(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) **World Intellectual Property Organization**
International Bureau

(43) **International Publication Date**
28 October 2021 (28.10.2021)

WIPO | PCT

(10) **International Publication Number**
WO 2021/216263 A1

(51) **International Patent Classification:**
G05D 1/02 (2020.01) *G01C 22/00* (2006.01)

(21) **International Application Number:**
PCT/US2021/025294

(22) **International Filing Date:**
01 April 2021 (01.04.2021)

(25) **Filing Language:** English

(26) **Publication Language:** English

(30) **Priority Data:**

63/013,634	22 April 2020 (22.04.2020)	US
16/893,608	05 June 2020 (05.06.2020)	US

(71) **Applicant: BOSTON DYNAMICS, INC.** [US/US]; 200 Smith Street, Suite 4100, Waltham, MA 02451 (US).

(72) **Inventor: KLINGENSMITH, Matthew, Jacob**; 200 Smith Street, Suite 4100, Waltham, MA 02451 (US).

(74) **Agent: KRUEGER, Brett, A.**; Honigman LLP, 650 Trade Centre Way, Suite 200, Kalamazoo, MI 49002-0402 (US).

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, IT, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(54) **Title:** ROBOT LOCALIZATION USING VARIANCE SAMPLING

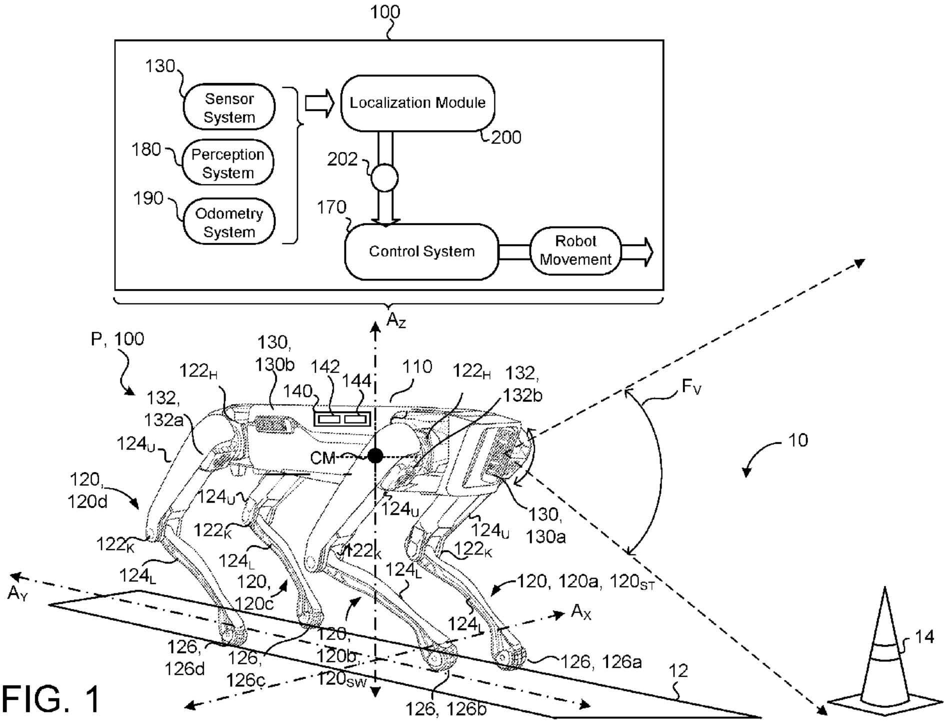

FIG. 1

(57) **Abstract:** A method (310) of localizing includes receiving odometry information (192) plotting locations (202) and sensor data (134) of an environment (10). The method includes obtaining a series of odometry information members (315), each including a respective odometry measurement at a respective time (dt). The method also includes obtaining a series of sensor data members (313), each including a respective sensor measurement at the respective time. The method also includes, for each sensor data member of the series of sensor data members, (i) determining a localization (321) at the respective time based on the respective sensor data, and (ii) determining an offset (323) of the localization relative to the odometry measurement at the respective time. The method also includes determining whether a variance (s^2offset) of the offsets determined for the localizations exceeds a

WO 2021/216263 A1

[Continued on next page]

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

Published:

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

threshold variance (s^2threshoid). When the variance among the offsets exceeds the threshold variance, a signal (204) is generated.

Robot Localization Using Variance Sampling

TECHNICAL FIELD

[0001] This disclosure relates robot localization using variance sampling for robots.

BACKGROUND

[0002] A robot is generally defined as a programmable and multifunctional system designed to navigate an environment and/or to move material, parts, tools, or specialized devices through variable programmed motions for a performance of tasks. Robots may be mobile robots that move throughout an environment (e.g., using legs, wheels, or traction based mechanisms), manipulators that are physically anchored (e.g., industrial robotic arms), or some combination of a manipulator and a mobile robot. Robots are utilized in a variety of industries including, for example, manufacturing, transportation, hazardous environments, exploration, and healthcare. As such, the ability of robots to accurately determine location within the robot environment provides additional benefits to such industries.

SUMMARY

[0003] One aspect of the disclosure provides a method for robot localization. The method includes, while a robot traverses an environment, receiving, at data processing hardware, odometry information plotting a location of the robot over time. While the robot traverses the environment, the method also includes receiving, at the data processing hardware, from at least one sensor of the robot in communication with the data processing hardware, sensor data of the environment about the robot. The method also includes obtaining, by the data processing hardware, an odometry information sample including a series of odometry information members collected by the robot while traversing the environment. Each odometry information member of the series of odometry information members includes a respective odometry measurement at a respective time. The method also includes obtaining, by the data processing hardware, a sensor data sample including a series of sensor data members collected by the robot while traversing the environment. Each sensor data member of the series of sensor data members includes a respective sensor measurement at the respective time. For each sensor data member of the series of sensor data members, the method includes determining, by the data processing hardware, a localization of the robot at the respective time based on the respective sensor data, and determining, by the data processing hardware, an offset of the localization of the robot relative to the odometry measurement at the respective time. The method also includes determining, by the data processing hardware, whether a variance of the offsets determined for the localizations exceeds a threshold variance. The method further includes, when the variance among the offsets exceeds the threshold variance, generating, by the data processing hardware, a signal indicating that the sensor data is unreliable.

[0004] Implementations of the disclosure may include one or more of the following optional features. In some implementations, the method further includes adjusting, by the data processing hardware, relative weights of the odometry information and the sensor data based on whether the offsets determined for the localizations exceed the threshold variance. Optionally, the method includes determining, by the data processing hardware, a current location of the robot using the adjusted relative weights of the odometry information and the sensor data.

[0005] In some examples, the method includes, when the variance of the offsets exceeds the threshold variance, reducing, by the data processing hardware, a weight assigned to the sensor data. In some implementations, the method includes, when the variance of the offsets does not exceed the threshold variance, increasing, by the data processing hardware, a weight assigned to the sensor data.

[0006] In some configurations, the series of sensor data members includes a reference sensor data member and a plurality of comparison sensor data members. Optionally, determining the localization includes performing a pairwise analysis on one of the comparison sensor data members associated with the respective time with the reference sensor data member. In some implementations, the variance of the offsets is a sample variance.

[0007] Another aspect of the disclosure provides a system including data processing hardware and memory hardware in communication with the data processing hardware.

The memory hardware stores instructions that, when executed on the data processing hardware, cause the data processing hardware to perform operations. The operations include, while a robot traverses an environment, receiving odometry information plotting a location of the robot over time, and receiving from at least one sensor of the robot in communication with the data processing hardware, sensor data of the environment about the robot. The operations also include obtaining an odometry information sample including a series of odometry information members collected by the robot while traversing the environment. Each odometry information member of the series of odometry information members includes a respective odometry measurement at a respective time. The operations also include obtaining a sensor data sample including a series of sensor data members collected by the robot while traversing the environment. Each sensor data member of the series of sensor data members includes a respective sensor measurement at the respective time. For each sensor data member of the series of sensor data members, the operations include determining a localization of the robot at the respective time based on the respective sensor data, and determining an offset of the localization of the robot relative to the odometry measurement at the respective time. The operations also include determining whether a variance of the offsets determined for the localizations exceeds a threshold variance. The operations also include, when the variance among the offsets exceeds the threshold variance, generating, by the data processing hardware, a signal indicating that the sensor data is unreliable. This aspect of the disclosure may include one or more of the following optional features.

[0008] In some examples, the operations further include adjusting, by the data processing hardware, relative weights of the odometry information and the sensor data based on whether the offsets determined for the localizations exceed the threshold variance. In some implementations, the operations further comprise determining a current location of the robot using the adjusted relative weights of the odometry information and the sensor data.

[0009] In some configurations, the operations further comprise, when the variance of the offsets exceeds the threshold variance, reducing a weight assigned to the sensor data.

In some examples, the operations further comprise, when the variance of the offsets does not exceed the threshold variance, increasing weight assigned to the sensor data.

[0010] In some examples, the series of sensor data members includes a reference sensor data member and a plurality of comparison sensor data members. Optionally, determining the localization includes performing a pairwise analysis on one of the comparison sensor data members associated with the respective time with the reference sensor data member. In some implementations, the variance of the offsets is a sample variance.

[0011] Another aspect of the disclosure provides a robot including a body, data processing hardware, and memory hardware in communication with the data processing hardware. The hardware stores instructions that, when executed on the data processing hardware, cause the data processing hardware to perform operations. The operations include, while a robot traverses an environment, receiving odometry information plotting a location of the robot over time, and receiving from at least one sensor of the robot in communication with the data processing hardware, sensor data of the environment about the robot. The operations also include obtaining an odometry information sample including a series of odometry information members collected by the robot while traversing the environment. Each odometry information member of the series of odometry information members includes a respective odometry measurement at a respective time. The operations also include obtaining a sensor data sample including a series of sensor data members collected by the robot while traversing the environment. Each sensor data member of the series of sensor data members includes a respective sensor measurement at the respective time. For each sensor data member of the series of sensor data members, the operations include determining a localization of the robot at the respective time based on the respective sensor data, and determining an offset of the localization of the robot relative to the odometry measurement at the respective time. The operations also include determining whether a variance of the offsets determined for the localizations exceeds a threshold variance. The operations also include, when the variance among the offsets exceeds the threshold variance, generating, by the data processing hardware, a signal indicating that the sensor data is unreliable. This aspect of the disclosure may include one or more of the following optional features.

[0012] In some examples, the operations further include adjusting, by the data processing hardware, relative weights of the odometry information and the sensor data based on whether the offsets determined for the localizations exceed the threshold variance. In some implementations, the operations further comprise determining a current location of the robot using the adjusted relative weights of the odometry information and the sensor data.

[0013] In some configurations, the operations further comprise, when the variance of the offsets exceeds the threshold variance, reducing a weight assigned to the sensor data. In some examples, the operations further comprise, when the variance of the offsets does not exceed the threshold variance, increasing weight assigned to the sensor data.

[0014] In some examples, the series of sensor data members includes a reference sensor data member and a plurality of comparison sensor data members. Optionally, determining the localization includes performing a pairwise analysis on one of the comparison sensor data members associated with the respective time with the reference sensor data member. In some implementations, the variance of the offsets is a sample variance.

[0015] The details of one or more implementations of the disclosure are set forth in the accompanying drawings and the description below. Other aspects, features, and advantages will be apparent from the description and drawings, and from the claims.

DESCRIPTION OF DRAWINGS

[0016] FIG. 1 is a schematic view of an example robot within an environment.

[0017] FIG. 2 is a schematic view of example systems for the robot of FIG. 1.

[0018] FIG. 3 is a schematic view of a localization module and an example arrangement of operations executed by the localization module for determining a location of a robot.

[0019] FIG. 4 is a plot showing sensor location data for a robot represented with respect to a period of time.

[0020] FIG. 5A is a plot showing the sensor location data of FIG. 4 represented with respect to a ground truth trajectory.

[0021] FIG. 5B is a plot showing localizations based on the sensor location data of FIG. 5A represented with respect to the ground truth trajectory.

[0022] FIG. 6A is a plot showing offsets of the localizations of FIG. 5B relative to corresponding odometry location information.

[0023] FIG. 6B is a plot showing offsets of the localizations of FIG. 5B relative to an origin.

[0024] FIGS. 7A and 7B are plots showing example distributions of values of the offsets relative to the corresponding odometry location information.

[0025] FIG. 8 is a schematic view of an example computing device that may be used to implement the systems and methods described herein.

[0026] Like reference symbols in the various drawings indicate like elements.

DETAILED DESCRIPTION

[0027] Mobile robots are typically programmed to navigate an environment by manually driving the robot around an environment and building a map of the environment using sensor systems of the robot. Based on the map, the robot must be capable of autonomously navigating the environment to achieve certain tasks (such as taking additional sensor measurements, or manipulating objects). During navigation, the robot my continuously conduct a localization operation to determine the actual (i.e. local) position of the robot within the environment.

[0028] Autonomous robots navigating and localizing in real environments using camera or laser sensors face many challenges -- sensor noise and drift, misleading or view-dependent features, and ambiguous geometry, etc. Accordingly, the robot must know whether to trust data received from the sensor systems a priori. If a robot does not know what sensor data to trust, the robot can become lost or confused by the misleading data. Misleading data from the sensor system is predominantly caused by environmental conditions (such as the shininess of surrounding surfaces, the distance to walls, the presence or absence of strong textures or reference objects in the environment, etc.) rather than properties intrinsic to the sensor itself. Environmental conditions that lead to misleading sensor data may be referred to as "feature deserts," as the environment lacks sufficient reference features for the sensor system to rely upon in determining the location of the robot.

[0029] Traditional approaches to determining whether sensor data is reliable have relied on modelling sensor properties directly, and applying this forward model to live sensor data to determine whether the data can be trusted. However, not all sensor error can be efficiently modeled. Other approaches consider learning the sensor model, but these approaches rely on extensive training data and are not easy to inspect.

[0030] The present disclosure provides a system and process for a robot to determine when the sensor data associated with the surrounding environment is reliable, and therefore, can be trusted by the robot for navigating and performing localization in the environment. In particular, implementations herein are directed toward executing a process for determining sensor data reliability online (e.g., while the robot is making a map of its environment) to permit the robot to provide a user feedback indicating the quality of the sensor data so that the user can take immediate corrective action (if necessary). For example, responsive to user feedback indicating that the quality of the sensor data is poor, the user may move the robot closer to walls and/or add additional reference features in the environment when building the map of the environment. As will become apparent, the techniques disclosed in the present disclosure eliminate or reduce the need for an expensive offline system for analyzing sensor data quality, which analyzes robot sensor data after-the-fact to make a prediction about the quality of the sensor data and the presence of feature deserts.

[0031] Referring to FIG. 1, the robot 100 includes a body 110 with locomotion based structures such as legs 120a–d coupled to the body 110 that enable the robot 100 to move about an environment 10. In some examples, each leg 120 is an articulable structure such that one or more joints 122 permit members 124 of the leg 120 to move. For instance, each leg 120 includes a hip joint 122_H coupling an upper member 124_U of the leg 120 to the body 110 and a knee joint 122_K coupling the upper member 124_U of the leg 120 to a lower member 124_L of the leg 120. Although FIG. 1 depicts a quadruped robot with four legs 120a–d, the robot 100 may include any number of legs or locomotive based structures (e.g., a biped or humanoid robot with two legs) that provide a means to traverse the terrain within the environment 10.

[0032] In order to traverse the terrain, each leg 120 has a distal end 126 (also referred to as a foot 126 of the robot 100) that contacts a surface of the terrain (i.e., a traction surface). In other words, the distal end 126 of the leg 120 is the end of the leg 120 used by the robot 100 to pivot, plant, or generally provide traction during movement of the robot 100. For example, the distal end 126 of a leg 120 corresponds to a foot of the robot 100. In some examples, though not shown, the distal end 126 of the leg 120 includes an ankle joint such that the distal end 126 is articulable with respect to the lower member 124_L of the leg 120.

[0033] The robot 100 has a vertical gravitational axis (e.g., shown as a Z-direction axis A_Z) along a direction of gravity, and a center of mass CM, which is a point where the weighted relative position of the distributed mass of the robot 100 sums to zero. The robot 100 further has a pose P based on the center of mass CM relative to the vertical gravitational axis A_Z (i.e., the fixed reference frame with respect to gravity) to define a particular attitude or stance assumed by the robot 100. The attitude of the robot 100 can be defined by an orientation or an angular position of the robot 100 in space. Movement by the legs 120 relative to the body 110 alters the pose P of the robot 100 (i.e., the combination of the position of the CM of the robot and the attitude or orientation of the robot 100). Here, a height generally refers to a distance along the z-direction. A ground plane (also referred to as a transverse plane) spans the X-Y plane by extending in directions of the x-direction axis A_X and the y-direction axis A_Y. The ground plane refers to a ground surface 12 where distal ends 126 of the legs 120 of the robot 100 may generate traction to help the robot 100 move about the environment 10. Another anatomical plane of the robot 100 is the frontal plane that extends across the body 110 of the robot 100 (e.g., from a left side of the robot 100 with a first leg 120a to a right side of the robot 100 with a second leg 120b). The frontal plane spans the X-Z plane by extending in directions of the x-direction axis A_X and the z-direction axis A_Z.

[0034] When a legged-robot 100 moves about the environment 10, the legs 120 of the robot undergo a gait cycle. Generally, a gait cycle begins when a leg 120 touches down or contacts a ground surface 12 and ends when that same leg 120 once again contacts the ground surface 12. The gait cycle may predominantly be divided into two phases, a swing phase and a stance phase. During the swing phase, a leg 120 performs (i) lift-off from the ground surface 12 (also sometimes referred to as toe-off and the transition between the stance phase and swing phase), (ii) flexion at a knee joint 122_K of the leg 120, (iii) extension of the knee joint 122_K of the leg 120, and (iv) touchdown back to the ground surface 12. Here, a leg 120 in the swing phase is referred to as a swing leg 120_{SW}. As the swing leg 120_{SW} proceeds through the movement of the swing phase 120_{SW}, one or more of the other legs 120 performs the stance phase. The stance phase refers to a period of time where a distal end 126 (e.g., a foot) of the leg 120 is on the ground surface 12. During the stance phase a leg 120 performs (i) initial ground surface contact which triggers a transition from the swing phase to the stance phase, (ii) loading response where the leg 120 dampens ground surface contact, (iii) mid-stance support for when the contralateral leg (i.e., the swing leg 120_{SW}) lifts-off and swings to a balanced position (about halfway through the swing phase), and (iv) terminal-stance support from when the robot's center of mass CM is over the leg 120 until the contralateral leg 120 touches down to the ground surface 12. Here, a leg 120 in the stance phase is referred to as a stance leg 120_{ST}.

[0035] In order to maneuver about the environment 10, the robot 100 includes a sensor system 130, 130a–b with one or more sensors 132, 132a–n (e.g., shown as a first sensor 132, 132a and a second sensor 132, 132b). The sensors 132 may include vision/image sensors 132, inertial sensors (e.g., an inertial measurement unit (IMU)), force sensors, and/or kinematic sensors. Some examples of image sensors 132 include a camera such as a stereo camera, a scanning light-detection and ranging (LIDAR) sensor, or a scanning laser-detection and ranging (LADAR) sensor. In some examples, the image sensors 132 have a corresponding field(s) of view F_V defining a sensing range or region corresponding to the image sensors 132_I. For instance, FIG. 1 depicts a field of a view F_V for the robot 100. Each image sensor 132 may be pivotable and/or rotatable such that the image sensor 132 may, for example, change the field of view F_V about one or more axis (e.g., an x-axis, a y-axis, or a z-axis in relation to a ground plane).

[0036] In some implementations, the sensor system 130 includes position sensor(s) 132 coupled to a joint 122. In some examples, these position sensors 132 couple to a motor that operates a joint 122 of the robot 100 (e.g., sensors 132, 132a–b). Here, these sensors 132 generate joint dynamics 134, 134_{JD} in the form of joint-based sensor data 134. Joint dynamics 134_{JD} collected as joint-based sensor data 134 may include joint angles (e.g., an upper member 124_U relative to a lower member 124_L), joint speed (e.g., joint angular velocity or joint angular acceleration), and/or joint torques experienced at a joint 122 (also referred to as joint forces). Here, joint-based sensor data 134 generated by one or more sensors 132 may be raw sensor data, data that is further processed to form different types of joint dynamics 134_{JD}, or some combination of both. For instance, a sensor 132 measures joint position (or a position of member(s) 124 coupled at a joint 122) and systems of the robot 100 perform further processing to derive velocity and/or acceleration from the positional data. In other examples, a sensor 132 is configured to measure velocity and/or acceleration directly.

[0037] When surveying a field of view F_V with a sensor 132, the sensor system 130 generates sensor data 134 (also referred to as image data) corresponding to the field of view F_V. In some examples, the sensor data 134 is image data 134 that corresponds to a three-dimensional volumetric point cloud generated by a three-dimensional volumetric image sensor 132. The image data 134 may be based on reference features 14 situated within the environment 10 that can be easily distinguished and observed by the sensors 132. Additionally or alternatively, when the robot 100 is maneuvering about the environment 10, the sensor system 130 gathers pose data for the robot 100 that includes inertial measurement data 134 (e.g., measured by an IMU). In some examples, the pose data includes kinematic data and/or orientation data about the robot 100, for instance, kinematic data and/or orientation data about joints 122 or other portions of a leg 120 of the robot 100. With the image data 134 and the inertial measurement data 134, a perception system 180 of the robot 100 may generate maps 182 for the terrain about the environment 10.

[0038] While the robot 100 maneuvers about the environment 10, the sensor system 130 gathers sensor data 134 relating to the terrain of the environment 10 and/or structure of the robot 100 (e.g., joint dynamics and/or odometry of the robot 100). For instance, FIG. 1 depicts the sensor system 130 gathering sensor data 134 about a room of the environment 10 of the robot 100. As the sensor system 130 gathers sensor data 134, a computing system 140 is configured to store, process, and/or communicate the sensor data 134 to various systems of the robot 100 (e.g., a control system 170, the perception system 180, an odometry system 190, and/or a localization module 200). In order to perform computing tasks related to the sensor data 134, the computing system 140 of the robot 100 includes data processing hardware 142 and memory hardware 144. The data processing hardware 142 is configured to execute instructions stored in the memory hardware 144 to perform computing tasks related to activities (e.g., movement and/or movement based activities) for the robot 100. Generally speaking, the computing system 140 refers to one or more locations of data processing hardware 142 and/or memory hardware 144.

[0039] In some examples, the computing system 140 is a local system located on the robot 100. When located on the robot 100, the computing system 140 may be centralized (i.e., in a single location/area on the robot 100, for example, the body 110 of the robot 100), decentralized (i.e., located at various locations about the robot 100), or a hybrid combination of both (e.g., where a majority of centralized hardware and a minority of decentralized hardware). To illustrate some differences, a decentralized computing system 140 may allow processing to occur at an activity location (e.g., at motor that moves a joint of a leg 120) while a centralized computing system 140 may allow for a central processing hub that communicates to systems located at various positions on the robot 100 (e.g., communicate to the motor that moves the joint of the leg 120).

[0040] Additionally or alternatively, the computing system 140 includes computing resources that are located remotely from the robot 100. For instance, as shown in FIG. 2, the computing system 140 may communicate via a network 150 with a remote system

160 (e.g., a remote computer/server or a cloud-based environment). Much like the computing system 140, the remote system 160 includes remote computing resources such as remote data processing hardware 162 and remote memory hardware 164. Here, sensor data 134 or other processed data (e.g., data processing locally by the computing system 140) may be stored in the remote system 160 and may be accessible to the computing system 140. In some examples, the computing system 140 is configured to utilize the remote resources 162, 164 as extensions of the computing resources 142, 144 such that resources of the computing system 140 may reside on resources of the remote system 160.

[0041] In some implementations, as shown in FIGS. 1 and 2, the robot 100 includes the control system 170 and the perception system 180. The perception system 180 is configured to receive the sensor data 134 from the sensor system 130 and process the sensor data 134 to generate maps 182. With the maps 182 generated by the perception system 180, the perception system 180 may communicate the maps 182 to the control system 170 in order perform controlled actions for the robot 100, such as moving the robot 100 about the environment 10. In some examples, by having the perception system 180 separate from, yet in communication with the control system 170, the control system 170 focuses on controlling the robot 100 while the perception system 180 separately focuses on interpreting the sensor data 134 gathered by the sensor system 130. For instance, these systems 170, 180 execute their processing in parallel to ensure accurate, fluid movement of the robot 100 in an environment 10.

[0042] In some examples, the control system 170 includes at least one controller 172, a path generator 174, a step planner 176, and a body planner 178. The control system 170 may be configured to communicate with at least one sensor system 130 and any other system of the robot 100 (e.g., the perception system 180, the odometry system 190, and/or the localization module 200). The control system 170 performs operations and other functions using hardware 140. The controller 172 is configured to control movement of the robot 100 to traverse about the environment 10 based on input or feedback from the systems of the robot 100 (e.g., the control system 170, the perception system 180, the odometry system 190, and/or the localization module 200). This may include movement between poses and/or behaviors of the robot 100. For example, the controller 172 controls different footstep patterns, leg patterns, body movement patterns, or vision system sensing patterns.

[0043] In some examples, the controller 172 includes a plurality of controllers 172 where each of the controllers 172 has a fixed cadence. A fixed cadence refers to a fixed timing for a step or swing phase of a leg 120. For example, each controller 172 instructs the robot 100 to move the legs 120 (e.g., take a step) at a particular frequency (e.g., step every 250 milliseconds, 350 milliseconds, etc.). With a plurality of controllers 172 where each controller 172 has a fixed cadence, the robot 100 can experience variable timing by switching between controllers 172. In some implementations, the robot 100 continuously switches/selects fixed cadence controllers 172 (e.g., re-selects a controller 172 every 3 milliseconds) as the robot 100 traverses the environment 10.

[0044] Referring to FIG. 2, the path generator 174 is configured to determine horizontal motion for the robot 100. For instance, the horizontal motion refers to translation (i.e., movement in the X-Y plane) and/or yaw (i.e., rotation about the Z-direction axis Az) of the robot 100. The path generator 174 determines obstacles within the environment 10 about the robot 100 based on the sensor data 134. The path generator 174 communicates the obstacles to the step planner 176 such that the step planner 176 may identify foot placements for legs 120 of the robot 100 (e.g., locations to place the distal ends 126 of the legs 120 of the robot 100). The step planner 176 generates the foot placements (i.e., touchdown locations for the foot 126) for each step using inputs from the perceptions system 180 (e.g., map(s) 182) and the localization module 200. The body planner 178, much like the step planner 176, receives inputs from the perceptions system 180 (e.g., map(s) 182). Generally speaking, the body planner 178 is configured to adjust dynamics of the body 110 of the robot 100 (e.g., rotation, such as pitch or yaw and/or height of COM) to successfully move about the environment 10.

[0045] The perception system 180 of the robot 100 is configured to help the robot 100 move more precisely in a terrain with various obstacles. As the sensors 132 collect sensor data 134 for the space about the robot 100 (i.e., the robot's environment 10), the perception system 180 uses the sensor data 134 to form one or more maps 182 for the environment 10. Once the perception system 180 generates a map 182, the perception system 180 is also configured to add information to the map 182 (e.g., by projecting sensor data 134 on a preexisting map) and/or to remove information from the map 182.

[0046] Referring further to FIG. 2, the odometry system 190 is configured to measure where the robot 100 is located within a world reference frame (e.g., the environment 10) and how fast the robot 100 is moving in that world reference frame. In other words, the odometry system 190 generates odometry information 192 as one or more estimations (e.g., measurements) for a characteristic of the robot 100 relative to a world reference frame. In some examples, the odometry system 190 receives sensor data 134 from a sensor 132 such as an IMU (e.g., accelerometer(s) and/or gyro(s)). With the sensor data 134, the odometry system 190 may generate odometry information 192 based on an assumption that when a distal end 126 of a leg 120 is in contact with the ground surface 12 and not slipping, the distal end 126 is stationary. By combining this assumption with the sensor data 134, the odometry system 190 generates odometry information 192 regarding robot motion relative to the world reference frame (e.g., the environment 10). In other words, the odometry system 190 accounts for kinematics and inertial measurements to produce estimations about the robot 100 with respect to the world reference frame.

[0047] As shown in FIG. 2, the localization module 200 is configured to evaluate the sensor data 134, the map 182, and the odometry information 192 received from the systems (e.g., the sensor system 130, the perception system 180, the odometry system 190, and/or the localization module 200) to determine location data 202 indicating a current location of the robot 100 within the environment 10. For example, the control system 170 of the robot 100 may use the odometry information 192 (e.g., speed, direction) to estimate a current location of the robot 100 within the environment 10 based on movements of the robot 100 relative to the prior location data 202_{prior}. However, the odometry information 192 may include noise associated with events (e.g., slips, obstructions) and drift (e.g., measurement error), which result in errors in the estimated current location data 202. Accordingly, the localization module 200 is configured to periodically correct the estimated location of the robot 100 based on the image data 134.

Particularly, the localization module 200 compares image data 134 received from the sensor system 130 to the map 182 generated by the perception system 180 to determine the actual location of the robot 100 relative to the map 182. The localization module 200 then adjusts the current location data 202 (i.e., the odometry information 192) of the robot 100 based on the localization. The current location data 202 is used by the control system 170 and the odometry system 190 in planning future movements and estimated locations within the environment 10. However, in some situations, there may not be a map 182 associated with the environment 10 and/or the sensor data 134 may be of poor quality, which makes accurate localization more difficult.

[0048] As discussed in greater detail below, the localization module 200 of the present disclosure is configured to perform localization of the robot 100 when the sensor data 134 (e.g., image data 134) received from the sensors 132 is unreliable and without needing a pre-existing map 182. Particularly, the localization module 200 uses the image data 134 to obtain localization results and compares these localization results to the odometry information 192 to determine whether or not the image data 134 is reliable. More specifically, when the robot 100 travels a relatively short period (i.e., time or distance), the localization module 200 is configured to assume that the noise (e.g., drift) in the odometry information 192 within that relatively short period is negligible and that the odometry information 192 is reliable. Accordingly, when the localization module 200 determines that the localization results obtained from the image data 134 vary significantly from the odometry information 192 within this relatively short period, the localization module 200 provides greater weight to the odometry information 192 based on the assumption that the odometry information 192 is accurate and reliable. In some instances, the localization module 200 determines that the image data 134 is unreliable and cannot be used by the robot 100. Additionally, the localization module 200 may generate a notification 204 instructing an operator/user of the robot 100 that the area of the environment 10 associated with the unreliable image data 134, and therefore unreliable current location data 202 obtained therefrom, lacks sufficient reference features 14 for the sensor system 132 to measure.

[0049] Described with reference to FIGS. 4–7B, FIG. 3 shows an example process 310 executed by the localization module 200 for determining whether or not sensor data 134 is reliable as the robot 100 moves through the environment 10. The localization module 200 may execute the process 310 on-line while the robot 100 is traversing the environment 10 to build a map 182. As discussed previously, the localization module 200 iteratively updates the current location data 202 based on the sensor data 134 and the odometry information 192 provided by the sensor system 130 and the odometry system 190, respectively. While the localization module 200 typically gives the sensor data 134 greater weight than the odometry information 192 when determining the current location data 202, in some instances, the localization module 200 determines that the sensor data 134 is unreliable and reduces the weight given to the sensor data 134 when determining the current location of the robot 100.

[0050] At step 312, the localization process 310 receives sensor data 134 associated with an area or region of the environment 10. Particularly, the sensor data 134 may include vision data 134 measured by one or more of the image sensors 132. The localization process 310 may also receive one or more of the maps 182 from the perception system 180. Based on the image data 134, the process 310 determines sensor location data 313, 313a–n representing locations of the robot 100 within the environment 10. Particularly, the sensor location data 313 represents the locations of the robot 100 relative to one or more reference features 14 within the environment 10, which may be included in the map 182, if available. However, in some situations, the sensor data 134 received by the localization module 200 may be unreliable, and may result in inconsistent sensor location data 313. For example, in areas of the environment 10 where reference features 14 are sparsely populated (e.g., empty spaces, buildings), repetitive (e.g., lights, pillars), or ambiguous (e.g., partially obstructed, reflective, blurry), the localization module 200 may struggle to accurately identify suitable reference features 14 that can be evaluated to determine the location of the robot 100.

[0051] At step 314, the localization process 310 receives odometry information 192 associated with movements of the robot 100 within the environment. Optionally, the localization module 200 receives prior location data 202_{prior} of the robot 100. The prior location data 202_{prior} represents a last-known location of the robot 100 within the environment 10 based on a preceding iteration 310_{i-1} of the localization process 310. Using the odometry information 192, and optionally, the prior location data 202_{prior} (i.e., the last known position), the localization module 200 generates odometry location information 315, 315a–n representing estimated movements and/or locations of the robot 100 within the environment.

[0052] To generate the odometry location information 315, the localization module 200 evaluates the odometry information 192 to determine a change in location with respect to the prior location data 202_{prior} of the robot 100. For example, the odometry information 192 may specify that the robot 100 has moved at a known speed and/or distance along a known direction during a preceding time period. While the odometry information 192 may include some noise/error associated with unplanned events (e.g., slips, obstructions) and measurement errors (e.g., joint and sensor tolerances), the noise/error in the odometry information 192 will be negligible for short time periods and/or distances.

[0053] At step 316, the localization process 310 obtains a sample 317 of the sensor location data 313 including a plurality of sensor location data members 313_{ref}, 313b–313e (FIG. 4) within a target sample basin 410 (FIG 4). The sample basin 410 may be associated with a period of time, an estimated distance traveled, or a target size for the sensor location data sample 317. FIG. 4 shows a representation of the sensor location data sample 317 including a series of five subsequent sensor location data members 313_{ref}, 313b–313e. For the sake of clarity, each sensor location data member 313_{ref}, 313b–313e is associated with a respective delta time dt within a sample basin 410 corresponding to a sample time period T_{410}. As discussed in greater detail below, the sensor location data sample 317 includes a reference sensor location data member 313_{ref} and a plurality of comparison sensor location data members 313b–313e. Subsequent operations of the localization process 310 are performed on the comparison sensor location data members 313b–313e relative to the reference sensor location data member 313_{ref}.

[0054] Referring back to FIG. 3, at step 318, the localization process 310 obtains a sample 319 of the odometry location information 315 including a plurality of odometry location information members 315_{ref}, 315b–315e within the target sample basin 410. Each of the odometry location information members 315ref, 315b–315e corresponds in time dt to a respective one of the sensor location data members 313_{ref}, 313b–313e. Accordingly, FIG. 4 shows the odometry location information sample 319 including a reference odometry location information member 315_{ref} and a plurality of comparison odometry location information members 315b–315e.

[0055] As described in greater detail below, the localization process 310 ultimately determines the reliability of the sensor data 134 based on the assumption that the comparison odometry location information members 315b–315e of the odometry location information sample 319 are accurate relative to the reference odometry location information member 315ref. Accordingly, the time period T_{410} of the sample basin 410 may be tuned by an operator to balance the benefits of collecting a large sample size with the benefits of obtaining a reliable odometry location information sample 319. For instance, a smaller sample basin 410 may be selected to provide a pool of members 313n, 315n where the locations represented by the odometry location information members 315ref, 315b–315e are more likely to correspond the locations represented by the sensor location data members 313_{ref}, 313b–313e. On the other hand, a larger sample basin 410 provides a larger pool of members 313n, 315n for determining variance (discussed below), but may result in a less accurate odometry location information sample 319 as noise within the odometry location information members 315a-315n accumulates over time.

[0056] At step 320, the localization process 310 determines a plurality of localizations 321, 321_{ref}, 321b–321e based on the sensor location data sample 317 and the odometry location information sample 319. Particularly, the localization process 310 performs a pairwise analysis on each comparison sensor location data member 313b–313e with the reference sensor location data member 313_{ref} to determine how much the comparison sensor location data members 313b–313e agree with the reference sensor location data member 313_{ref}. In some examples, a localization algorithm includes a sixdegree-of-freedom (e.g., 3 translation axes, and 3 rotation axes) pairwise rigid transform from each comparison sensor location data member 313b–313e to the reference sensor position data member 313_{ref} based on the assumption that the environment 10 is static and the robot 100 is moving.

[0057] An example of the practical results of step 320 is represented in schematic views 500a, 500b of FIGS. 5A and 5B. FIG. 5A shows representations of the sensor location data members 313_{ref}, 313b–313e as respective locations of the robot 10 relative to a common reference feature 14 within the environment 10. When a map 182 is not available and/or image data 134 is poor, the robot 100 may move based on the assumption that movements of the robot 100 are consistent along the ground truth trajectory GT. Accordingly, in FIG. 5A, movement of the robot 100 appears consistent along a ground truth trajectory GT (e.g., the X-axis), while the location of the reference feature 14 is represented as being variable relative to the respective sensor location data members 313_{ref}, 313b–313e. For clarity, the relative location of the reference feature 14, 14ref, 14b–14e corresponding to each sensor location data member 313_{ref}, 313b–313e is labeled with a corresponding suffix (e.g., reference feature 14b represents the location of the reference feature 14b relative to the sensor position data members 313b). However, in reality, the represented locations of the reference feature 14ref, 14b–14e shown in FIG. 5A all correspond to the same reference feature 14, which has a fixed location relative to the environment 10 (e.g., a building column or wall).

[0058] At step 320, the localization process 310 performs the pairwise analysis to match the image data 134 (e.g., the reference feature 14) for each comparison sensor location data member 313b–313e to the image data 134_I of the reference sensor location data member 313_{ref}. In other words, the localization process 310 matches observed locations 14b–14e of the reference feature 14 for the comparison sensor location data members 313b–313e to the observed location 14_{ref} of the reference feature 14 for the reference sensor location data member 313_{ref}. FIG. 5B shows resulting localizations 321_{ref}, 321b–321e based on performing the pairwise analysis on the comparison sensor location data members 313b–313c and the reference sensor location data member 313_{ref}. As shown, the reference localization 321_{ref} is fixed at the origin and the comparison localizations 321b–321e are shifted relative to the reference localization 321_{ref}. After performing the localization algorithm, the comparison localizations 321b–321e are inconsistently arranged along the ground truth trajectory GT, while the reference features 14_{ref}, 14b-14e is shown at a common location.

[0059] Described with respect to FIGS. 6A and 6B, at step 322, the localization process 310 determines offsets between the localizations 321b–321e and the sample 319 of the odometry location information members 315b–315e associated with the sample basin 410. As discussed above, the noise associated with drift in the odometry location information sample 319 is negligible within the sample basin 410, such that the localization process 310 assumes that the comparison odometry location information members 315b–315e are accurate relative to the reference odometry location information member 315_{ref}, which is set as the origin in FIG. 6A. In other words, while the odometry location information members 315b–315e may not be accurate with respect to the ground truth, the localization process 310 assumes that any errors in the odometry location information members 315b–315e are relatively small and consistent within the sample basin 410. The localization process 310 sets the reference localization 321_{ref} at the origin (i.e., aligned with the reference odometry location information member 315_{ref}) and then determines the respective offsets 323b–323e between the comparison localizations 321b–321e and corresponding odometry location information members 315b–315e.

[0060] FIG. 6B shows the respective offsets 323b–323e of the comparison localizations 321b–321e relative to the corresponding odometry location information members 315b–315e, which are assumed to be accurate relative to the reference odometry location information member 315_{ref} and are set at the origin. As shown, the comparison localizations 321b–321e are inconsistently offset by different magnitudes and in different directions from the corresponding odometry location information members 315b–315e. Generally, the inconsistent offsets 323b–323e indicate that the underlying image data 134_I associated with the sensor location data 313 and the localizations 321 may be unreliable, as discussed below.

[0061] At step 324, the localization process 310 calculates a sample variance s^2_{offset} of the offsets 323b–323e. As shown in FIG. 7A, values of the offsets 323b–323e are plotted as positive and negative values representing a difference in the position of a respective localization 321b–321e from the corresponding odometry location information members 315b–315e. FIG. 7A shows a relatively wide distribution with respect to the difference from the odometry location information members 315b-315e, which ultimately corresponds to a relatively large variance (step 326) and unreliable image data 134_I. For comparison, FIG. 7B shows an representation of example offsets $323_{n–n+3}$ having a relatively narrow distribution with respect to the difference in odometry. In the example of FIG. 7B, the offsets $323_{n–n+3}$ are tightly grouped along the negative direction. Consistent offsets 323 in one direction, or tight grouping of the offsets 323 is more likely to represent an underlying sensor error (e.g., out of calibration) than unreliable sensor data 134. Thus, determining the sample variance s^2_{offset} of the offsets 323b–323e is an effective way to determine whether inaccurate sensor data 134 is caused by the environment 10 (FIG. 7A) or the robot 100 (FIG. 7B).

[0062] At step 326, the localization process determines whether the sample variance s^2_{offset} of the offsets 323b–323e exceeds a threshold sample variance $s^2_{\text{threshold}}$. The threshold sample variance $s^2_{\text{threshold}}$ may be a tuned value selected by the operator depending on a desired sensitivity of the localization process 310. For instance, a lower threshold variance $s^2_{\text{threshold}}$ may be selected where the operator wants the localization process 310 to be more sensitive (e.g., less trusting) to inconsistent image data 134_I and a higher threshold variance $s^2_{\text{threshold}}$ may be selected where the operator wants the localization process to be less sensitive (e.g., more trusting) to inconsistent image data 134_I.

[0063] When the localization process 310 determines that the sample variance s^2_{offset} of the offsets 323b–323e exceeds the threshold variance $s^2_{\text{threshold}}$ (e.g., FIG. 7A) (i.e., step 326 is “Yes”), the localization process 310 proceeds to step 328 and generates instructions 329 to reduce the weight of the sensor data 134 (e.g., image data) relative to the odometry information 192 when determining the current location data 202 of the robot 100 (step 334). Conversely, when the localization process 310 determines that the sample variance s^2_{offset} of the offsets 323b–323e does not exceed the threshold variance $s^2_{\text{threshold}}$ (e.g., FIG. 7B) (i.e., step 326 is “No”), the localization process 310 proceeds to step 330 and generates instructions 331 to maintain or increase the weight of the sensor data 134 (e.g., image data 134_I) relative to the odometry information 192 when determining the current location data 202 of the robot (step 334).

[0064] The instructions 329, 331 may include weight values or adjustments corresponding the magnitude of the sample variance s^2_{offset} of the offsets 323b–323e. Particularly, the weight values or adjustments may be proportional to the difference between the sample variance s^2_{offset} of the offsets 323b–323e and the threshold variance $s^2_{threshold}$. For example, when the sample variance s^2_{offset} of the offsets 323b–323e exceeds the threshold variance $s^2_{threshold}$ by a relatively large magnitude, the localization process 310 assigns a greater weight to the odometry information 192 than when the sample variance s^2_{offset} of the offsets 323b–323e exceeds the threshold variance $s^2_{threshold}$ by a lesser magnitude. Similar principles are applied to increasing the weight of the sensor data 134 (step 330).

[0065] Optionally, when the localization process 310 determines that the sample variance s^2_{offset} of the offsets 323b–323e exceeds the threshold variance $s^2_{threshold}$ at step 326 (i.e., the response at step 226 is "Yes"), the localization process 310 generates a notification 204 or report 333 indicating that the sensor data 134 corresponding to the region of the environment 10 associated with the sample basin 410 is unreliable. The sensor data report 333 can be used by an operator to determine regions of the environment 10 where additional and/or better reference features 14 are needed for the image data 134_I to be improved.

[0066] At step 334, the localization process 310 determines the current location data 202 of the robot 100 by performing a localization algorithm based on the instructions 329, 331. The localization module 200 provides the current location data 202 to the control system 170, which uses the current location data 202 to determine future movements within the environment 10. The current location data 202 may also be used in a subsequent iteration 310_{i+1} of the localization process 310.

[0067] The localization process 310 can be executed by the localization module 200 while the robot 100 is online, which allows the localization module 200 to determine regions of the environment 10 where the sensor data 134 is unreliable before the environment is fully mapped by the perception system 180. Additionally, the localization process 310 is model-free, such that localization module 200 does not need to know the localization algorithm or sensor properties to determine whether the image data 134_I is reliable.

[0068] FIG. 8 is schematic view of an example computing device 800 that may be used to implement the systems and methods described in this document. The computing device 800 is intended to represent various forms of digital computers, such as laptops, desktops, workstations, personal digital assistants, servers, blade servers, mainframes, and other appropriate computers. The components shown here, their connections and relationships, and their functions, are meant to be exemplary only, and are not meant to limit implementations of the inventions described and/or claimed in this document.

[0069] The computing device 800 includes a processor 810 (e.g., data processing hardware 502), memory 820, a storage device 830, a high-speed interface/controller 840 connecting to the memory 820 and high-speed expansion ports 850, and a low speed interface/controller 860 connecting to a low speed bus 870 and a storage device 830. Each of the components 810, 820, 830, 840, 850, and 860, are interconnected using various busses, and may be mounted on a common motherboard or in other manners as appropriate. The processor 810 can process instructions for execution within the computing device 800, including instructions stored in the memory 820 or on the storage device 830 to display graphical information for a graphical user interface (GUI) on an external input/output device, such as display 880 coupled to high speed interface 840. In other implementations, multiple processors and/or multiple buses may be used, as appropriate, along with multiple memories and types of memory. Also, multiple computing devices 800 may be connected, with each device providing portions of the necessary operations (e.g., as a server bank, a group of blade servers, or a multi-processor system).

[0070] The memory 820 (e.g. memory hardware 504) stores information non-transitorily within the computing device 800. The memory 820 may be a computer-readable medium, a volatile memory unit(s), or non-volatile memory unit(s). The non-transitory memory 820 may be physical devices used to store programs (e.g., sequences of instructions) or data (e.g., program state information) on a temporary or permanent basis for use by the computing device 800. Examples of non-volatile memory include, but are not limited to, flash memory and read-only memory (ROM) / programmable read-only memory (PROM) / erasable programmable read-only memory (EPROM) / electronically erasable programmable read-only memory (EEPROM) (e.g., typically used for firmware, such as boot programs). Examples of volatile memory include, but are not limited to, random access memory (RAM), dynamic random access memory (DRAM), static random access memory (SRAM), phase change memory (PCM) as well as disks or tapes.

[0071] The storage device 830 (e.g., memory hardware 504) is capable of providing mass storage for the computing device 800. In some implementations, the storage device 830 is a computer-readable medium. In various different implementations, the storage device 830 may be a floppy disk device, a hard disk device, an optical disk device, or a tape device, a flash memory or other similar solid state memory device, or an array of devices, including devices in a storage area network or other configurations. In additional implementations, a computer program product is tangibly embodied in an information carrier. The computer program product contains instructions that, when executed, perform one or more methods, such as those described above. The information carrier is a computer- or machine-readable medium, such as the memory 820, the storage device 830, or memory on processor 810.

[0072] The high speed controller 840 manages bandwidth-intensive operations for the computing device 800, while the low speed controller 860 manages lower bandwidth-intensive operations. Such allocation of duties is exemplary only. In some implementations, the high-speed controller 840 is coupled to the memory 820, the display 880 (e.g., through a graphics processor or accelerator), and to the high-speed expansion ports 850, which may accept various expansion cards (not shown). In some implementations, the low-speed controller 860 is coupled to the storage device 830 and a low-speed expansion port 890. The low-speed expansion port 890, which may include various communication ports (e.g., USB, Bluetooth, Ethernet, wireless Ethernet), may be coupled to one or more input/output devices, such as a keyboard, a pointing device, a scanner, or a networking device such as a switch or router, e.g., through a network adapter.

[0073] The computing device 800 may be implemented in a number of different forms, as shown in the figure. For example, it may be implemented as a standard server 800a or multiple times in a group of such servers 800a, as a laptop computer 800b, or as part of a rack server system 800c.

[0074] Various implementations of the systems and techniques described herein can be realized in digital electronic and/or optical circuitry, integrated circuitry, specially designed ASICs (application specific integrated circuits), computer hardware, firmware, software, and/or combinations thereof. These various implementations can include implementation in one or more computer programs that are executable and/or interpretable on a programmable system including at least one programmable processor, which may be special or general purpose, coupled to receive data and instructions from, and to transmit data and instructions to, a storage system, at least one input device, and at least one output device.

[0075] These computer programs (also known as programs, software, software applications or code) include machine instructions for a programmable processor, and can be implemented in a high-level procedural and/or object-oriented programming language, and/or in assembly/machine language. As used herein, the terms "machine-readable medium" and "computer-readable medium" refer to any computer program product, non-transitory computer readable medium, apparatus and/or device (e.g., magnetic discs, optical disks, memory, Programmable Logic Devices (PLDs)) used to provide machine instructions and/or data to a programmable processor, including a machine-readable medium that receives machine instructions as a machine-readable signal. The term "machine-readable signal" refers to any signal used to provide machine instructions and/or data to a programmable processor.

[0076] The processes and logic flows described in this specification can be performed by one or more programmable processors executing one or more computer programs to perform functions by operating on input data and generating output. The processes and logic flows can also be performed by special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit). Processors suitable for the execution of a computer program include, by way of example, both general and special purpose microprocessors, and any one or more processors of any kind of digital computer. Generally, a processor will receive instructions and data from a read only memory or a random access memory or both. The essential elements of a computer are a processor for performing instructions and one or more memory devices for storing instructions and data. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto optical disks, or optical disks. However, a computer need not have such devices. Computer readable media suitable for storing computer program instructions and data include all forms of non-volatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto optical disks; and CD ROM and DVD-ROM disks. The processor and the memory can be supplemented by, or incorporated in, special purpose logic circuitry.

[0077] To provide for interaction with a user, one or more aspects of the disclosure can be implemented on a computer having a display device, e.g., a CRT (cathode ray tube), LCD (liquid crystal display) monitor, or touch screen for displaying information to the user and optionally a keyboard and a pointing device, e.g., a mouse or a trackball, by which the user can provide input to the computer. Other kinds of devices can be used to provide interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback, e.g., visual feedback, auditory feedback, or tactile feedback; and input from the user can be received in any form, including acoustic, speech, or tactile input. In addition, a computer can interact with a user by sending documents to and receiving documents from a device that is used by the user; for example, by sending web pages to a web browser on a user's client device in response to requests received from the web browser.

[0078] A number of implementations have been described. Nevertheless, it will be understood that various modifications may be made without departing from the spirit and scope of the disclosure. Accordingly, other implementations are within the scope of the following claims.

WHAT IS CLAIMED IS:

1. A method (310) comprising:

while a robot (100) traverses an environment (10):

receiving, at data processing hardware (142), odometry information (192) plotting a location (202) of the robot (100) over time; and

receiving, at the data processing hardware (142), from at least one sensor (132) of the robot (100) in communication with the data processing hardware (142), sensor data (134) of the environment (10) about the robot (100);

obtaining, by the data processing hardware (142), an odometry information sample (319) including a series of odometry information members (315) collected by the robot (100) while traversing the environment (10), each odometry information member (315) of the series of odometry information members (315) comprising a respective odometry measurement at a respective time;

obtaining, by the data processing hardware (142), a sensor data sample (317) including a series of sensor data members (313) collected by the robot (100) while traversing the environment (10), each sensor data member (313) of the series of sensor data members (313) comprising a respective sensor measurement at the respective time;

for each sensor data member (313) of the series of sensor data members (313):

determining, by the data processing hardware (142), a localization of the robot (100) at the respective time based on the respective sensor data (134); and

determining, by the data processing hardware (142), an offset (323) of the localization of the robot (100) relative to the odometry measurement at the respective time;

determining, by the data processing hardware (142), whether a variance (s^2_{offset}) of the offsets (323) determined for the localizations (321) exceeds a threshold variance ($s^2_{\text{threshold}}$); and

when the variance (s^2_{offset}) among the offsets (323) exceeds the threshold variance ($s^2_{\text{threshold}}$), generating, by the data processing hardware (142), a signal (204) indicating that the sensor data (134) is unreliable.

2. The method (310) of claim 1, further comprising adjusting, by the data processing hardware (142), relative weights of the odometry information (192) and the sensor data (134) based on whether the variance (s^2_{offset}) of the offsets (323) determined for the localizations (321) exceed the threshold variance ($s^2_{threshold}$).

3. The method (310) of claim 2, further comprising determining, by the data processing hardware (142), a current location of the robot (100) using the adjusted relative weights of the odometry information (192) and the sensor data (134).

4. The method (310) of any of claims 1–3, further comprising, when the variance of the offsets (323) exceeds the threshold variance ($s^2_{threshold}$), reducing, by the data processing hardware (142), a weight assigned to the sensor data (134).

5. The method (310) of any of claims 1–4, further comprising, when the variance of the offsets (323) does not exceed the threshold variance, increasing, by the data processing hardware (142), a weight assigned to the sensor data (134).

6. The method (310) of any of claims 1–5, wherein the series of sensor data members (313) includes a reference sensor data member (313_{ref}) and a plurality of comparison sensor data members (313a–f).

7. The method (310) of claim 6, wherein determining the localization includes performing a pairwise analysis on one of the comparison sensor data members (313a–f) associated with the respective time (dt) with the reference sensor data member (313_{ref}).

8. The method (310) of any of claims 1–7, wherein the variance (s^2_{offset}) of the offsets (323) is a sample variance.

9. A system comprising:

data processing hardware (142); and

memory hardware (140) in communication with the data processing hardware (142), the memory hardware (140) storing instructions that when executed on the data processing hardware (142) cause the data processing hardware (142) to perform operations comprising:

while a robot (100) traverses an environment (10):

receiving odometry information (192) plotting a location (202) of the robot (100) over time; and

receiving from at least one sensor (132) of the robot (100) in communication with the data processing hardware (142), sensor data (134) of the environment (10) about the robot (100);

obtaining an odometry information sample (319) including a series of odometry information members (315) collected by the robot (100) while traversing the environment (10), each odometry information member (315) of the series of odometry information members (315) comprising a respective odometry measurement at a respective time;

obtaining a sensor data sample (317) including a series of sensor data members (313) collected by the robot (100) while traversing the environment (10), each sensor data member (313) of the series of sensor data members (313) comprising a respective sensor measurement at the respective time;

for each sensor data member (313) of the series of sensor data members (313):

determining a localization of the robot (100) at the respective time based on the respective sensor data (134); and

determining an offset (323) of the localization of the robot (100) relative to the odometry measurement at the respective time;

determining whether a variance (s^2_{offset}) of the offsets (323) determined for the localizations (321) exceeds a threshold variance ($s^2_{\text{threshold}}$); and

when the variance (s^2_{offset}) among the offsets (323) exceeds the threshold variance ($s^2_{\text{threshold}}$), generating, by the data processing hardware (142), a signal (204) indicating that the sensor data (134) is unreliable.

10. The system of claim 9, wherein the operations further comprise adjusting, by the data processing hardware (142), relative weights of the odometry information (192) and the sensor data (134) based on whether the variance (s^2_{offset}) of the offsets (323) determined for the localizations (321) exceed the threshold variance ($s^2_{\text{threshold}}$).

11. The system of claim 10, wherein the operations further comprise determining a current location of the robot (100) using the adjusted relative weights of the odometry information (192) and the sensor data (134).

12. The system of any of claims 9–11, wherein the operations further comprise, when the variance of the offsets (323) exceeds the threshold variance, reducing a weight assigned to the sensor data (134).

13. The system of any of claims 9–12, wherein the operations further comprise, when the variance of the offsets (323) does not exceed the threshold variance, increasing weight assigned to the sensor data (134).

14. The system of any of claims 9–13, wherein the series of sensor data members (313) includes a reference sensor data member (313) and a plurality of comparison sensor data members (313).

15. The system of claim 14, wherein determining the localization includes performing a pairwise analysis on one of the comparison sensor data members (313) associated with the respective time with the reference sensor data member (313).

16. The system of any of claims 9–15, wherein the variance (s^2_{offset}) of the offsets (323) is a sample variance.

17. A robot (100) comprising:

a body (110);

data processing hardware (142); and

memory hardware (140) in communication with the data processing hardware (142), the memory hardware (140) storing instructions that when executed on the data processing hardware (142) cause the data processing hardware (142) to perform operations comprising:

while a robot (100) traverses an environment (10):

receiving odometry information (192) plotting a location (202) of the robot (100) over time; and

receiving from at least one sensor (132) of the robot (100) in communication with the data processing hardware (142), sensor data (134) of the environment (10) about the robot (100);

obtaining an odometry information sample (319) including a series of odometry information members (315) collected by the robot (100) while traversing the environment (10), each odometry information member (315) of the series of odometry information members (315) comprising a respective odometry measurement at a respective time;

obtaining a sensor data sample (317) including a series of sensor data members (313) collected by the robot (100) while traversing the environment (10), each sensor data member (313) of the series of sensor data members (313) comprising a respective sensor measurement at the respective time;

for each sensor data member (313) of the series of sensor data members (313):

determining a localization of the robot (100) at the respective time (dt) based on the respective sensor data (134); and

determining an offset (323) of the localization of the robot (100) relative to the odometry measurement at the respective time (dt);

determining whether a variance (s^2_{offset}) of the offsets (323) determined for the localizations (321) exceeds a threshold variance ($s^2_{\text{threshold}}$); and when the variance (s^2_{offset}) among the offsets (323) exceeds the threshold variance ($s^2_{threshold}$), generating, by the data processing hardware (142), a signal (204) indicating that the sensor data (134) is unreliable.

18. The robot (100) of claim 17, wherein the operations further comprise adjusting, by the data processing hardware (142), relative weights of the odometry information (192) and the sensor data (134) based on whether the variance (s^2_{offset}) of the offsets (323) determined for the localizations (321) exceed the threshold variance ($s^2_{threshold}$).

19. The robot (100) of claim 18, wherein the operations further comprise determining a current location of the robot (100) using the adjusted relative weights of the odometry information (192) and the sensor data (134).

20. The robot (100) of any of claims 17–19, wherein the operations further comprise, when the variance (s^2_{offset}) of the offsets (323) exceeds the threshold variance ($s^2_{threshold}$), reducing a weight assigned to the sensor data (134).

21. The robot (100) of any of claims 17–20, wherein the operations further comprise, when the variance (s^2_{offset}) of the offsets (323) does not exceed the threshold variance ($s^2_{threshold}$), increasing a weight assigned to the sensor data (134).

22. The robot (100) of any of claims 17–21, wherein the series of sensor data members (313) includes a reference sensor data member (313) and a plurality of comparison sensor data members (313).

23. The robot (100) of claim 22, wherein determining the localization includes performing a pairwise analysis on one of the comparison sensor data members (313) associated with the respective time with the reference sensor data member (313).

24. The robot (100) of any of claims 17–23, wherein the variance (s^2_{offset}) of the offsets (323) is a sample variance.

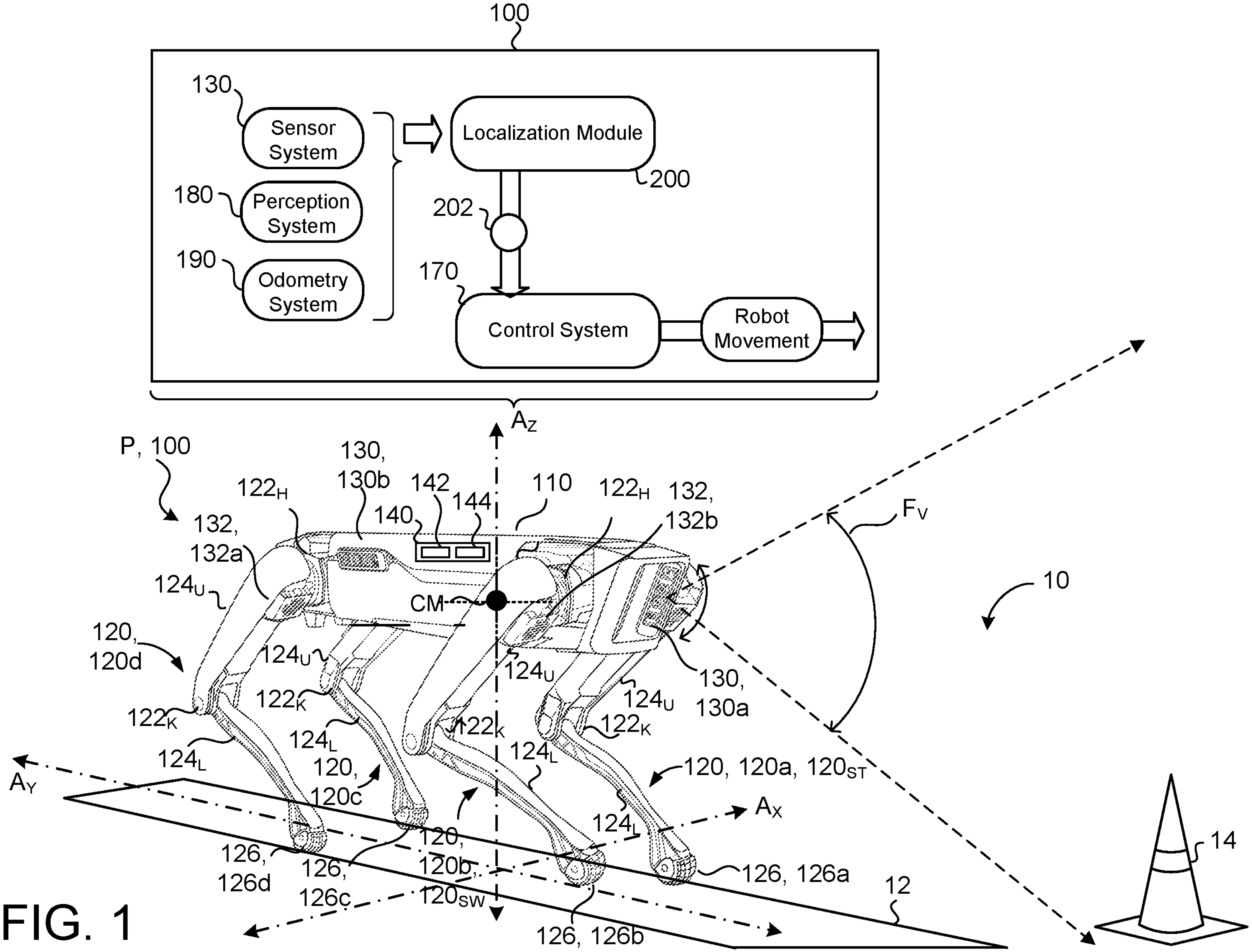
100
130
Sensor System
180
Perception System
190
Odometry System
Localization Module
200
202
170
Control System
Robot Movement
A_Z
P, 100
122_H
130, 130b
142
144
140
110
122_H
132, 132b
F_V
10
132, 132a
124_U
CM
120, 120d
124_U
124_U
130, 130a
124_U
122_K
122_K
122_K
122_K
124_L
124_L
124_L
120, 120c
120, 120a, 120_{ST}
A_Y
A_X
124_L
14
126, 126d
126, 126c
120, 120b, 120_{SW}
126, 126a
12
126, 126b

FIG. 1

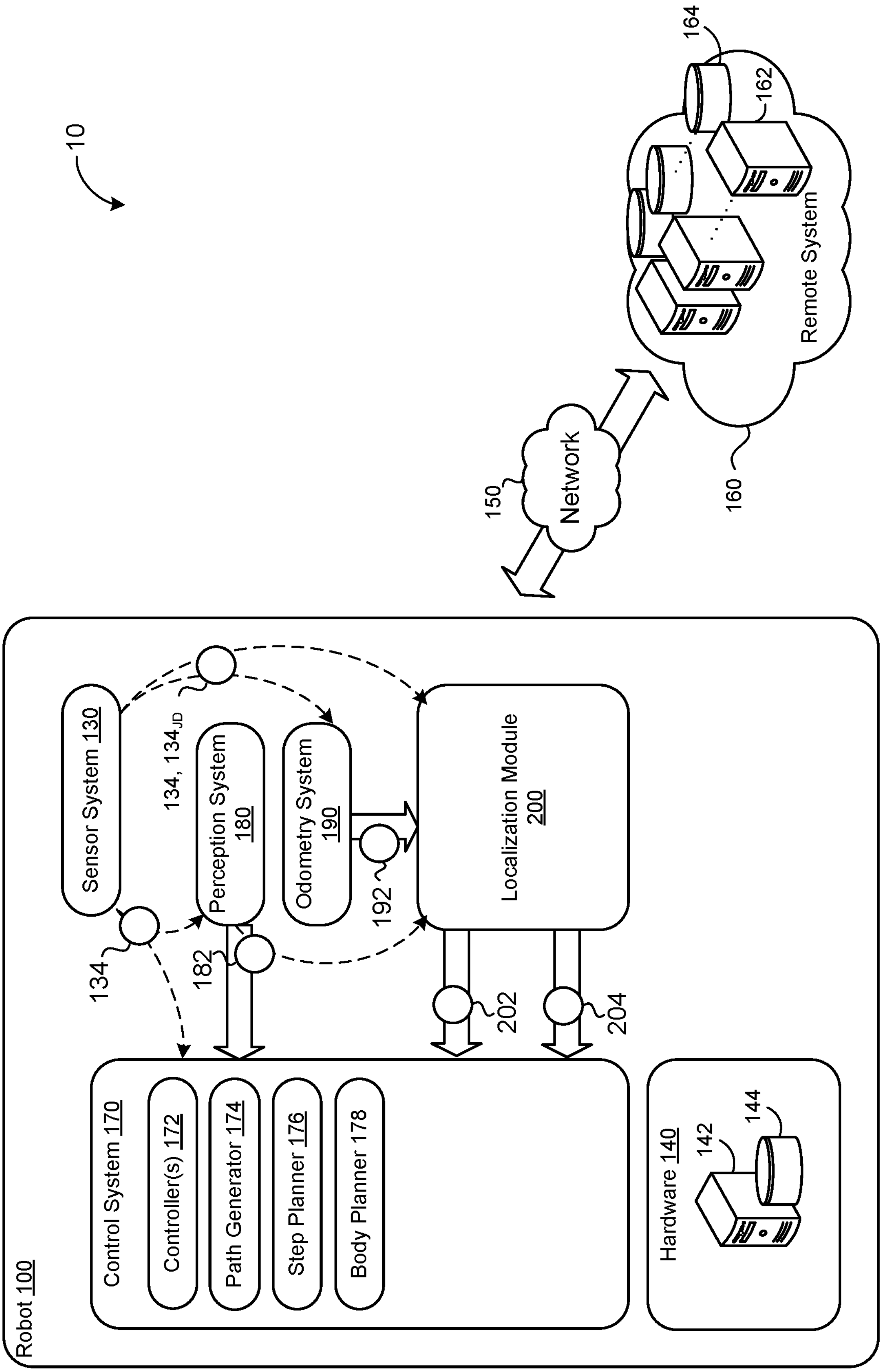
10
164
162
Remote System
Network
150
160
Robot 100
Sensor System 130
134, 134JD
Perception System 180
Odometry System 190
Localization Module 200
192
134
182
202
204
Control System 170
Controller(s) 172
Path Generator 174
Step Planner 176
Body Planner 178
Hardware 140
142
144

FIG. 2

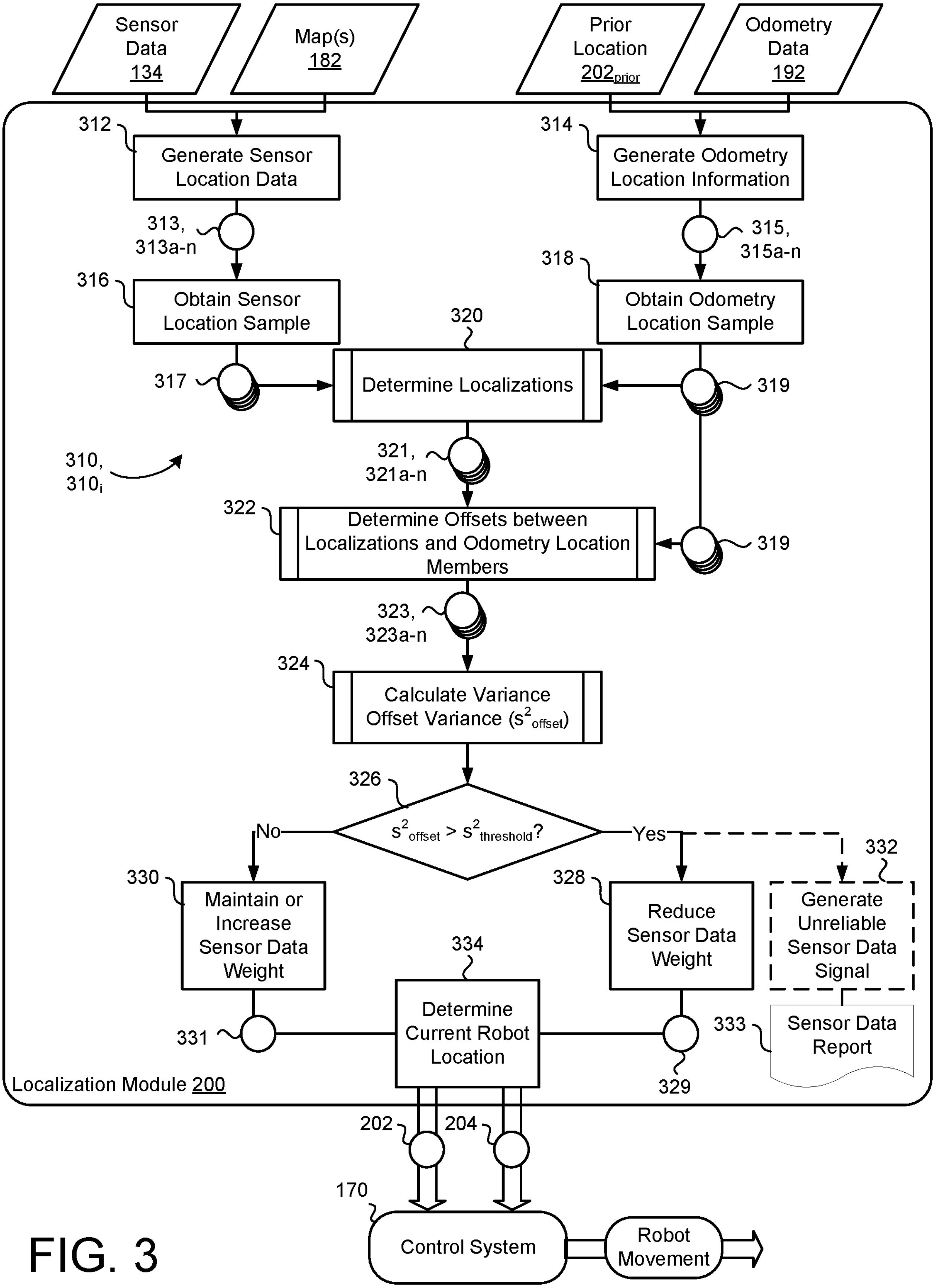
Sensor Data 134
Map(s) 182
Prior Location 202_{prior}
Odometry Data 192
312
Generate Sensor Location Data
314
Generate Odometry Location Information
313, 313a-n
315, 315a-n
316
Obtain Sensor Location Sample
318
Obtain Odometry Location Sample
320
Determine Localizations
317
319
310, 310_i
321, 321a-n
322
Determine Offsets between Localizations and Odometry Location Members
319
323, 323a-n
324
Calculate Variance Offset Variance (s^2_{offset})
326
$s^2_{offset} > s^2_{threshold}$?
No
Yes
330
Maintain or Increase Sensor Data Weight
328
Reduce Sensor Data Weight
332
Generate Unreliable Sensor Data Signal
334
Determine Current Robot Location
331
329
333
Sensor Data Report
Localization Module 200
202
204
170
Control System
Robot Movement

FIG. 3

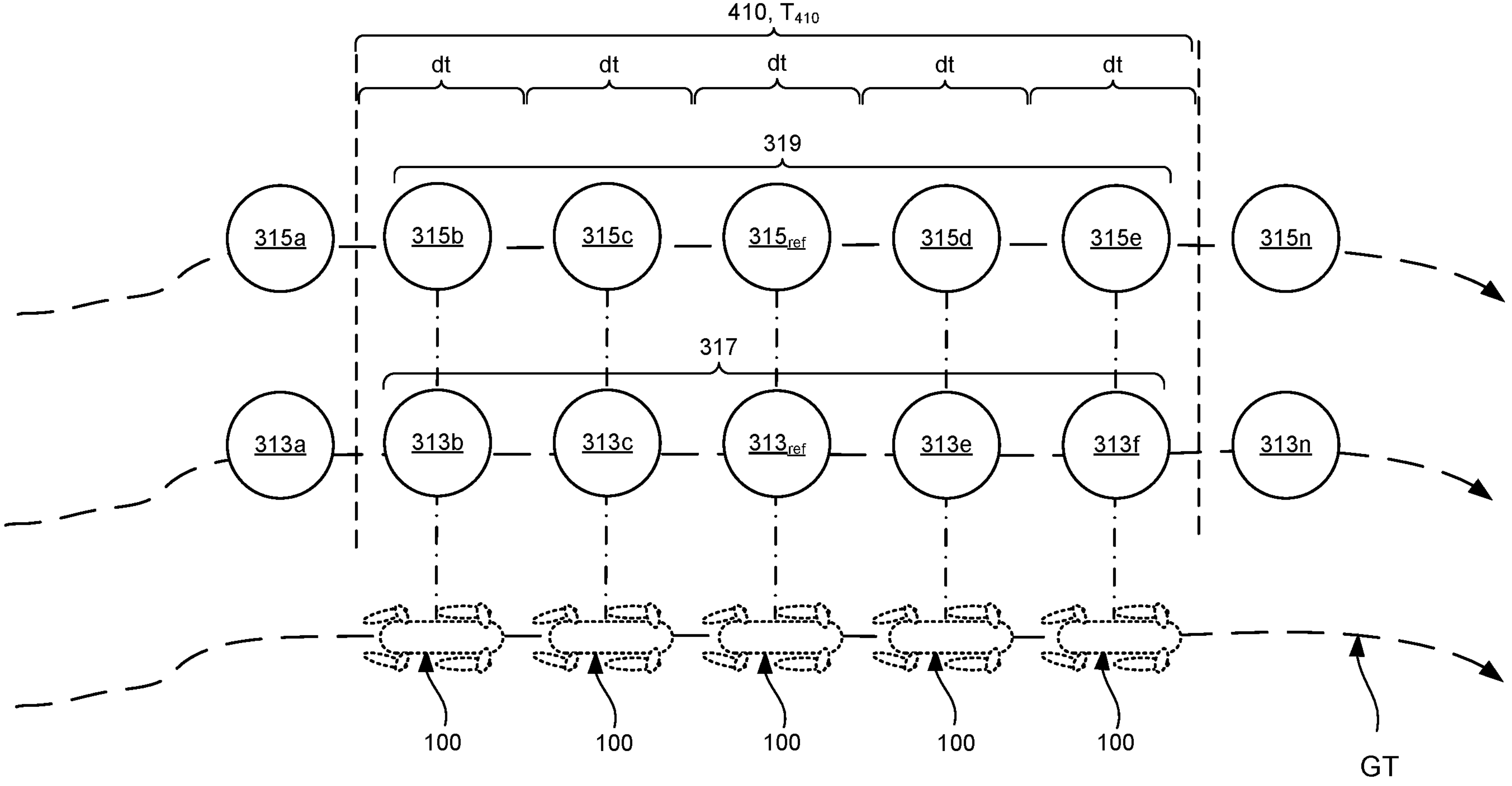
410, T_{410}
dt
dt
dt
dt
dt
319
315a
315b
315c
315_{ref}
315d
315e
315n
317
313a
313b
313c
313_{ref}
313e
313f
313n
100
100
100
100
100
GT

FIG. 4

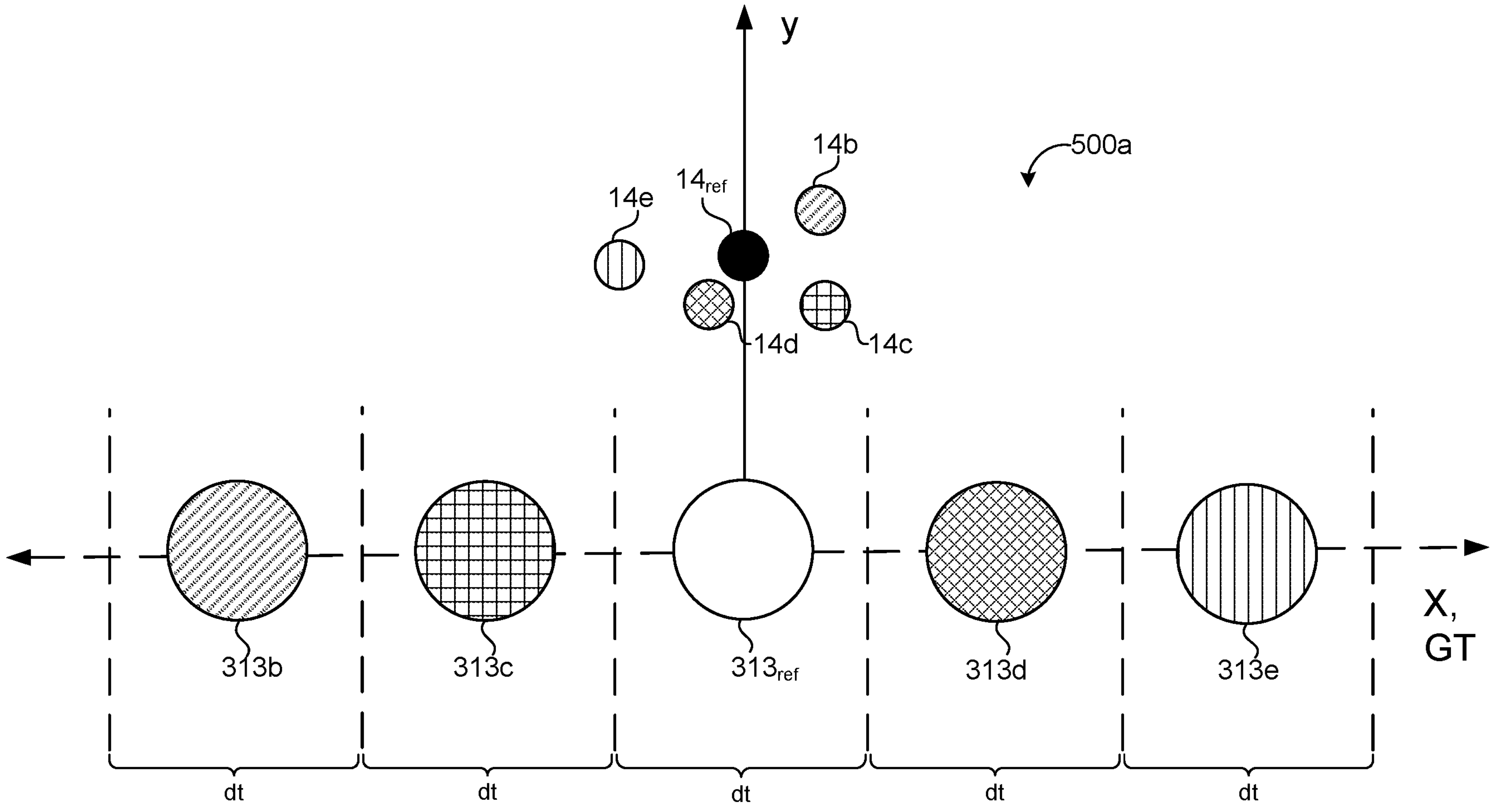
y
500a
14b
14e
14_{ref}
14d
14c
313b
313c
313_{ref}
313d
313e
X,
GT
dt
dt
dt
dt
dt

FIG. 5A

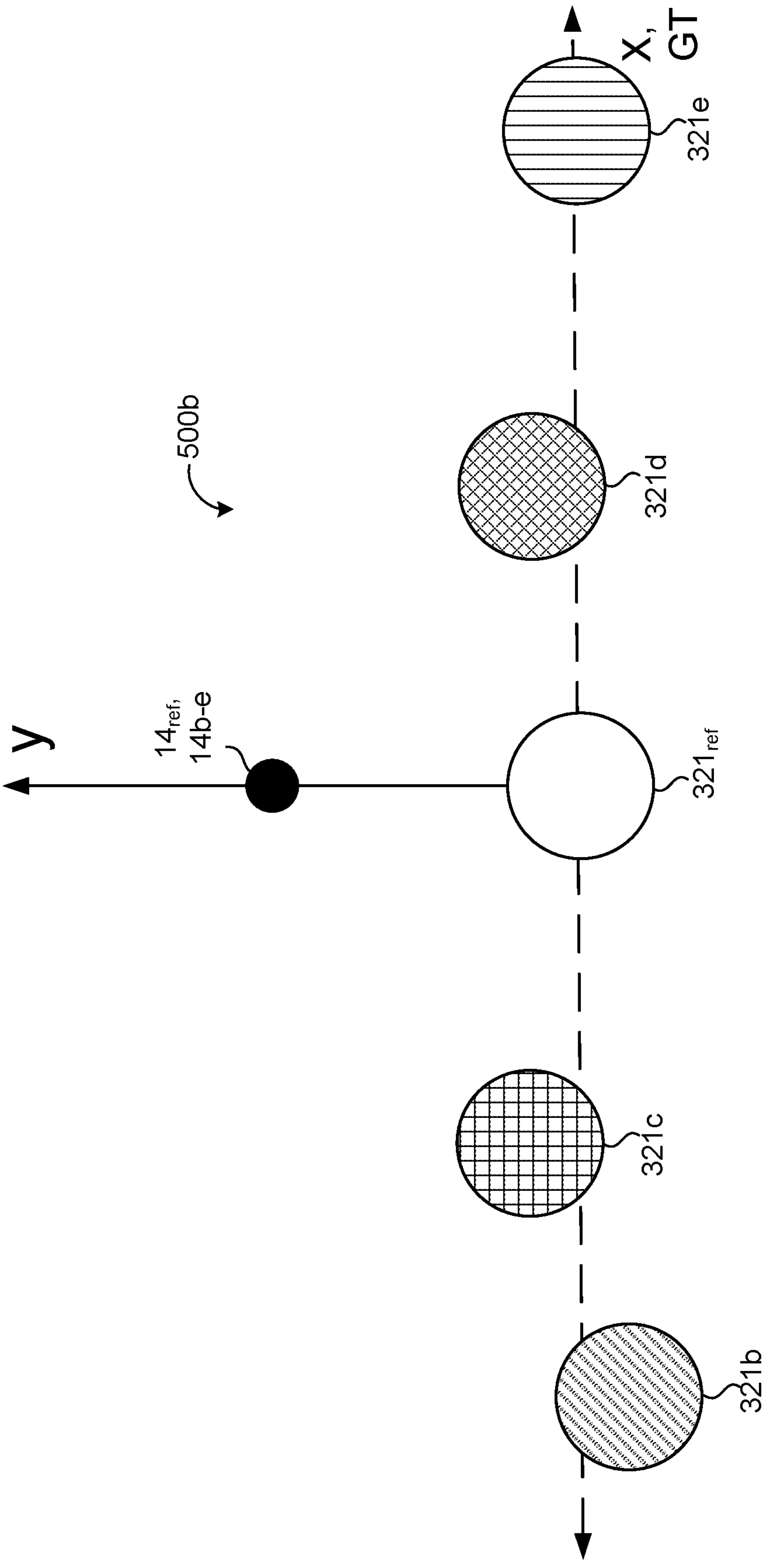

FIG. 5B

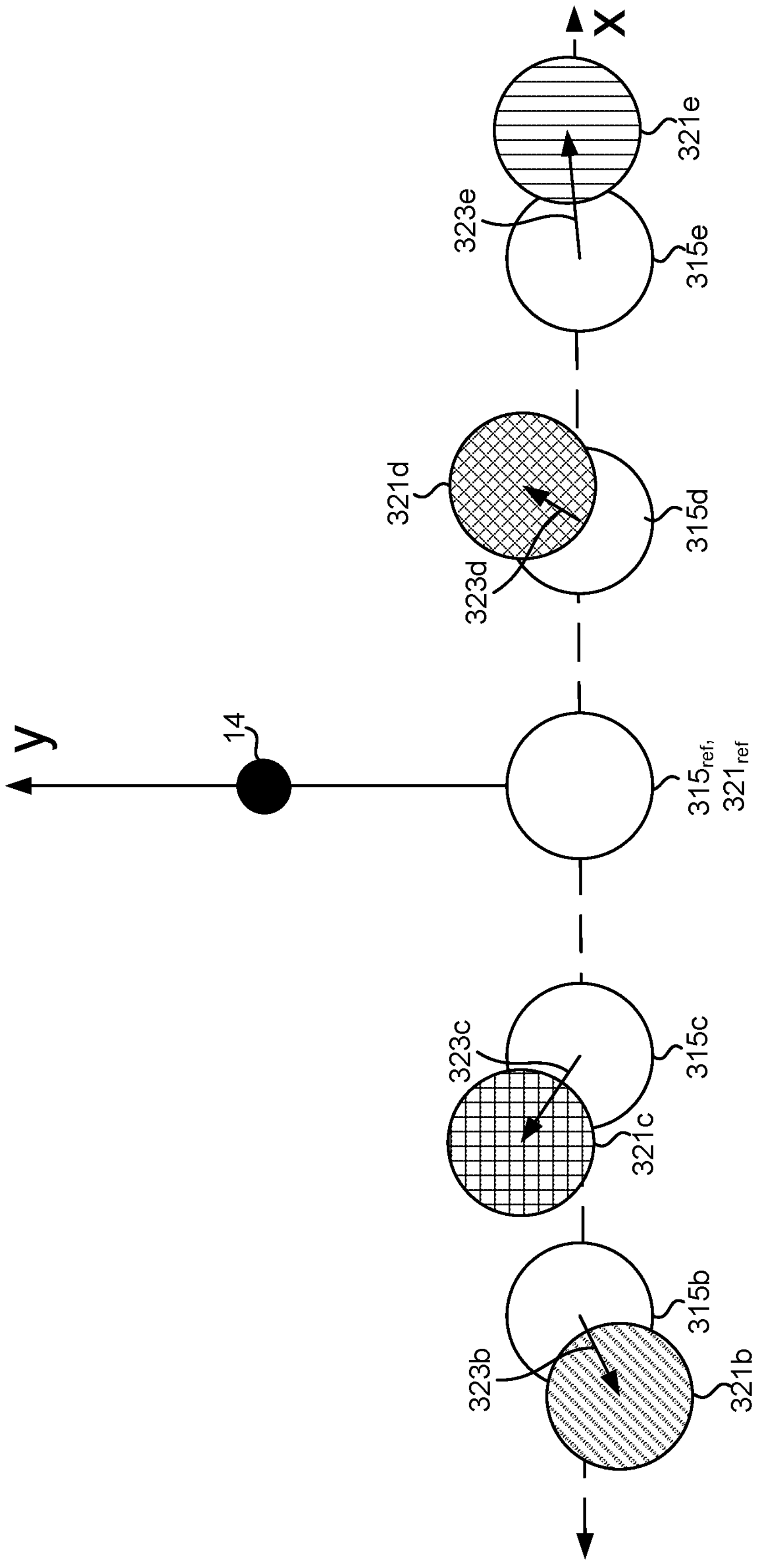
x
321e
323e
315e
321d
323d
315d
y
14
315_{ref},
321_{ref}
323c
315c
321c
315b
323b
321b

FIG. 6A

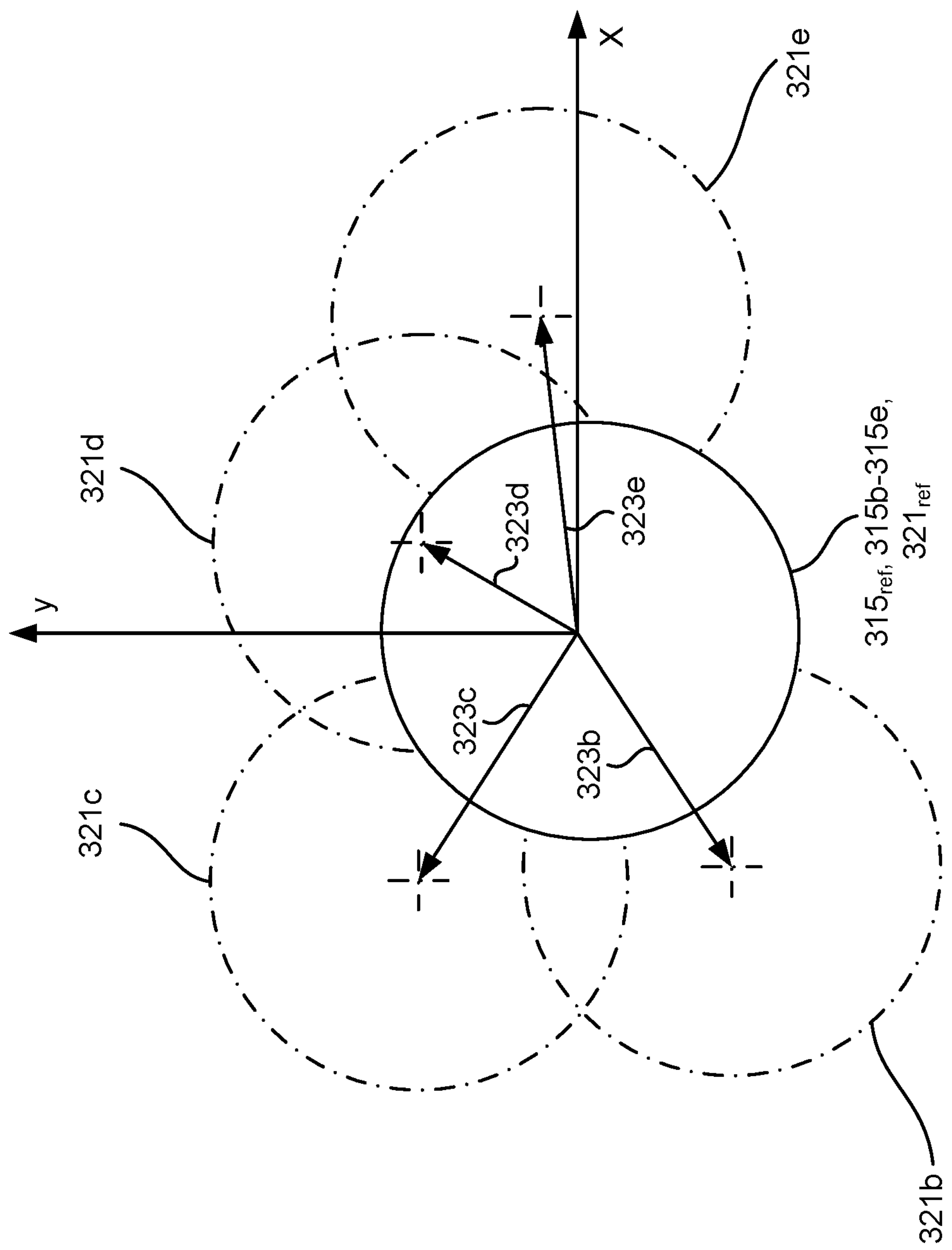
X
321e
321d
y
323d
323e
315$_{ref}$, 315b-315e,
321$_{ref}$
323c
323b
321c
321b

FIG. 6B

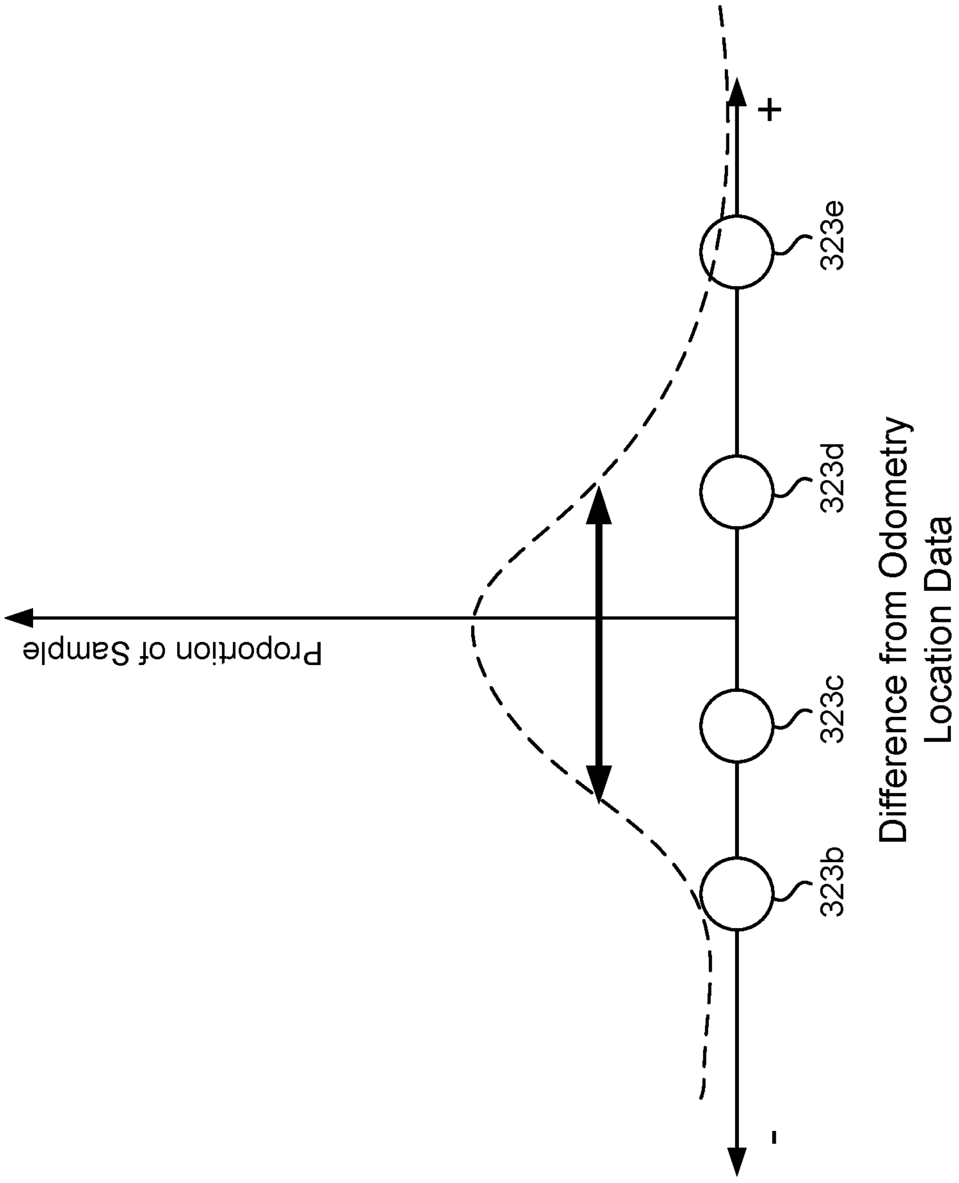
Proportion of Sample
+
-
323b
323c
323d
323e
Difference from Odometry Location Data

FIG. 7A

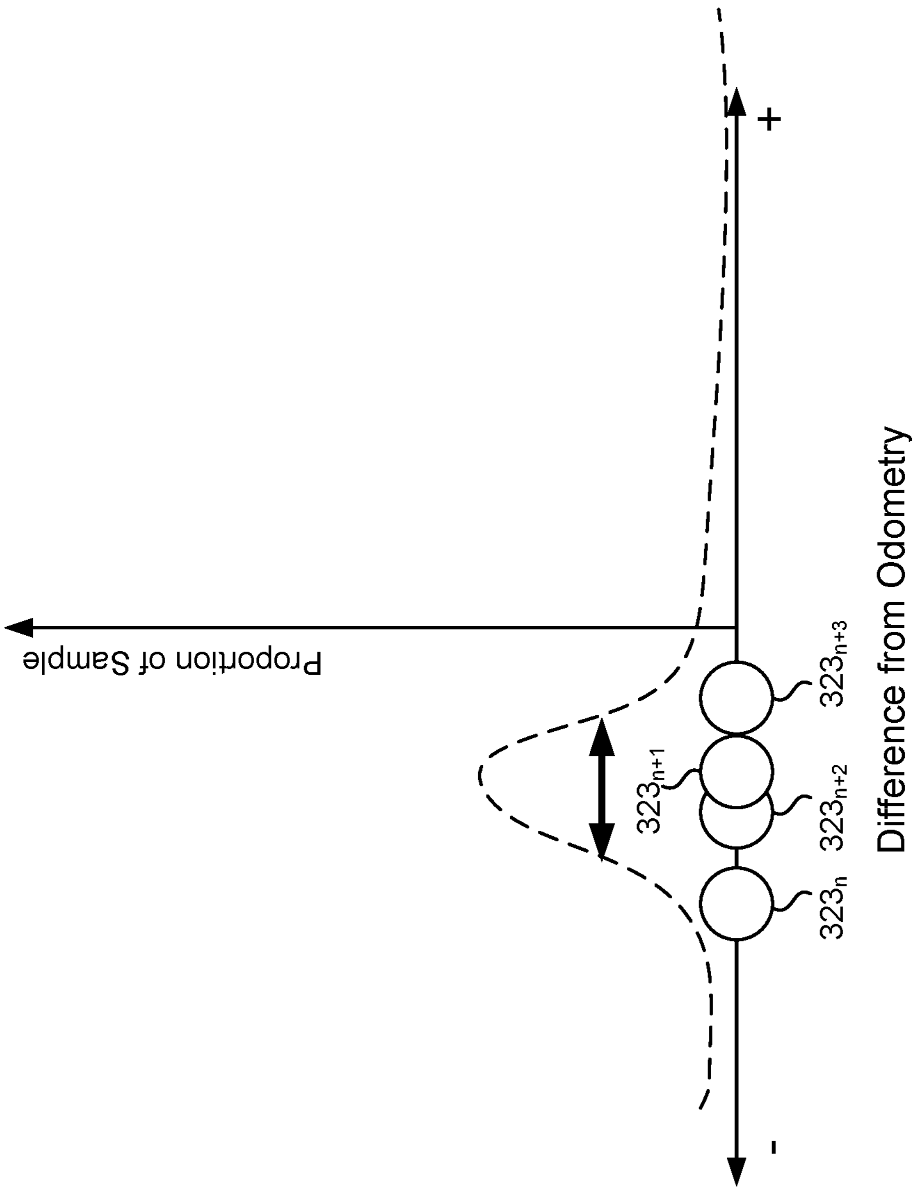
Proportion of Sample
+
-
Difference from Odometry
323n
323n+1
323n+2
323n+3

FIG. 7B

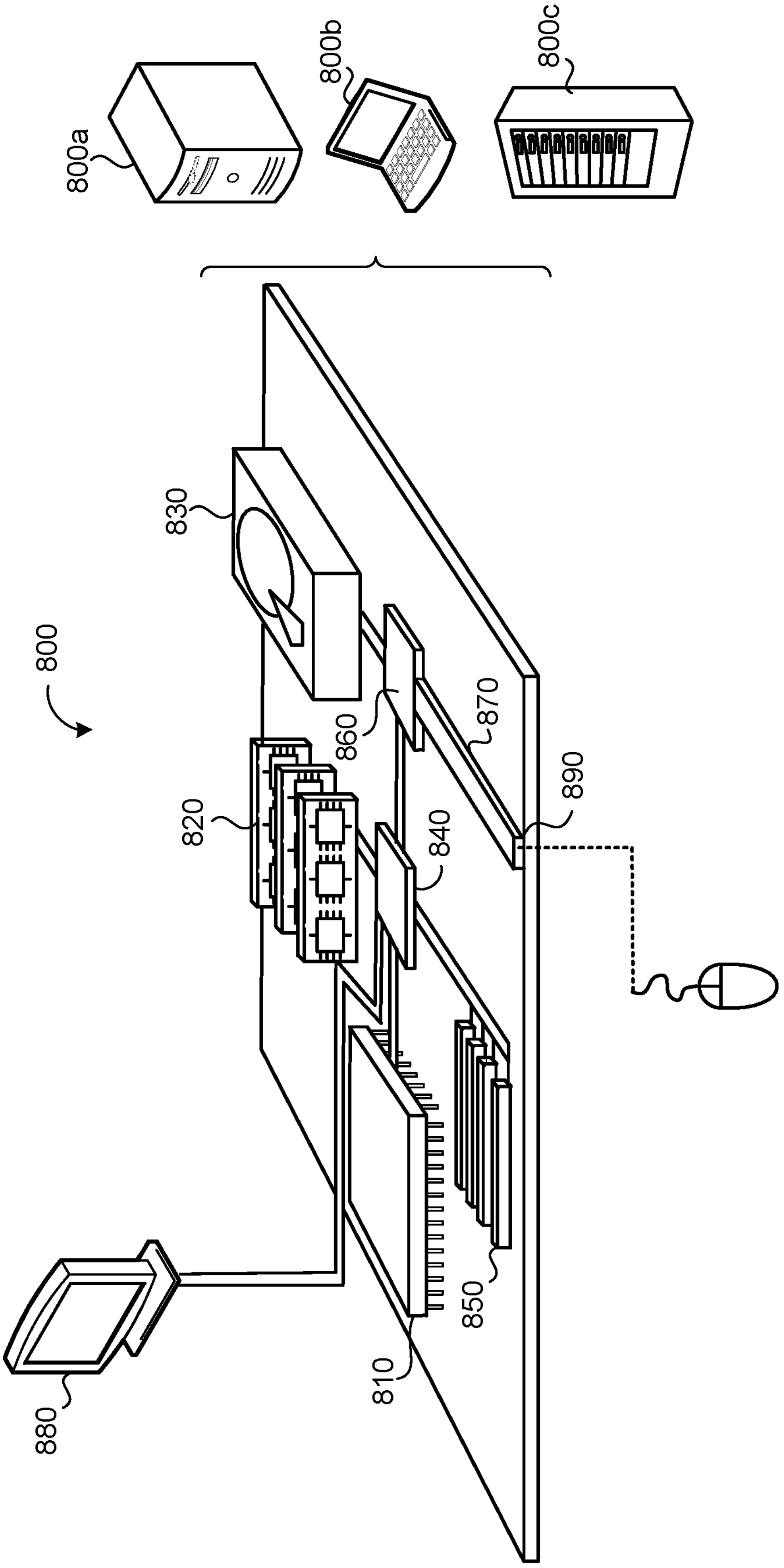

FIG. 8

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2021/025294

A. CLASSIFICATION OF SUBJECT MATTER
INV. G05D1/02 G01C22/00
ADD.

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G05D G01C

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2019/183727 A1 (AVIDBOTS CORP [CA]) 3 October 2019 (2019-10-03) paragraphs [1023], [1024], [1045], [1041], [1021]; figures 1-7 -----	1-24
A	DE 10 2018 101388 A1 (VALEO SCHALTER & SENSOREN GMBH [DE]) 25 July 2019 (2019-07-25) paragraphs [0047] - [0053]; figure 2 ----- -/--	1-24

☒ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search 4 June 2021	Date of mailing of the international search report 14/06/2021
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer Camerer, Stephan

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2021/025294

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WINKLER ALEXANDER ET AL: "Path planning with force-based foothold adaptation and virtual model control for torque controlled quadruped robots", 2014 IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION (ICRA), IEEE, 31 May 2014 (2014-05-31), pages 6476-6482, XP032650971, DOI: 10.1109/ICRA.2014.6907815 [retrieved on 2014-09-22] the whole document -----	1-24

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2021/025294

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2019183727 A1	03-10-2019	CA 3095222 A1	25-09-2020
		WO 2019183727 A1	03-10-2019
DE 102018101388 A1	25-07-2019	DE 102018101388 A1	25-07-2019
		EP 3743318 A1	02-12-2020
		KR 20200109365 A	22-09-2020
		US 2021031795 A1	04-02-2021
		WO 2019145152 A1	01-08-2019