The International Journal of Robotics Research
31(11) 1251–1262

DOI: 10.1177/0278364912455720
ijr.sagepub.com

Autonomous navigation of a humanoid robot over unknown rough terrain using a laser range sensor

Koichi Nishiwaki[1,2], Joel Chestnutt[3] and Satoshi Kagami[1,2]

Abstract

The present paper describes the integration of laser-based perception, footstep planning, and walking control of a humanoid robot for navigation over previously unknown rough terrain. A perception system that obtains the shape of the surrounding environment to an accuracy of a few centimeters is realized based on input obtained using a scanning laser range sensor. A footstep planner decides the sequence of stepping positions using the obtained terrain shape. A walking controller that can cope with a few centimeters of error in terrain shape measurement is achieved by combining the generation of a 40-ms cycle online walking pattern and a ground reaction force controller with sensor feedback. An operational interface was developed to send commands to the robot. A mixed-reality display was adopted to realize an intuitive interface. The navigation system was implemented on the HRP-2, a full-size humanoid robot. The performance of the proposed system for navigation over unknown rough terrain and the accuracy of the terrain shape measurement were investigated through several experiments.

Keywords

Humanoid robots, human-centered and life-like robots, legged robots, design and control, range sensing, sensing and perception, computer vision, mapping, mobile and distributed robotics, SLAM

1. Introduction

Bipedal robots are considered to be suitable for use on terrain having obstacles, discontinuous height changes, and roughness. On the other hand, since bipedal robots are naturally unstable, enabling a bipedal robot to walk over such terrain is a challenging problem. The solution to this problem requires an accurate perception system, a path planner that decides step placement, and a robust walking controller.

We developed a walking controller for a humanoid robot capable of handling unknown roughness, such as an unknown height change of up to a few centimeters and an unknown inclination of up to 10° (Nishiwaki and Kagami, 2009). Recently, small scanning-type laser range sensors that can be mounted on a humanoid robot have become available, and an accuracy of a few centimeters is achieved when observing an area a few meters away from the robot. We use this performance matching to realize the navigation of a humanoid robot over unknown rough terrain.

Integrations of perception and online walking control for avoiding obstacles while walking on a horizontal flat floor were reported in Nishiwaki et al. (2003), Kuffner et al. (2003), and Buschmann et al. (2010). An integration of perception and online walking control, which also realized stepping up onto obstacles, was reported by Cupec et al. (2005). The authors introduced constraints on the environment, such as there being only straight edges. They achieved accurate modeling of the environment by introducing these constraints. Stereo-camera systems were used for these integrations.

Successful integration of on-board sensing of the terrain shape, footstep planning, and walking control were reported using a stereo-camera system (Gutmann et al., 2005), a laser range sensor (Chestnutt et al., 2009b), and a monocular camera (Michel et al., 2007). In order to achieve sufficient measurement accuracy, the realizations in Gutmann et al. (2005) and Chestnutt et al.(2009b) operated under the assumption that the environment consists of horizontal plane segments. The integration in Michel et al. (2007)

[1] Digital Human Research Center, National Institute of Advanced Industrial Science and Technology (AIST), Japan
[2] Japan Science and Technology Agency, CREST, Japan
[3] Boston Dynamics, Inc., USA

Corresponding author:
Koichi Nishiwaki, Digital Human Research Center, National Institute of Advanced Industrial Science and Technology (AIST), 2-3-26, Aomi, Koto-ku, Tokyo 135-0064, Japan.
Email: k.nishiwaki@aist.go.jp

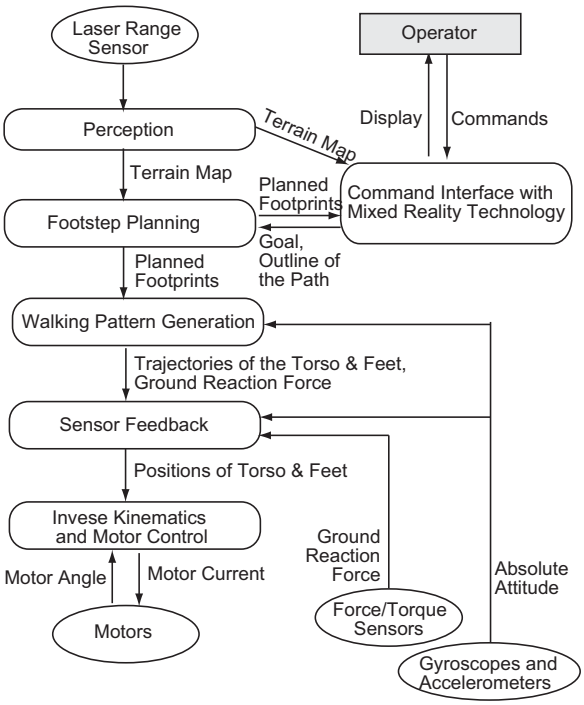

Fig. 1. Overview of the autonomous navigation system.

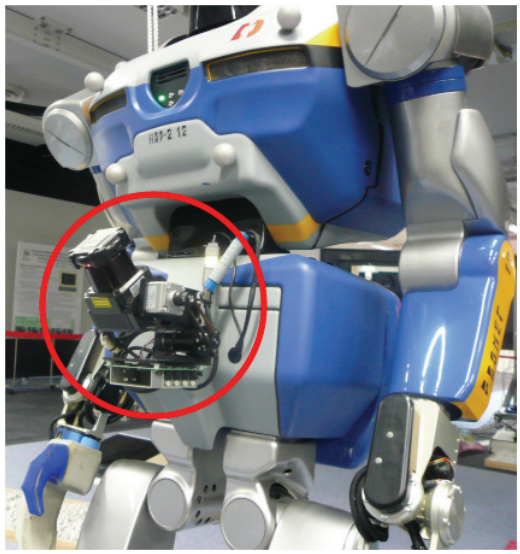

Fig. 2. Scanning-type laser range sensor with a swinging mechanism mounted on the torso of a robot.

requires advance knowledge of the shapes of the objects in the environment. The advances in the system proposed in this paper are that the assumptions or previous knowledge of the shapes of the objects in the environment are not used, and stepping on inclined terrain is also allowed. The key contributions that realized the advances are: (a) a robust walking control system that allows perception errors and (b) a footstep planning method that can evaluate the quality of a foot placement from a grid height map, which contains measurement errors.

Although the shape of the surrounding environment can be obtained with sufficient accuracy online, it is difficult to implement knowledge of the environment, such as identifying objects that should not be stepped on. In addition, we do not assume the existence of a global map. We herein present an interface for assigning commands to the robot using the high-level knowledge of a human operator. An outline of the path or movement direction are commands given to the robot, and the system plans the locations of steps locally using the terrain shape obtained. The planned footsteps can be checked through a mixed-reality interface.

Figure 1 shows an overview of the autonomous navigation system. We describe each component in Sections 2 through 5. Experiments using the full-size humanoid HRP-2 robot are described in Section 6, followed by a discussion and conclusions.

2. Laser-based perception system

This section describes terrain shape map generation using a laser range sensor. The terrain map for the footstep planning is represented as a grid of cells. Each cell has a height value and an information value that indicates whether the height of the cell is observed. A cell size of 0.02×0.02 m is used for the system presented herein.

We adopted the UTM-X002S (Hokuyo Automatic Co Ltd) as a scanning-type laser range sensor. The scanning frequency is 100 Hz, and the angular resolution is 0.625° . The UTM-X002S can perform measurement from a distance of up to 30 m. The sensor is attached to a swinging mechanism, which is mounted on the torso of a humanoid robot as shown in Figure 2.

2.1. Basic strategy

Distance data from a single sweep of the swinging mechanism is gathered and converted into absolute 3D positions using the position of the robot and the angle of the swinging mechanism. The height of each cell is calculated by averaging the height of the 3D positions that are inside the cell region. The map is updated by using the height information of the newer sweep for observed cells. We used this approach in the current implementation because it can better handle changes in the environment and the expected accuracy of the measurement relative to the robot position will be better when the point is measured from nearby.

The results of an experiment to observe terrain shape on a flat office floor while standing still is shown in Figures 3 and 4. Figure 3 (left) shows the experimental set-up. The map region is limited to the colored area of the floor (a 5×5 m rectangle). The sensor swung between 15° and 90° from the horizontal plane. Each sweep (a single upward or downward scan) required 1 s. Figure 3 (right) shows the cells whose heights were observed in a single sweep. The calculated heights of the cells with respect to the distance from

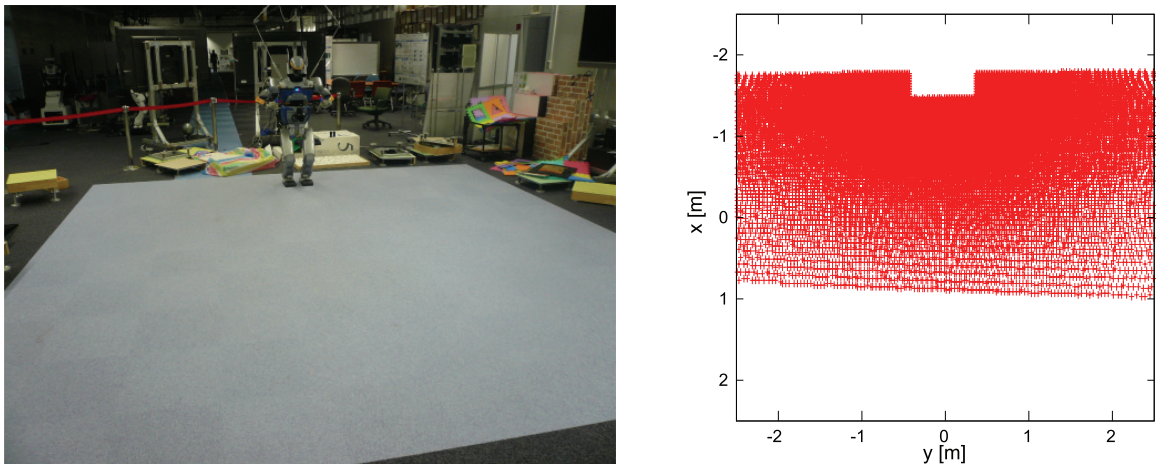

Fig. 3. Terrain shape measurement of a flat office floor. Left: Photograph of the experimental set-up. Right: Observed cells.

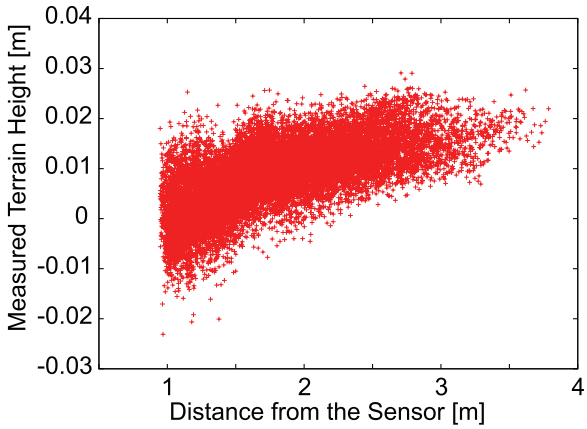

Fig. 4. Distribution of observed height (0 m is the true value).

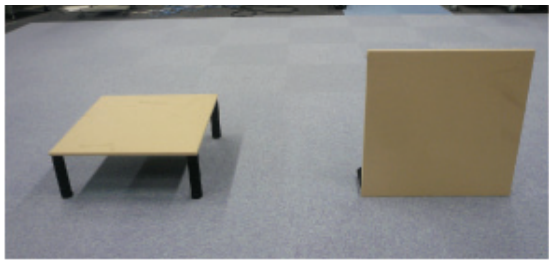

Fig. 5. An example of a terrain with regions of different heights and a wall. (The viewpoint is approximately the same as the position of the laser range sensor from which the data in the following three figures was collected.)

the laser range sensor are plotted in Figure 4 (0 m is the true value).

2.2. *Managing walls and cells with multiple heights*

The average heights of the measured positions are used to reduce the effect of random noise-type errors of the adopted laser range sensor. However, this does not lead to the desired

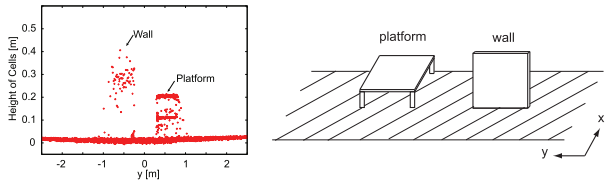

Fig. 6. Calculated heights of cells (left) for the given region (right).

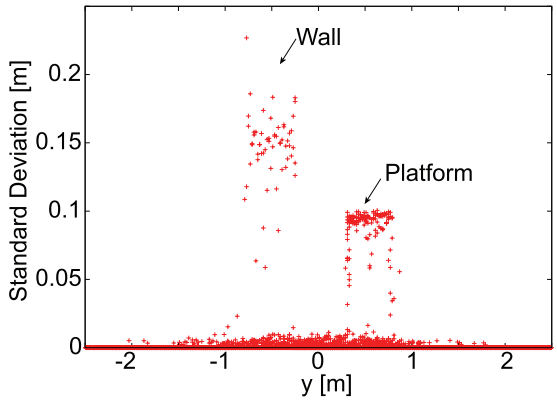

Fig. 7. Standard deviation for each cell.

results for cells with multiple heights or where there are walls, as shown in Figure 5. Figure 6 (left) shows the heights of the cells in the hatched area shown in Figure 6 (right). The heights are plotted along the y-axis. The height of the wall was 0.5 m and that of the platform was 0.19 m. False height values were calculated for the platform and the wall since all the measured values were averaged for each cell. Figure 7 shows the standard deviation of the height of each cell. The value is approximately less than 0.01 m for the horizontal floor, and it becomes large for the multiple height region and the wall. (The standard deviation is 0 for cells where only one point was measured.)

The desired value for each cell height is the highest point inside the cell since it is used to generate the height map

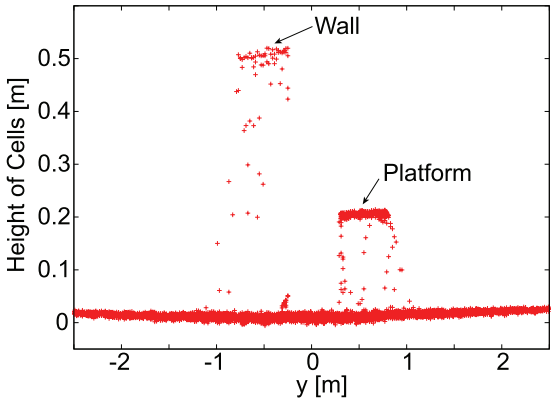

Fig. 8. Calculated heights of cells using the improved method.

for footstep planning. We developed a method, described below, to meet this requirement as well as reducing the random noise effect. The procedure to determine the height of a cell from the 3D positions inside the cell obtained by a single sweep is:

1. If the standard deviation of the heights of the positions is less than 0.01 m, the average of the heights is adopted as the height of the cell.
2. If the standard deviation is greater than 0.01 m:
   a. Sort the height values from the largest to the smallest.
   b. Calculate the standard deviation of the top n highest values, starting from $n = 1$.
   c. Repeat step b, incrementing n by 1 while the standard deviation is less than 0.01 m.
   d. Adopt the average of the selected top n height values as the height of the cell.

Figure 8 shows the calculated heights of the cells for the same sensor data used for Figure 6. Figure 6 shows a false intermediate plane at a height of about 0.1 m for the cells in which both the floor and the platform surface were observed by the laser sensor. The heights of the cells with the wall were mainly estimated at between 0.2 m and 0.4 m since the height values of the measured points on the wall were averaged. Figure 8 no longer shows the false intermediate plane in the platform region, and the height values of the cells for the wall region are near 0.5 m. The height representations for the wall and the platform are improved as desired. The remaining false height cells are mainly caused by the false distance data of the laser scanner, which often occurs at the edges of objects where there are discontinuous distance changes.

2.3. *Handling odometry errors of the height component*

The odometry information calculated by the walking controller is used to obtain the absolute position and posture of the laser range sensor. The details for calculating the odometry while walking are described in Section 4. There are no drifts in the rotational angle around the two horizontal axes since the gravitational direction is used for calculating the odometry. There are drifts in the rotational angle around the gravitational axis and the three components of the translational position. These drifts are basically handled by repetitive measurement of the terrain shape relative to the estimated robot position and replanning. We also have heuristic safety margins for handling the drifts and the execution error. This margin does not work well for the vertical position because it is in the direction of stepping.

Though we may introduce general simultaneous localization and mapping (SLAM) for better localizing the robot, we developed a method of compensating for the vertical drift between sweeps using a low-cost calculation. The difference in the heights between two sweeps for commonly observed cells are calculated and the average of the difference is adopted as the drift of the height between the two sweeps. We excluded cells whose standard deviation in either sweep is larger than 0.01 m when calculating the average to give a better estimate.

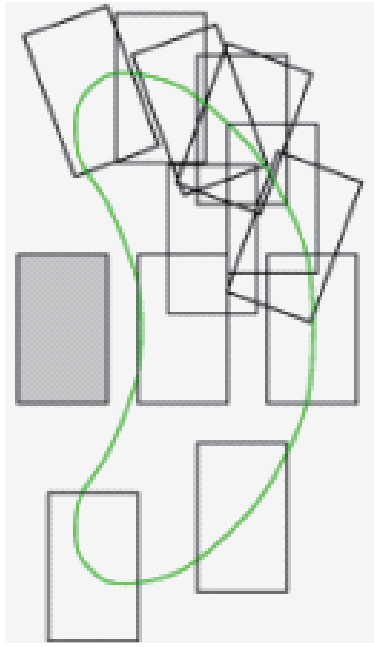

Fig. 9. Example of transition sets for footstep planning.

3. Footstep planning

The footstep planner uses an A* search to generate a sequence of footstep locations for reaching a given goal state. Possible foot transitions are restricted by limiting the landing positions of the swing foot relative to the stance foot to a finite number. An example of a limited transition set is shown in Figure 9. The figure shows the possible landing positions of the right foot as white rectangles and the left stance foot as a gray rectangle. The terrain shape of the stepping position is evaluated. Then, whether the robot can

step on this position is judged and the quality of the position is calculated as the cost of the location. The inclination, roughness, stability, largest bump, and safety of the terrain are used as metrics for both the above-mentioned judgment and the cost calculation (Chestnutt et al., 2003).

Using fewer possible transitions in the candidate set will facilitate making the planner available online. On the other hand, if the number of possible transitions is too small, a sequence of footprints that traverses rough terrain will not be found because possible landing positions will be limited by the shape of the terrain. We proposed an adaptive action model to address this problem (Chestnutt et al., 2007). The basic idea is to try to maintain the number of possible transitions even over rough terrain. If a possible transition is not valid because of the terrain shape, an appropriate transition near the original position is considered as an alternative position. This method generates suitable landing position candidates, which fit the given terrain shape.

In some cases, we prefer to give not only the goal but also an outline of the path, so that the knowledge and intention of the operator can be transferred to the navigation system more clearly. We built a path planner, which takes the guide curve into consideration (Chestnutt et al., 2009a). A heuristic for the A* search is generated from the guide curve so that the search will be carried out along the curve.

Free-leg trajectories, which will not hit the terrain while moving, are planned after deciding the sequence of the footprints.

4. Walking control

The requirements for walking control in an autonomous navigation system are the realization of footprints and the free-leg trajectory assigned online by the footstep planner and managing errors in the terrain shape generated by the perception system.

Humanoid walking has often been realized by constructing a dynamically stable trajectory in advance using dynamics parameters and executing this trajectory with sensor feedback, if necessary. This procedure was adopted because bipedal humanoids have a complicated dynamic model.

In recent years, several studies on the online generation of dynamically stable walking patterns have been published (e.g. Setiawan et al., 1999; Nishiwaki et al., 2001; Yokoi et al., 2001; Kaneshima et al., 2001; Nishiwaki et al., 2002; Löffler et al., 2003; Harada et al., 2004; Sugihara and Nakamura, 2005; Park et al., 2006; Buschmann et al., 2007; Dimitrov et al., 2008). We extended this approach and constructed a system that generates walking patterns at a very high frequency, e.g. a cycle of 40 ms, and which considers the actual motion of the robot as the initial condition of each generation. Maintenance of the dynamic stability of actual walking is realized by this high-frequency generation from the actual motion conditions. We achieved a walking controller that can handle an unknown change of level of a few centimeters or an unknown inclination of up to 10°

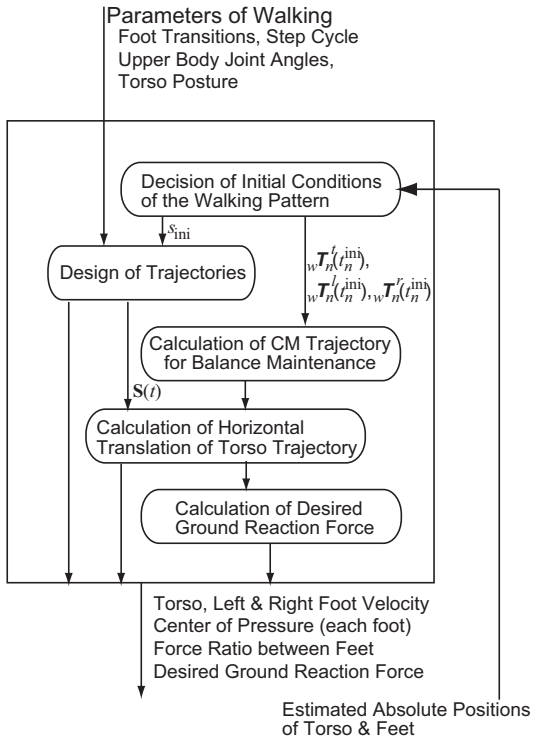

Fig. 10. Overview of dynamically stable trajectory generation.

by combining the high-frequency walking pattern generation system with a ground reaction force controller having sensor feedback (Nishiwaki and Kagami, 2009). This controller was adopted to realize footprints assigned online and to manage perception errors.

4.1. Dynamically stable pattern generation

Figure 10 shows an overview of the dynamically stable trajectory-generation system. The role of this system is to generate dynamically feasible walking trajectories that start from the estimated actual motion.

Since trajectory generation takes some time (we currently start each generation at 36 ms before the starting time of 40 ms-trajectory length), the initial conditions at some point in the future should be estimated using the current motion status. Let t_n^{ini} and $t_n^{est} = t_n^{ini} - T_{est}$ be the start time of the n-th motion and the decision time for the n-th initial condition, respectively ($T_{est} = 0.036$ s in the current implementation). The initial position of the torso in the transformation matrix representation of the n-th trajectory is decided by using the estimated absolute position of the torso at t_n^{est} (${}_wT_{est}^t(t_n^{est})$), as follows

$$ {}_wT_n^t(t_n^{ini}) = {}_wT_{est}^t(t_n^{est}) \left({}_wT_{n-1}^t(t_n^{est})\right)^{-1} {}_wT_{n-1}^t(t_n^{ini}) \qquad (1) $$

The initial positions of the left and right feet are decided in the same manner.

The trajectories for the horizontal torso position are used for maintaining balance. Foot position and posture trajectories and torso height and posture trajectories (**s**(t) in Figure 10) are designed to follow the given command smoothly from the estimated initial conditions.

The initial position of the center of mass is calculated from the estimated initial torso and foot positions ($_wT^t_n(t^{ini}_n)$, $_wT^l_n(t^{ini}_n)$, and $_wT^r_n(t^{ini}_n)$). Then, the desired trajectory of the center of mass (CM) is generated from the reference zero moment point (ZMP) trajectory by applying preview control theory (Kajita et al., 2003b). The desired CM trajectory is converted to a horizontal torso position trajectory by applying a slightly modified version of the resolved momentum control method (Kajita et al., 2003a). Here, the trajectories of the feet and other components of the torso are used in deciding the horizontal torso position.

Finally, inverse-dynamics calculations are carried out to decide the desired ground reaction force necessary to execute the generated motion. The calculated ground reaction force will be the control reference of the sensor feedback in the system.

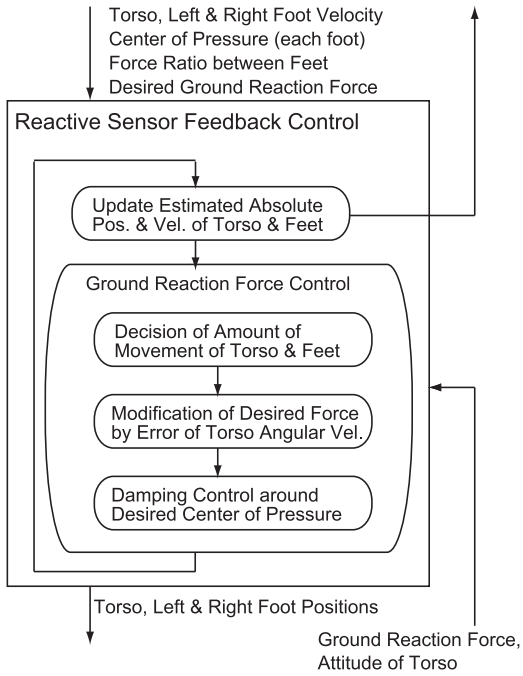

Fig. 11. Overview of the sensor feedback control system.

4.2. Sensor feedback modification

The goal of sensor feedback modification is to realize the specified torso motion in the absolute coordinate system in the short term, even if the terrain shape is different than expected. Gradual divergence from the given trajectories will not be a significant problem because the repetitive trajectory generation compensates for the divergence by using the actual motion for the initial conditions. We basically adopted two types of feedback control in the sensor feedback modification: ground reaction force control and control of the absolute posture of the torso. In order to ensure that the torso motion is not sensitive to differences in the terrain shape, we attempt to control the ground reaction force at the feet rather than the actual foot positions.

Figure 11 shows an overview of the sensor feedback control system. Estimation of the position and posture in the absolute coordinate system is carried out in this loop using a cycle of 1 ms. The estimated information is used to decide the orientation of the coordinate system, which is used for the ground reaction force control, and is sent to the trajectory generation system. The velocities of the torso and the feet are also estimated here, and are sent to the trajectory generation system for deciding the initial conditions of the next pattern generation.

Ground reaction force control is implemented as impedance control of the foot positions with a zero spring coefficient. The inputs of the control are the desired reaction force and velocity. The desired velocity of the feet assigned by the trajectory generation system is modified according to the ground contact phase. The assigned ground reaction force is then modified according to the error in the angular velocity of the torso. This works as a torso posture damping control and attempts to maintain the absolute torso posture specified by the trajectory generation system. Finally, impedance control is carried out using the desired center of pressure as the control point, and the desired foot positions are decided.

Let $\mathbf{p}^l$ be the control point of the left-foot impedance control, which coincides with the desired center of pressure. The translational velocity of $\mathbf{p}^l$ in the absolute coordinate system and the rotation velocity around the two horizontal axes of the absolute coordinate system are decided by the impedance control. Rotation around the vertical axis is not modified by this control in the current implementation.

Let

$$\mathbf{F^l_d} = \left(f^l_x, f^l_y, f^l_z, \tau^l_x, \tau^l_y\right) \tag{2}$$

be the reference force and torque of the controlled components at the control point whose axes are fixed in the absolute coordinate system, and let

$$\dot{\mathbf{Q}}^l = \left(v^l_x, v^l_y, v^l_z, \omega^l_x, \omega^l_y\right) \tag{3}$$

be the reference velocity vector, each component of which corresponds to the desired force and torque vector. The actual reaction force at the control point can be calculated by transforming the output of the 6-axis force sensor.

Let $\dot{q}_d$, f_d, and f be the reference velocity, the reference reaction force, and the measured reaction force of one of the components, respectively. The velocity of the component, $\dot{q}$, is given by

$$f - f_d = m\ddot{q} + d(\dot{q} - \dot{q}_d) \tag{4}$$

Here, m and d are the virtual inertia and damping coefficients for the impedance control, respectively. Then $\ddot{q}(t)$ is

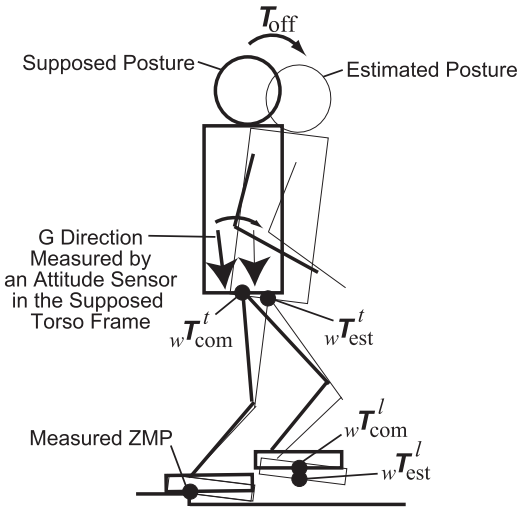

Fig. 12. Estimating the absolute posture.

approximated as follows by using the control cycle Δt (= 1 ms) to discretize the equation.

$$\ddot{q}(t) = \frac{\dot{q}(t) - \dot{q}(t - \Delta t)}{\Delta t} \tag{5}$$

The velocity of the component at every control cycle $\dot{q}(t)$ is given by

$$\dot{q}(t) = \frac{f(t) - f_d(t) + d\dot{q}_d + (m/\Delta t)\,\dot{q}(t - \Delta t)}{d + m/\Delta t} \tag{6}$$

The position and posture of the control point on the left foot are updated using the calculated velocity vector. The position and posture of the right foot are decided in the same manner.

4.3. Estimating the positions and velocities of the torso and feet

The positions and postures of the torso and the feet in the absolute coordinate system are estimated in the sensor feedback control loop in cycles of 1 ms. This is used for updating the coordinate system for sensor feedback control and deciding the initial conditions for the generation of the walking pattern. The position and orientation of the laser range sensor, which are used for generating the height map, are also calculated using the estimated position and posture of the torso.

The commanded positions of the torso and feet as the result of the sensor feedback control are basically used for estimating the positions in the absolute coordinate system. The assumption here is that at least one foot is in contact with the ground and there is no translational slippage between the robot and the ground. Additionally, the absolute torso posture measured by an inertial measurement unit (IMU) is used to compensate for the effect of unexpected rotation of the robot caused by unstable contact with the ground. The IMU consists of three closed-loop fiber optic gyroscopes and three servo accelerometers, and outputs the orientation of the torso frame in the absolute coordinate system. (There is a drift around the vertical axis due to the nature of the IMU.) There are drifts in the translational positions and rotation around the vertical axis, but there are no drifts in the rotation around the two axes in the horizontal plane. Moreover, the relative positions and postures between the torso and the feet in the absolute coordinate system are not affected by the drifts. This consistency is important when using the estimated positions for walking control.

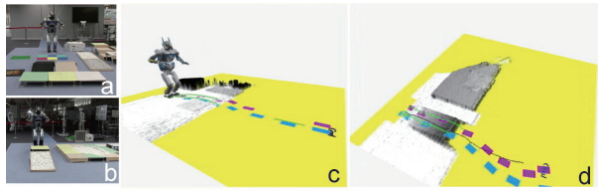

Fig. 13. Graphical user interface for navigation control.

Assuming that the unintentional rotation of the robot occurs around the measured ZMP, the following procedure is adopted for the estimation:

1. Update the position and posture of the torso and the feet as commanded in the last control loop.
2. Evaluate the difference of the torso posture between the updated one and the IMU-measured one.
3. Rotate the robot around the measured ZMP in order to make these two postures coincide (Figure 12).

The velocities of the torso and the feet are estimated as follows using the same assumption:

1. The rotational velocity of the torso in the absolute coordinate system is directly calculated by the outputs of the gyroscopes and the estimated torso posture.
2. The translational torso velocity is estimated using the assumption that the error of the rotational velocity of the torso from that commanded is caused by the rotation around the measured ZMP.
3. The velocities of the feet are calculated using the estimated torso velocity assuming that the motor control is ideal.

5. Operational interface

Three different operational interfaces for giving commands to the autonomous navigation system are presented in this section.

5.1. Graphical user interface

The most basic interface specifies the goal on the terrain map obtained using a GUI (see Figure 13(c) – the corresponding experimental set-up is shown in Figure 13(a)). An outline of the path can also be specified using this interface (see Figure 13(d) – the corresponding experimental

Fig. 14. Mixed-reality interface for navigation control.

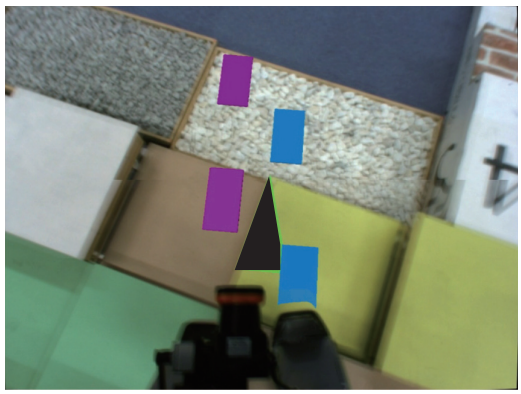

Fig. 15. Joystick interface for navigation control.

set-up is shown in Figure 13(b)). The operator's knowledge of the terrain and the preferred route can be transferred to the navigation system by providing an outline of the path. A problem with this interface is that the goal may be in an area where the terrain shape has not been obtained, as shown in Figure 13. The operator needs to determine the target environment by some other means and its correspondence to the map shown in the GUI system.

5.2. MR interactive interface

The second interface is a mixed-reality (MR) interface. The operator wears a head-mounted display (HMD) and uses a game controller. The positions of both the HMD and the controller are localized by the optical motion capture system (Motion Analysis Corp. Eagle Digital System) in the same absolute coordinate system as that in which the robot is localized. The operator can draw an outline of a path or specify a goal using the controller, and the drawn path and the goal are superimposed on the HMD view, as well as the planned path (Figure 14). Commands to the robot, such as 'start walking', can be assigned by pointing at the virtual menu using the same interface, as shown in Figure 14 (right). The operator can specify the goal or the outline path intuitively even if the target position has not yet been observed by the perception system. A drawback of this interface is that the operator must be in the operational area in order to specify the positions.

5.3. Joystick interface

In order to overcome the drawback of the previous interface, another mixed-reality interface was introduced. The camera view of the robot is displayed on a monitor, and the desired direction and rotation of locomotion are assigned by manipulating the triangle shape in the view using a joystick (Figure 15 – the triangle is colored black in the figure for better visibility). The footstep planner is used for local path planning rather than global path planning. The footstep planner plans a sequence of a few steps that realizes the desired movement as well as adaptation to the local terrain shape. The interface realized remote control of the autonomous locomotion. However, there is a drawback in that the operator has to become more involved with the locomotion control.

6. Experiments

We implemented the proposed control system on the HRP-2, a full-size 38-DOF humanoid robot. Terrain map building and footstep planning were carried out on a computer external to the robot, and the laser range sensor data and the planned result were sent from/to the robot via an Ethernet. At each step the robot takes, the planner begins to compute a new plan based on the position of the robot and the step that is being taken. By computing a new plan at each step, the robot can use the recent perception results and make adjustments for slips or other deviations from the plan. Footstep planning is repeated more frequently with the joystick interface in order to achieve a better response to the change of the joystick input.

We installed a Core 2 3.06-GHz CPU board in the robot. Dynamically stable walking pattern generation and sensor feedback control were implemented on the CPU together with sensor processing and motor servo control. The dynamically stable trajectory generation runs at 40-ms cycles and requires approximately 30 ms to generate a pattern of 40 ms in length. At the same time, the sensor feedback control runs at a 1-ms cycle for executing the generated trajectories.

6.1. Navigation experiments

Photographs of experiments to investigate autonomous navigation over previously unknown rough terrains are shown in Figures 16, 17, and 18, and corresponding video clips are provided as Extensions 1, 2, and 3. Each experiment uses one of the different interface system described in the previous section. An optical motion capture system was used for localizing the robot's position during the experiments in this subsection.

In the experiment shown in Figure 16 and Extension 1, pebbles were placed on the slope. The GUI was used to assign the commands. The GUI view is superimposed in the top right corner of each photograph. Since the operator knows the environment, the operator drew an outline of the path, including the area in which the height was unknown. The terrain map was incrementally built during walking, and the map obtained was used online for planning the

Fig. 16. Photographs of an experiment on autonomous navigation over unknown terrain (GUI) (see also Extension 1).

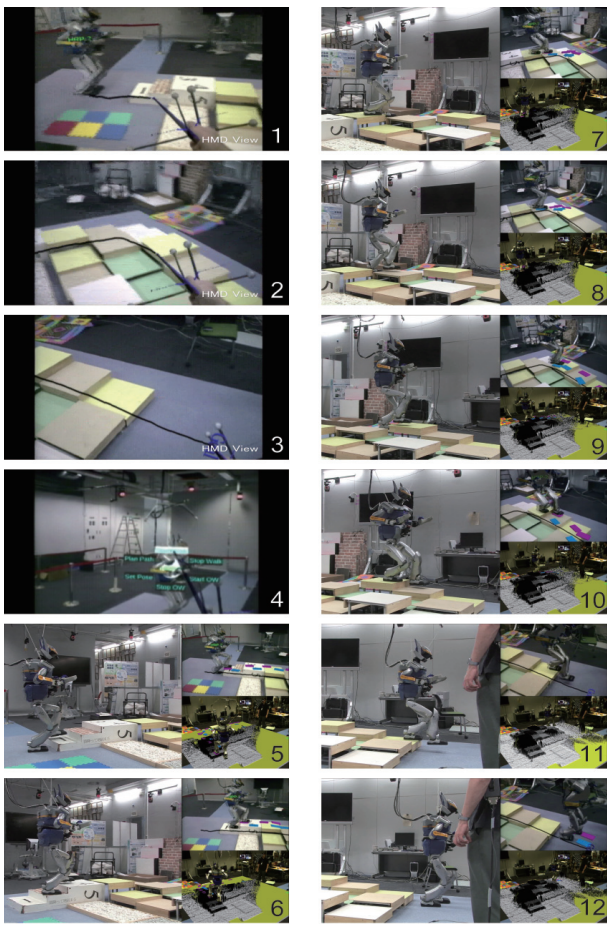

Fig. 17. Photographs of an experiment on autonomous navigation over unknown terrain (mixed-reality interface) (see also Extension 2).

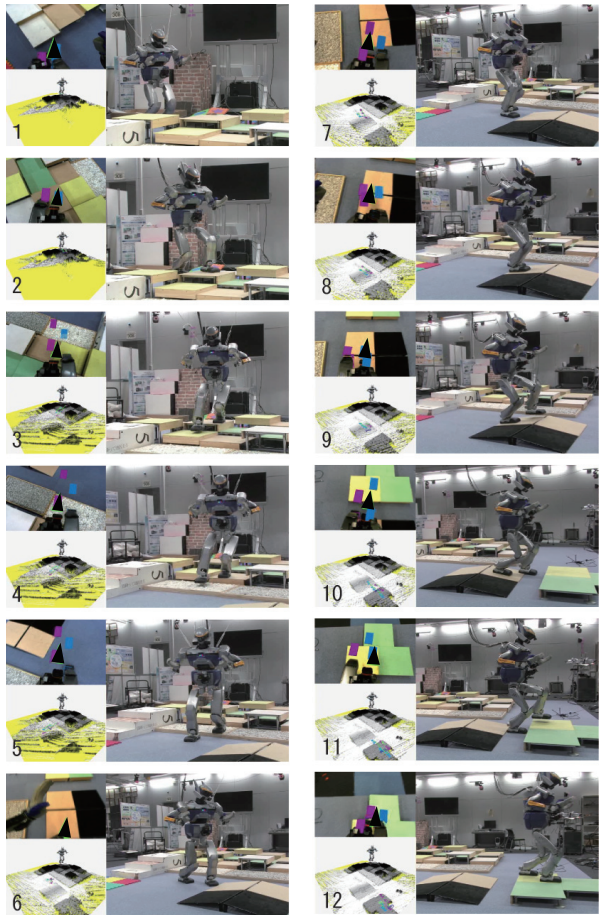

Fig. 18. Photographs of an experiment on autonomous navigation over unknown terrain (joystick interface) (see also Extension 3).

footprints that fit the terrain. Online incremental map building and replanning worked well, and the robot successfully reached the goal by approximately following the provided outline.

Figure 17 and Extension 2 show the case in which the mixed-reality interface was used. The view of the operator through the HMD is shown in the first four photographs and is superimposed in the top right corner of each of the other photographs. Starting from the fifth photograph, the view superimposed in the bottom right corner shows the obtained terrain map superimposed on the image captured by the ceiling camera. The robot successfully walked through the previously unknown rough terrain by approximately following the provided outline.

An autonomous navigation experiment involving the joystick interface was conducted as shown in Figure 18 and Extension 3. The view shown to the operator is superimposed in the top left corners of the photographs. The triangle indicates the commanded direction and rotation. (The triangles are colored black in the figure for better visibility.) The map obtained and the planned footprints projected onto the map are shown in the bottom left corner of the photographs. The operator successfully guided the robot using

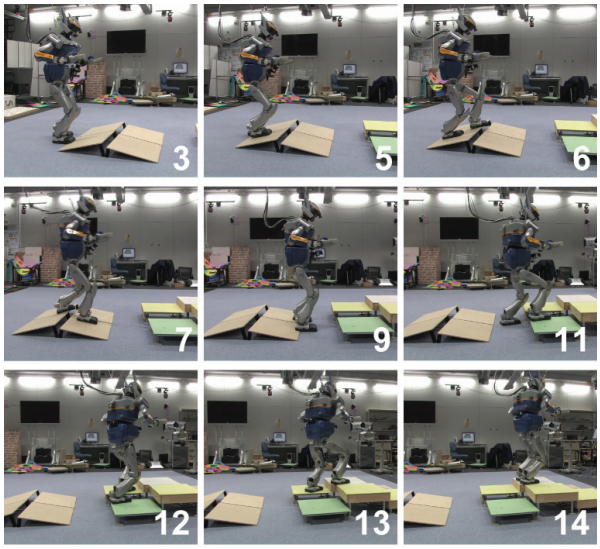

Fig. 19. Evaluating the accuracy of terrain measurement (see also Extension 4). The number in each picture is the number of the step just landed.

the joystick interface by watching only the view shown at the top left.

6.2. *Evaluation of measurement accuracy for height and inclination*

We then carried out an experiment to evaluate the accuracy of terrain shape measurement while executing autonomous navigation over unknown rough terrain. A rough terrain was constructed using rigid platforms and slopes whose dimensions are accurately known. Figure 19 and Extension 4 show scenes from the experiment. Since the ground truth for each stepping position is available for the height and the inclination, we evaluated those values.

The commanded stepping height relative to the previous step and the commanded absolute inclination relative to horizontal are shown in Table 1. The ground-truth values of inclination are not available for steps that would be on the edges of a slope (e.g. Steps number 3 and 9 in Figure 19). The ground-truth values for relative height are not available when the step or the previous step are on a slope (including the edges). The largest error of the absolute value of the inclination was 0.180 rad, which was for the sixth step. The step was an overhanging step and the area for calculating the inclination was smaller than the normal steps. The inclination errors for the other steps are rather small. (The second largest was 0.067 rad.) The largest error of the absolute value of the height was 11.4 mm, which was for the 13th step, one of the climbing up onto the platform steps. We confirmed that the errors of the inclination and the height stayed within the allowed limits for the walking control system adopted for this autonomous navigation.

7. Conclusion

An integrated system of online perception, footstep planning, and walking control for a humanoid robot for navigating on previously unknown rough terrain was proposed. A laser-based perception system realized online measurement of the surrounding terrain shape with an accuracy of a few centimeters. The robust walking control enabled a humanoid robot to walk over previously unknown rough terrain with an unknown roughness of a few centimeters. Operational interfaces using a mixed-reality display enabled commands to be fed to the system intuitively. Several autonomous navigation experiments on the HRP-2 showed the performance of the proposed system. We also confirmed that the perception error of the terrain shape meets the requirements and can be handled by the proposed robust walking controller.

We achieved autonomous navigation of a humanoid on unknown rough terrains without using any assumptions or previous knowledge of the shapes of the objects in the environment. One implicit assumption is that the stepping positions do not deform significantly. Another implicit assumption is that the environment is not so complicated as to cause collisions between the robot and the environment other than its sole (e.g. knee collisions with the platform edges and arm collisions with high obstacles were not considered). One reason for introducing interfaces to supervise the autonomy was to manage these limitations.

Localizing the robot using the laser sensor data as well as the construction and use of a larger-scale map will be investigated in future research. Reactively changing the stepping position and timing for maintaining the balance of walking becomes important when the supposed ground reaction force cannot be obtained at the stepped position, such as for deformable terrain. Developing a footstep planner that provides the walking controller with a changeable region of stepping position, which can be used for balance recovery during actual walking, is another future research topic.

Funding

This research was partly supported by JST, CREST.

References

Buschmann T, Lohmeier S, Bachmayer M, Ulbrich H and Pfeiffer F (2007) A collocation method for real-time walking pattern generation. In: *Proceedings of international conference on humanoid robots*, pp. 1–6.

Buschmann T, Lohmeier S, Schwienbacher M, Favot V, Ulbrich H, Hundelshausen F, Rohe G and Wuensche HJ (2010) Walking in unknown environments – a step towards more autonomy. In: *Proceedings of international conference on humanoid robots*, pp. 237–244.

Chestnutt J, Kuffner J, Nishiwaki K and Kagami S (2003) Planning biped navigation strategies in complex environments. In: *Proceedings of international conference on humanoid robots*.

Table 1. Accuracy of commanded stepping heights and inclinations.

Step number		1	2	3	4	5	6
Relative Height	Commanded	–5.0	7.0	12.2	2.2	57.6	40.4
(mm)	Ground truth	0.0	0.0	N/A	N/A	N/A	N/A
	Error	–5.0	7.0	–	–	–	–
Inclination	Commanded	4.9	2.5	4.3	20.1	20.1	8.8
(10^{-2} rad)	Ground truth	0.0	0.0	N/A	N/A	26.9	26.9
	Error	4.8	2.5	–	–	–6.7	**–18.0**

7	8	9	10	11	12	13	14	15	Abs. Avg.
–38.0	–71.2	–2.8	12.8	109.2	61.4	88.7	–11.0	5.6	
N/A	N/A	N/A	N/A	108.0	60.0	100.0	0.0	0.0	
–	–	–	–	1.2	1.4	**–11.4**	–11.0	5.6	6.1
16.3	16.3	7.9	3.0	1.0	6.3	0.6	3.9	1.5	
18.8	N/A	N/A	0.0	0.0	0.0	0.0	0.0	0.0	
–2.6	–	–	3.0	1.0	6.3	0.6	3.9	1.5	4.8

Chestnutt J, Nishiwaki K, Kuffner J and Kagami S (2007) An adaptive action model for legged navigation planning. In: *Proceedings of international conference on humanoid robots*, pp. 196–202.

Chestnutt J, Nishiwaki K, Kuffner J and Kagami S (2009a) Interactive control of humanoid navigation. In: *Proceedings of international conference on intelligent robots and systems*. pp. 3519–3524.

Chestnutt J, Takaoka Y, Suga K, Nishiwaki K, Kuffner J and Kagami S (2009b) Biped navigation in rough environments using on-board sensing. In: *Proceedings of international conference on intelligent robots and systems*, pp. 3543–3548.

Cupec R, Schmidt G and Lorch O (2005) Vision-guided walking in a structured indoor scenario. *Automatika* 46(1–2): 49–57.

Dimitrov D, Ferreau J, Wieber P and Diehl M (2008) On the implementation of model predictive control for on-line walking pattern generation. In: *Proceedings of international conference on robotics and automation*, pp. 2685–2690.

Gutmann J, Fukuchi M and Fujita M (2005) Real-time path planning for humanoid robot navigation. In: *Proceedings of international joint conferences on artificial intelligence*, pp. 1232–1237.

Harada K, Kajita S, Kaneko K and Hirukawa H (2004) An analytical method on real-time gait planning for a humanoid robot. In: *Proceedings of international conference on humanoid robots*, no. 60.

Kajita S, Kanehiro F, Kaneko K, Fujiwara K, Harada K, Yokoi K and Hirukawa H (2003a) Resolved momentum control: Humanoid motion planning based on the linear and angular momentum. In: *Proceedings of international conference on intelligent robots and systems*, pp. 1644–1650.

Kajita S, Kanehiro F, Kaneko K, Fujiwara K, Harada K, Yokoi K and Hirukawa H (2003b) Biped walking pattern generation by using preview control of zero-moment point. In: *Proceedings of international conference on robotics and automation*, pp. 1620–1626.

Kaneshima Y, Sugahara Y, Ando S, Sato M, Lim H and Takanishi A (2001) Quasi real-time motion pattern generation for bipedal humanoid robots. In: *Proceedings of annual conference of the Robotics Society of Japan*, pp. 987–988 (in Japanese).

Kuffner J, Kagami S, Nishiwaki K, Inaba M and Inoue H (2003) Online footstep planning for humanoid robots. In: *Proceedings of international conference robotics and automation*, pp. 932–937.

Löffler K, Gienger M and Pfeiffer F (2003) Sensor and control design of a dynamically stable biped robot. In: *Proceedings of international conference on robotics and automation*, pp. 484–490.

Michel P, Chestnutt J, Kagami S, Nishiwaki K, Kuffner J and Kanade T (2007) GPU-accelerated real-time 3D tracking for humanoid locomotion and stair climbing. In: *Proceedings of international conference on intelligent robots and systems*, pp. 463–469.

Nishiwaki K and Kagami S (2009) Frequent walking pattern generation that directly uses estimated actual posture for robust walking control. In: *Proceedings of international conference on humanoid robots*, pp. 535–541.

Nishiwaki K, Kagami S, Kuffner J, Okada K, Kuniyoshi Y, Inaba M and Inoue H (2003) Online humanoid walking control and 3D vision-based locomotion. In: Experimental robotics VIII, Siciliano B and Dario P Eds., Springer, pp. 85–94.

Nishiwaki K, Kagami S, Kuniyoshi Y, Inaba M and Inoue H (2002) Online generation of humanoid walking motion based on a fast generation method of motion pattern that follows desired ZMP. In: *Proceedings of international conference on intelligent robots and systems*, pp. 2684–2689.

Nishiwaki K, Sugihara T, Kagami S, Inaba M and Inoue H (2001) Realtime generation of humanoid walking trajectory by mixture and connection of pre-designed motions – online control by footprint specification. In: *Proceedings of international conference on robotics and automation*, pp. 4110–4115.

Park I, Kim J, Lee J and Oh J (2006) Online free walking trajectory generation for biped humanoid robot KHR-3 (HUBO). In: *Proceedings of international conference on robotics and automation*, pp. 2667–2672.

Setiawan S, Hyon S, Yamaguchi J and Takanishi A (1999) Physical interaction between human and a bipedal humanoid robot – realization of human-follow walking. In: *Proceedings of international conference on robotics and automation*, pp. 361–367.

Sugihara T and Nakamura Y (2005) A fast online gait planning with boundary condition relaxation for humanoid robots. In: *Proceedings of international conference on robotics and automation*, pp. 306–311.

Yokoi K, Kanehiro F, Kaneko K, Fujiwara K, Kajita S and Hirukawa H (2001) A Honda humanoid robot controlled by AIST software. In: *Proceedings of international conference on humanoid robots*, pp. 259–264.

Appendix A: Index to multimedia extensions

Extension	Media Type	Description
1	Video	Autonomous navigation over unknown terrain (GUI) (corresponds to Figure 16)
2	Video	Autonomous navigation over unknown terrain mixed-reality interface) (corresponds to Figure 17)
3	Video	Autonomous navigation over unknown terrain (joystick interface) (corresponds to Figure 18)
4	Video	An experiment for evaluating the accuracy of terrain measurement (corresponds to Figure 19)