost (e.g., according to Equation (1)) is smaller than a cost threshold (**244**, **246**). In some embodiments, the pose optimization process is iterated until a change in the foot-shank cost from one iteration to a next iteration is smaller than a change threshold. In some embodiments, the pose optimization process is iterated until a change in one or more of the parameters in the foot-shank cost function falls below respective parameter thresholds. In some embodiments, the pose optimization process is iterated until the number of iterations reaches a maximum number of iterations, n_{max} . A Kaman filter may be implemented for the pose parameters to smoothen the trajectory and to reject outliers.

[0071] The outcome of the optimization may be embodied in the form of a database of pose-

related information for one or more subjects **90 (280)**. The robot **100** may be configured to perform one or more physiotherapy/kinesiotherapy/physical therapy-related methods customized to a selected subject, based on the pose-related information obtained (**280**) as described above for the selected subject.

[0072] To illustrate, the following describes one example in which the robot **100** is configured to perform a method of mobility-related assistance (**290**) enabled by the pose-related information (**280**). The controller **102** may be configured to extract gait parameters and identify one or gait events based on the pose-related information obtained (**280, 290**). The gait events may include but are not limited to one or more of the following: stride length, stride width, stride time, stance time, swing time, and/or combinations thereof.

[0073] One approach for determining gait events may use the distance between a foot model and the ground plane. This approach is found to be prone to false alarms and misdetection, for example, when the foot is far from the camera, or when one foot blocks another foot from the view of the camera. Advantageously, as described below, embodiments of the present disclosure are found to overcome these problems.

[0074] According to embodiments of the present disclosure, the controller **102** is configured to identify various gait events including, but not limited to, the examples illustrated in Figs. 7A to 7D. Fig. 7A is a back view of the subject in a standing pose **710**, and Fig. 7B is a back view of the subject walking **720**. Fig. 7C is a side view showing a heel-strike event **730** occurring with heel **732** of the front leg **734** striking the ground **10**. Fig. 7D is a side view showing a toe-off event **740** occurring with the toe **742** of the back leg **744** pushing off the ground **10**. In one embodiment, the controller is configured to identify gait events **102** based on the anterior-posterior (AP) distance of the foot **80** from the camera **106**, which is in turn determined from the pose information or pose-related information. A local maxima of the AP distance (e.g., the front foot or the foot being furthest away from the camera) may be determined and used as indicative of a heel-strike (HS). A local minima of the AP distance (e.g., the back foot or the foot being closest to the camera **106** in an instant before the same foot swings forward) may be determined from the pose-related information, and the local minima may be used as indicative of a toe-off (TO). Based on the pose-related information, the controller **102** may be configured to identify the occurrence of a heel-strike, and to label

the heel-strike as an end of one gait cycle and a start of a next gait cycle for a foot. The gait monitoring method 200 may be configured to use one or more pose parameters to determine the AP distance, and to use the AP distance to detect or identify a gait event.

[0075] In another example, the walking direction $\vec{v}_{dir}$ may be computed as the normalized velocity of the lower limbs. Using $\vec{d}_{step}$ to represent a distance vector between the feet, spatial gait parameters can be computed as follows:

$$d_{steplength} = |\vec{d}_{step} \cdot \vec{v}_{dir}| \quad (24)$$

$$d_{stepwidth} = \sqrt{\|\vec{d}_{step}\|^2 - d_{steplength}^2} \quad (25)$$

[0076] Similarly, the stride time, the stance time, and the swing time can be determined based on the timing of the heel-strike and the toe-off events. The stride time refers to the time taken by the subject to take one stride (from toe-off to heel-strike of the same foot). The stance time refers to a length of time between successive strides, during which both feet are in contact with the ground. Alternatively, the stance time refers to a length of time between a heel-strike of one foot and the toe-off of the other foot of the subject. The swing time refers to a length of time when a foot is in the air, as opposed to the stance time. A detection window of about 19 RGB-D frames was used to capture the local maxima and the local minima. The latency (between the actual occurrence of a gait event and the detection/determination of the gait event by the controller **102**) was only approximately 0.4 seconds if the gait analysis is performed in real time. The effect of noise can be addressed by considering the frames valid only if the AP-distance between the frames exceed a threshold Δd_{min} .

[0077] A proposed system based on an embodiment of the gait monitoring method **200** was tested against a marker-based motion capture system in a series of trials involving different types of walking. The subjects were required to perform a series of trials A to E bare-footed, with infrared-red markers disposed on the lower limbs (for the marker-based system). Trials A involved a static trial in which the subject stood or held a fixed stance for at least 5 seconds, with both feet oriented forward. Trials B involved an overground walking (OW)

trial in which the subject walked a 7-meter straight walking path, with a prototype robot following behind the subject at a relatively fixed distance. The walking speed during the OW trials was 0.78 ± 0.11 meters per second. Trials C involved three rounds in which the subject walked on a treadmill at a normal speed (approximately 1.0 meters per second) for 30 seconds. Trials D involved three rounds in which the subject walked on the treadmill at a low speed (approximately 0.4 meters per second) for 30 seconds. Trials E involved three rounds in which the subject walked on the treadmill at a relatively high speed (approximately 1.3 meters per second) for 30 seconds. The camera 106 used for the proposed system was an RGB-D camera Intel RealSense D415 configured at 50 frames per second. In the overground walking (OW) trial, the camera was disposed on the prototype robot and positioned about 10 centimeters (camera height) above the ground. The marker-based system used a lower body marker set with a Miquis M3 (2MP) motion capture system from Qualisys of Gothenburg, Sweden. For the sake of comparing the proposed system with the marker-based system, two types of pose errors (rotational error and translational error) were generated for each frame of the recordings used. The rotational error is defined as the minimum angle between the orientation measured by the marker-based system and the proposed system. The translational error is defined as the minimum distance between corresponding measurements from the marker-based system and the proposed system.

[0078] Since the marker-based system does not provide temporal data, the gait events in the marker-based system had to be manually labelled with the timing, and the results were the input into an additional third-party software (Visual3D Professional version 6 available from C-Motion Inc. of Maryland, U.S.A.) to generate spatio-temporal parameters of the gait for each step, from whence the gait cycles are determined.

[0079] With the proposed system, manual labelling is not required. The controller 102 of the proposed system may be configured to detect a gait cycle by finding two consecutive HS events between which there exists a TO event. The gait cycles detected by the proposed system and the gait cycles based on the marker-based system were compared to determine the precision, recall, and F1 score. For each gait cycle detected, the corresponding step length, step width, cycle time, stance time, and swing time were compared against that based on the marker-based system. Time discrepancies in the detection of the gait events were also analyzed.

[0080] The pose errors were analyzed in two scenarios: static and dynamic. The static pose errors were determined based on the data collected from the series under trials A. The dynamic pose errors were determined based on the data collected from trials B to E. Relative errors are expected to be related to the spatial-gait parameters, which are defined as the difference in pose between the left foot and the right foot of the subject. The results are tabulated in Tables 1A to 1C below. Table 2 tabulates the detection statistics of the gait cycle for trials B to E. Table 3 tabulates the heel-strike detection errors and the toe-off detection errors. Table 4 tabulates the gait spatial-parameters errors for trials B to E. Table 5 tabulates the gait temporal-parameters errors for trials B to E.

Table 1A. Pose Errors for the Right Foot

Trials	Description	Rotational Error (degrees)	Translational Error (millimeters)
A	Static	12.45±6.30	13.79±4.41
B	Overground Walking	26.71±15.89	62.25±52.65
C	Treadmill 0.4 m/s	17.86±9.07	33.09±27.26
D	Treadmill 1.0 m/s	28.84±22.49	70.59±62.20
E	Treadmill 1.3 m/s	37.30±30.26	98.54±81.68

Table 1B. Pose Errors for the Left Foot

Trials	Description	Rotational Error (degrees)	Translational Error (millimeters)
A	Static	11.35±9.01	11.31±6.98
B	Overground Walking	19.69±13.94	63.16±61.93
C	Treadmill 0.4 m/s	19.08±7.61	29.51±19.94
D	Treadmill 1.0 m/s	24.95±17.91	66.09±63.16
E	Treadmill 1.3 m/s	40.92±39.79	95.95±82.08

Table 1C. Relative Pose Errors between the Left Foot and the Right Foot

Trials	Description	Rotational Error (degrees)	Translational Error (millimeters)
A	Static	5.31±3.91	11.32±9.24
B	Overground Walking	9.16±7.52	26.38±35.79
C	Treadmill 0.4 m/s	6.84±4.86	14.59±11.61
D	Treadmill 1.0 m/s	10.51±8.85	28.73±30.95
E	Treadmill 1.3 m/s	17.35±24.42	46.85±59.96

Table 2. Detection Statistics of the Gait Cycle and Trial Types

Trials	True Positive	False Positive	False Negative	Precision	Recall	F1
B	283	13	4	0.956	0.986	0.971
C	456	2	28	0.996	0.942	0.968
D	705	8	0	0.989	1.000	0.994
E	866	6	1	0.993	0.999	0.996

Table 3. Heel-Strike and Toe-Off Detection Errors

Trials	Description	Heel-Strike (milli-seconds)	Toe-Off (milli-seconds)
B	Overground Walking	62.70±51.89	31.15±44.01
C	Treadmill 0.4 m/s	76.47±37.46	33.98±75.69
D	Treadmill 1.0 m/s	43.15±25.52	30.71±24.25
E	Treadmill 1.3 m/s	39.88±28.19	37.48±25.26

Table 4. Gait Spatial-Parameters Errors for Different Trials

Trials	Description	Step Length (millimeters)	Step Width (millimeters)
B	Overground Walking	60.67±41.63	53.66±38.78
C	Treadmill 0.4 m/s	34.82±27.67	32.54±20.75
D	Treadmill 1.0 m/s	53.76±44.80	34.19±24.28
E	Treadmill 1.3 m/s	80.67±72.76	41.48±38.86

Table 5. Gait Temporal-Parameters Errors for Different Trials

Trials	Description	Cycle Time (milli-seconds)	Stance Time (milli-seconds)	Swing Time (milli-seconds)
B	Overground Walking	35.12±29.13	69.78±38.63	69.98±42.19
C	Treadmill 0.4 m/s	37.72±79.78	72.23±87.75	67.81±46.20
D	Treadmill 1.0 m/s	30.26±26.31	66.98±36.05	66.36±36.55
E	Treadmill 1.3 m/s	36.55±38.70	66.82±39.25	68.12±45.11

[0081] In general, the trials demonstrated that that the proposed system delivers results comparable to those of a marker-based system, but advantageously does away with the need for manual labelling, special hardware (e.g., markers and additional cameras), and additional third-party software.

[0082] Referring to Tables 1A to 1B, the static pose errors for rotational errors are found to be less than 13 degrees and those for translational errors are found to be less than 14

millimeters. For the dynamic trials involving walking, for a gait speed of 0.4 meters per second, the pose errors are less than 20 degrees for rotational errors and less than 35 millimeters for translation errors, respectively. Trials B (overground walking at a speed of about 0.78 ± 0.11 meters per second) and trial D (walking on a treadmill at 1.0 meters per second) have comparable performance. The relative errors are smaller than the individual gait errors for all cases except in the case of static translation errors.

[0083] Table 2 shows the detection rates of each gait cycle. The table reflect the fact that overground trials are more difficult to track (lower precision) than treadmill trials as there are more irregularities in the terrain traversed, etc. Nevertheless, the proposed system is able to achieve an F1 score of more than 95% in all trials, demonstrating the viability of the proposed system for use outside the treadmill or laboratory setting.

[0084] The experimental step length and step width errors at 0.4 meters per second are 12 millimeters and 29 millimeters respectively according to the proposed system, which are comparable to the figures of 16 millimeters and 25 millimeters reported elsewhere.

[0085] The average computational speed of the present method **200** is about 23.75 frames per second, which is significantly faster or computationally more efficient than conventional particle filtering methods which take about 14 to 19 seconds of processing time per frame. The time taken to perform object pixel identification and cost optimization are about 20.86 milli-seconds and 21.42 milli-seconds, respectively. The method of determining pose-information and/or gait parameters can be run on a single CPU (central processing unit) thread. This means that the robot **100** and/or method **200** are suitable for real-time applications, including via implementation in various rehabilitation and/or assistive devices.

[0086] As described above, embodiments of the present disclosure have experimentally established the ability to overcome several technical difficulties. For example, the method **200** is able to correctly acquire pose-related information, even if one of the limbs is occluded by the other (a common occurrence when the subject is making a turn). In other situations, the two limbs may come into contact with one another such that the point cloud would appear as one big cluster. The present method **200** has demonstrated that it is able to correctly distinguish between the data points of the different limbs, i.e., it is able to avoid assigning the pixels to the wrong limb.

[0087] The present method enables tracking of the subject's feet regardless of footwear and regardless of feet size. The subject is not required to wear special attire or shoes, or have tags attached to the body or any part of the attire. This gets around the need to rely on a marker-based motion capture system with multiple infra-red cameras. The subject may walk around bare-footed without being encumbered by tags, making the method suitable for use outside the laboratory and without the physical presence of a trained laboratory personnel in the same room as the subject. Advantageously, the present method is contactless and can be implemented without supervision by a trained healthcare professional. In other words, the method is suitable for use even when the subject is alone at home.

[0088] It is also not required to prepare three-dimensional models or pre-train neural networks prior to use. Conventionally, it can be challenging to relocate the foot once the algorithm has lost track of it. Human intervention or reliance on additional image sources (such as color images) were previously required to relocate the foot. Advantageously, the present method has demonstrated experimentally that it is able to overcome these challenges.

[0089] The present method does not require a new model to be trained for each subject. There are only a few shape parameters required by the present method **200**. In some embodiments, the controller may be configured to optimize the shape parameters **102** without manual input. In some other parameters, the shape parameters can be measured and input to the robot **100**.

[0090] Embodiments of the present disclosure overcomes the difficulties inherent in motion capture analysis of feet. Conventionally, it is difficult to relocate a foot once the algorithm loses track of the foot image, such that human intervention is required. In some other cases, quality color images are used to extract additional information for the purpose of relocating the foot. Overall, the performance of the present method improves over conventional methods, considering that feet are well-known to be difficult to track as they may seem to form part of the ground or the foot of one limb may be hidden/partially hidden by the other limbs in the course of moving, etc.

[0091] Traditionally, it is deemed more challenging to perform gait monitoring when subjects are walking than when they are running. One reason is that running involves larger

and more distinct pose and gait phases. Advantageously, embodiments of the robot and the marker-less gait monitoring method disclosed herein are found to be particularly suited for use with elderly subjects or patients who can only move very slowly. For example, chronic stroke patients usually have gait speeds between 0.29 meters per second and 0.6 meters per second. In fact, stroke survivors who walk slower than 0.4 meters per second can only ambulate at home. In other words, the robot and the method disclosed herein are configured to perform marker-less gait monitoring in cases where the gait speed of the subject is characteristic of a walking gait. Having a robot that can monitor their movements at home on a real-time basis can provide timely fall-preventive support. At the same time, the robot would not interfere with the subject's movement (unlike the case with markers) as the robot is configured to be positioned or to position itself out of the way behind the subject.

[0092] While the gait monitoring method **200** is intended to work with a back view of the limbs, it is also operable when the limbs are viewed from the sides. Thus, the gait monitoring method **200** is sufficiently robust for actual applications where the subject **90** (e.g., Fig. 2) may not always walk with the feet in the same orientation (unlike in a laboratory setting where subjects are essentially confined to walk along a designated straight track or on a treadmill).

[0093] Various embodiments of the present disclosure are useful in a wide range of applications (**290**), including but not limited to methods and devices for gait rehabilitation assistance. For example, the robot **100** may be configured to track and monitor the subject **90** over a relatively longer period of time (e.g., over a period of months). The robot **100** is able to do so in a relatively unobtrusive manner in the home of the subject **90**, outside the laboratory setting and without the physical presence of a healthcare professional (Fig. 8A). Gait monitoring of the subject **90** can be carried out as the subject **90** ambulates "over ground" as he/she carries out daily activities. The gait and pose information acquired is a better reflection of how the subject **90** performs in an actual environment, as opposed to the relatively artificial set-up in a kinematics laboratory. The gait and pose information generated and collected over time can be used to inform a more objective and data-based gait and balance analysis for the subject. The robot **100** may be configured to serve as a safety monitor for the subject **90**. In some applications, the robot **100** may be configured to use the pose and gait information to predict or anticipate a potential fall event, and

responsive to such a prediction, render support to the subject (e.g., by way of a brace **130**). Timely fall preventive measures can be triggered by/in the robot **100** as the gait monitoring method **200** is computationally efficient enough for real-time gait monitoring. In some applications, the motion of the lower limbs may help to predict the intention of the user, allowing more compliant human-robot interaction for the gait rehabilitation and assistive robot **100**.

[0094] All examples described herein, whether of apparatus, methods, materials, or products, are presented for the purpose of illustration and to aid understanding, and are not intended to be limiting or exhaustive. Various changes and modifications may be made by one of ordinary skill in the art without departing from the scope of the invention as claimed.

CLAIMS

1. A marker-less gait monitoring method for use with a subject, comprising:
 - acquiring a point cloud of data points based on a depth image of at least one of the subject's lower limbs;
 - allocating selected ones of the data points to one of a foot model and a shank model;
 - and
 - determining one or more pose parameters of the foot model based on optimizing parameters of a foot-shank model, wherein the foot-shank model includes the foot model and the shank model as adjoining parts that are rotatable relative to one another about a common origin.
2. The marker-less gait monitoring method according to claim 1, wherein the depth image is a back view image of the subject.
3. The marker-less gait monitoring method according to claim 1 or claim 2, further comprising:
 - acquiring the depth image, wherein the depth image is one of a plurality of depth images captured by at least one camera disposed on a robot, and wherein the robot in a tracking mode is configured to be positioned or to position itself behind the subject.
4. The marker-less gait monitoring method according to claim 2 or claim 3, wherein the robot is configured to keep the at least one camera within a range of trailing distances relative to the subject, and wherein the range of trailing distances is determined such that the depth image includes a foot and a shank of the at least one of the subject's lower limbs.
5. The marker-less gait monitoring method according to any one of claims 1 to 4, wherein the depth image is a back view image of one or both feet and shanks of the subject.
6. The marker-less gait monitoring method according to any one of claims 1 to 5, further comprising:

from the point cloud, identifying the data points corresponding to at least one of a right lower limb and a left lower limb of the subject.

7. The marker-less gait monitoring method according to any one of claims 1 to 6, comprising:

 sorting the data points into a plurality of clusters by respective depth values of the data points;

 responsive to finding two groups of clusters, allocating one of the two groups of clusters to a right lower limb of the subject and another of the two groups of clusters to a left lower limb of the subject; and

 responsive to finding only one group of clusters, allocating some of the data points to at least one lower limb of the subject.

8. The marker-less gait monitoring method according to any one of claims 1 to 7, further comprising: prior to allocating the selected ones of the data points to one of the foot model and the shank model, identifying the data points corresponding to each of a right lower limb and a left lower limb of the subject.

9. The marker-less gait monitoring method according to any one of claims 1 to 8, wherein the allocating selected ones of the data points to one of a foot model and a shank model comprises:

 using an expansion-segmentation method to find a segmentation line separating the data points of the foot model and the data points of the shank model, wherein the segmentation line is defined relative to a lowest point in a cluster of data points of the point cloud.

10. The marker-less gait monitoring method according to claims 1 to 7, wherein the allocating of some of the data points to the at least one lower limb of the subject is based on vertices of a respective projected foot-shank model, the respective project foot-shank model being associated with an earlier depth image.

11. The marker-less gait monitoring method according to claim 10, wherein the allocating of some of the data points is based on a contour-segmentation method in which two convex hulls are defined by the respective vertices of a foot model part of the respective projected foot-shank model and a shank model part of the respective projected foot-shank model.
12. The marker-less gait monitoring method according to any one of claims 1 to 11, wherein the optimizing parameters of the foot-shank model further comprises determining one or more shape parameters of the foot-shank model.
13. The marker-less gait monitoring method according to claim 12, wherein the determining of the one or more pose parameters and/or one or more shape parameters comprises optimizing a foot-shank cost function, and wherein the foot-shank cost function comprises a foot fitting cost and a shank fitting cost.
14. The marker-less gait monitoring method according to claim 13, wherein the foot-shank cost function further comprises a shape penalty to prevent oversizing of the foot model.
15. The marker-less gait monitoring method according to claim 13 or 14, wherein the foot-shank cost function further comprises an orientation penalty to penalize a non-alignment of the shank model relative to a vertical axis.
16. The marker-less gait monitoring method according to any one of claims 13 to 15, wherein the foot-shank cost function further comprises a foot flexing term configured to penalize a flexing movement of the foot model, and wherein the flexing movement includes any one or both of a dorsiflexion-plantarflexion of the foot and an inversion-eversion of the foot.
17. The marker-less gait monitoring method according to any one of the claims 13 to 16, wherein the foot-shank cost function further comprises a temporal penalty, and wherein the temporal penalty is based on an assumption that the subject moves relatively slowly.

18. The marker-less gait monitoring method according to any one of claims 1 to 17, wherein a gait speed of the subject is characteristic of a walking gait.

19. The marker-less gait monitoring method according to any one of claims 1 to 18, further comprising:

using the one or more pose parameters to determine an anterior-posterior distance;

and

using the anterior-posterior distance to identify a gait event.

20. The marker-less gait monitoring method according to any one of claims 1 to 19, further comprising:

using the one or more pose parameters and/or one or more shape parameters to generate gait and pose information.

21. A method of gait rehabilitation assistance for a subject, comprising:

tracking the subject over a length of time; and

performing a gait and balance analysis for the subject using the gait and pose information of the subject, wherein the gait and pose information is generated according to claim 20.

22. A robot for use with a subject, the robot comprising:

a camera;

a plurality of wheels, the plurality of wheels being coupled with the camera such that the robot in a tracking mode is configured to track the subject with the camera configured to concurrently capture back view depth images of the subject; and

a controller operably coupled with the camera, the controller being configured to acquire image data from the camera, wherein the controller is configured to use the image data to perform the marker-less gait monitoring method according to any one of claims 1 to 20.