The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Biped Navigation in Rough Environments using On-board Sensing

Joel Chestnutt†, Yutaka Takaoka‡, Keisuke Suga‡, Koichi Nishiwaki†, James Kuffner*†, and Satoshi Kagami†

†Digital Human Research Center
AIST
2-41-6 Aomi, Koto-ku
Tokyo 135-0064, Japan
{joel.chestnutt, s.kagami}@aist.go.jp

‡Partner Robot Division
TOYOTA Motor Corporation
Toyota-cho, Toyota
Aichi, 471-8571, Japan
{yutaka_takaoka, keisuke_suga}
@mail.toyota.co.jp

*The Robotics Insitute
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213, USA
kuffner@cs.cmu.edu

Abstract— **We present an approach to navigating a biped robot safely and efficiently through a complicated environment of previously unknown obstacles and terrain using only on-board sensing and odometry. Sensing of the environment is performed by a pivoting laser scanner, which continues to update the terrain representation as the robot walks. Safe stepping motions are planned from this data to follow the user's command, given in the form of an end goal, a rough path, or a joystick input. Results are demonstrated on a prototype robot in several environments.**

I. Introduction

One of the motivations for developing humanoid and legged robots is to obtain robots which are capable of going anywhere that a human can go, and performing the same kinds of tasks with the same tools that humans can use. Toward this end, research into stable and robust walking for biped robots is still a very active area. However, even when stable walking algorithms exists for a particular robot, safely navigating through real-world environments requires the ability to sense the environment and chose safe motions with respect to the terrain and obstacles present.

For complex indoor environments designed for humans, this problem includes dealing with furniture, walls, stairs, doors, and previously unknown obstacles on the floor. For outdoor environments, this includes the ability to navigate on rough terrain and uneven surfaces. Because legged robots have the ability to step over and onto obstacles in their path, they are uniquely suited to overcoming these difficulties. However, many existing navigation planning methods fail to consider these additional capabilities, because they were primarily designed for wheeled mobile robots.

In the case of on-board perception, a walking robot presents several concrete challenges. Approaches to robot localization and environment mapping must deliver accurate results to comply with the small error tolerances imposed by the walking controller if the robot is to successfully step onto surfaces or avoid obstacles. Moreover, rapid scene changes, large displacements and shakiness occur naturally with quickly moving highly articulated humanoids, and must be handled by the sensor system. Also, unlike for wheeled robots, pausing movement for deliberation or to gather sensor readings is usually not an option — perception must operate in real-time.

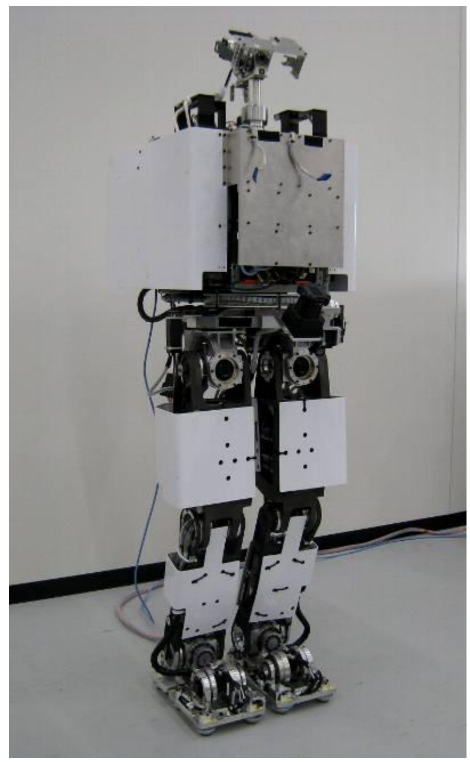

Fig. 1. The prototype biped robot

In this paper, we present methods for safely navigating previously unknown environments, using on-board sensing which operates continuously while the robot walks. The environment does not require a particular texture or shape of obstacles, and the robot does not require any knowledge about the obstacles or type of terrain in advance of sensing the scene. In addition, the sensing and planning operate at the same speed as the robot's walking cycle, so that the robot does not need to stop or interrupt its walking motion to sense or plan.

II. Background

Reliable walking biped robots have been developed only recently, although today there are several humanoid robots in use around the world. For these robots, developing sensing methods, navigation strategies, and walking control is still an ongoing area of research.

Due to the difficulties in sensing from a walking biped, relatively few humanoid robots have been able to overcome challenging terrain. The H7 humanoid has successfully climbed stairs[1], but this was after manual positioning in front of them, without sensing their position. H7 was later able to detect and avoid obstacles using stereo vision[2], but was unable to step on different level terrain using the stereo data.

Honda's ASIMO humanoid [3] first positions itself precisely with respect to a set of stairs equipped with fiducials and then executes fixed footstep sequence, adjusted according to the contact force with each step, to climb them. Additionally, off-board sensing has been used with color-segmentation for the identification and tracking of flat obstacles[4], allowing ASIMO to navigate through obstacle-filled environments as well as predictive avoidance[5].

The Johnnie robot was able to reactively avoid obstacles and climb stairs using on-board vision that found obstacle edges in the camera views[6], [7], [8].

Sony's QRIO robot uses stereo to reconstruct stairs and climb them gradually, step-by-step[9], [10], [11].

HRP-2 has been demonstrated overcoming stairs, platforms and other obstacles using off-board sensing through a motion capture system[12], as well as on-board edge-based visual tracking[13]. Both cases are able to provide very accurate 3D data about the environment, but require advance knowledge about the shape and number of obstacles present in the scene.

III. The Biped Robot

The robot used in these tests, shown in Figure 1, is a prototype biped with a height of 1.46m and a weight of 43.5kg. The robot has 6 degrees of freedom for each leg, plus a toe joint on each foot. Each foot has a size of 29cm×17cm. The step cycle for our trials was approximately 1 second.

Sensing of the environment was performed by an on-board Hokuyo laser scanner (shown in Figure 2), which swiveled up and down while operating.

By combining the scan data with the orientation of the scanner, the position of the robot using odometry, and the orientation of the robot using gyro sensor, a 3D point cloud of the environment can be constructed. The scanning was synchronized with the walking cycle, resulting in one sweep of the scanner per step. The dense area around the robot was used for plane and obstacle detection (about 24,000 points, seen on the right in Figure 3). This sensing approach has a high accuracy, and does not depend on having any preset models of the obstacles or any specific texture on the surfaces in the environment. However, the scanning speed is slow (approximately one second per scan), and the process of combining all the scan data into one 3D point cloud assumes that the world is static during the full scan.

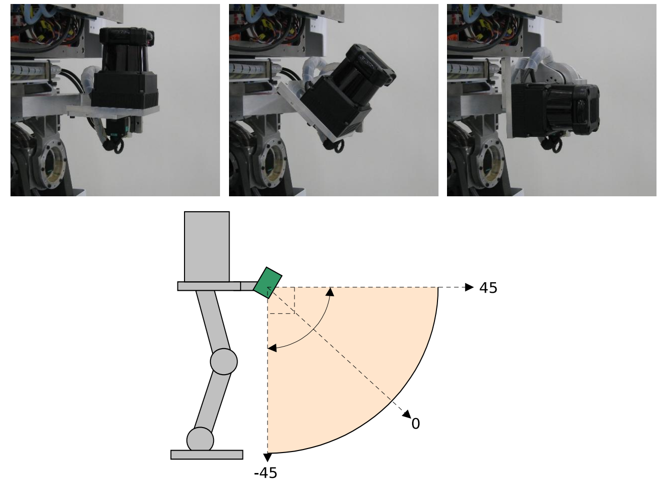

Fig. 2. The laser scanner for gathering environment information. The scanner continuously pivots up and down.

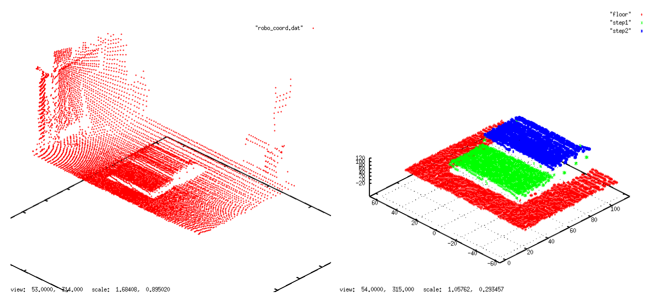

Fig. 3. The point cloud laser scans and the subsequently identified planes.

When the robot makes a step or moves quickly to avoid an obstacle, any vibration or slipping of the robot become significant problems for sensing accuracy. Correct 3D data is not obtained under those circumstances using only odometry and the gyro sensor. Instead, we extract the noisy 3D data using an acceleration sensor in the robot. When the acceleration sensor senses that the robot is unstable, we remove that noisy 3D data and use a previous scan data for recognition.

IV. Environment Representation

From the 3D laser data obtained during a step, a filtered height map representation of the environment is constructed. The terrain area is divided into a grid (with a cell size of 1.5cm in our trials). The filtering process finds planes in the environment, and each cell in the grid contains a height, as well as the id of the particular plane it is identified with. Cells that do not match any plane are identified as obstacles. Each plane detected in the environment also specifies a vector for its normal.

A. Plane Detection

From a 3D point cloud of the environment, we extract planes that have sufficient size and a low enough angle that the robot can safely step onto them and store that data in our height map representation. For finding these planes, we use a method of two-points random sampling[14]. Normal vectors of each plane are computed by the angle made from the two points that are sampled randomly from the 3D data. Using the normal vectors, partial planes are found with respect to each point's distance from the origin. As a result, 3D points on the plane are identified as a plane. Figure 3 shows an example of an input point cloud from the pivoting laser scanner, and the resulting identification of planes within that data.

The plane detection process has a high computational cost, so we restrict the detector to only find planes with a low enough angle ($\pm 30^\circ$ from horizontal in this paper) so that the robot may be able to step onto them. For fast random sampling, we use a template that has information about where to sample. These templates are made in an offline process from various environments (floor, stairs, table, obstacles) to sample points randomly. During online plane-detection, the best template is selected for the environment. From this template, many two-points sets are randomly sampled faster than usual. These methods allow the robot to quickly recognize planar segments and obstacles in the environment online during its walking.

B. *Creating the environment map*

In this paper, we construct a height map representation of the environments with steps, chairs, boxes, and tables. Generally it is very important for a biped robot to observe its surrounding environmental surface in dense 3D and to obtain high accuracy data. For this robot, it is difficult to sense the terrain immediately in front of its toes due to the position and motion of the scanner. As a result we use a wider map that is connected by sequential 3D data scans using robot odometry. Due to the accumulated error of robot odometry, 3D data older than a certain time is deleted. Each cell in the map stores a value about the last time its data was observed. This value is lower when the data is more recent. Using this accumulated map, the robot can walk to and step up or down into its sensory blind spot.

For data which is observed by the scanner but does not fit any of the detected planes, we mark the cells of the map as obstacles.

The plane detection is occasionally affected by the noise in the 3D data from sources that the acceleration sensor cannot sense and remove. The current-frame result of the plane detection is evaluated using sequential past-frame results. The normal vector and the height from the past scans are compared to the current one that has been detected. If the current plane is the same as past ones, it is registered on the map. Once a plane is found, it is tracked as long as it remains in sensing range. As a result, this system reliably recognizes the environment even during the robot's walking.

V. Planning Safe Steps

For navigation, we plan at the level of footsteps, generating safe stepping motions to keep the robot stable and collision free during its walking trajectory to the goal. By describing the kinds of terrain the robot can safely step on, as well as the limits of the walking controller's stepping capabilities, this abstraction of planning for footsteps provides a low-dimensional space which can be quickly searched. We use an A* search to find the optimal footstep path through the environment. This path is then turned into a full-body motion via the robot's walking controller.

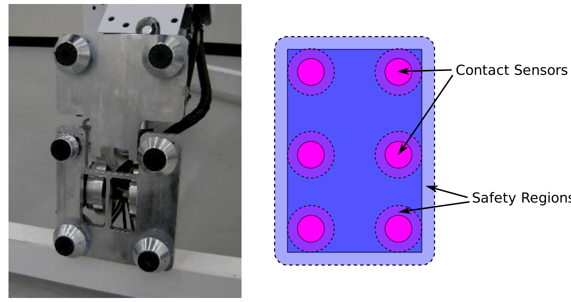

Fig. 4. *Left:* The feet of the robot. There are six contact sensors spaced across the bottom of each foot. *Right:* Step evaluation makes sure the contact sensors and their safety areas are well-supported, and that the rest of the foot and its safety region does not collide with the terrain.

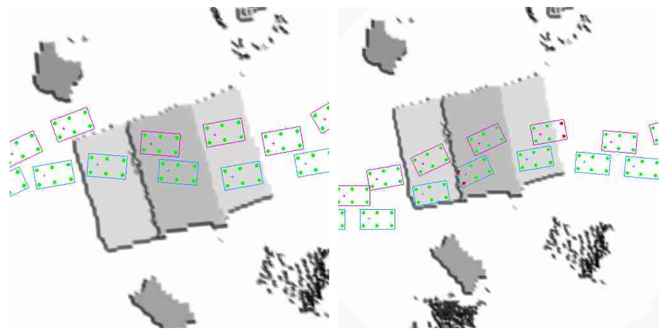

Fig. 5. *Left:* Supporting all six contact sensors. *Right:* Only the front four or rear four contacts need to be supported.

A. *Stable footholds*

Planning for this robot is similar to planning for the humanoid HRP-2[12], but adapted to the needs and abilities of the walking of the robot's feet and walking controller. In particular, the feet of this robot contain 6 contact sensors (shown in Figure 4), which are the only parts of the foot which contact the terrain.

To evaluate a potential stepping location in the environment, we consider two criteria:

1) Are the contact sensors well-supported?
2) Is there any protruding terrain under the foot?

In order to provide solid support for the robot, the contact sensors of the stance foot must be in one of the following configurations: all six sensors in solid contact (best case), or the front four in solid contact, or the rear four in solid contact (see Figure 5). For "solid contact," we want all the cells of the environment map which lie under the contact sensor regions to be in contact with the same plane, and for the angle of that plane not to exceed a preset threshold.

When the contact sensors are satisfied, we then ensure that the terrain does not pass above the contact sensors and

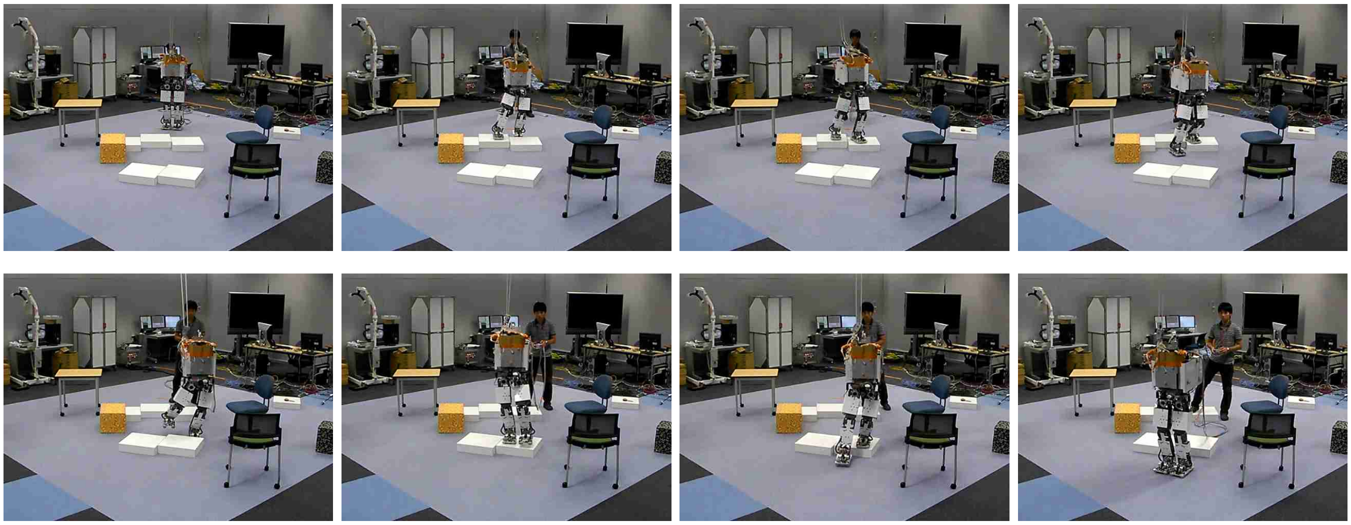

Fig. 6. The robot autonomously navigating an environment with multiple levels and obstacles. The goal position was specified at the start of the trial, and the robot autonomously found its way across the terrain, re-planning at each step.

touch the foot at any other place. For this we verify that all of the cells which lie under the foot have a height which is lower then the foot height at those locations by some safety margin. If a foothold meets both of these conditions, then it is a safe and stable location to which the robot can step.

B. Safe stepping motions

If a safe foothold is found, it still remains to determine a safe stepping motion to reach that foothold. During the planning process, a simple test is performed on the intervening terrain of two adjacent footholds to determine if any part is above the maximum allowable step height. Once a final path is found, a closer inspection of the intervening terrain is performed to generate a spline for the swing foot to follow which safely avoids collisions with the terrain or any obstacles. This spline is generated by choosing points around the convex hull of the terrain between adjacent steps.

For footstep locations or motions which are unsafe, a local search is performed during the planning process to adjust them and find a nearby safe location within the reachable region of the robot. This local adjustment is accomplished by the "informed local search" described in our previous work[12].

VI. Robot Experiments

The system was tested in a 4m×4m carpeted area filled with obstacles such as chairs, tables, boxes, platforms, and stairs. The terrain and obstacles did not have any special texture or instrumentation, and no knowledge of the type or number of obstacles was given to the robot prior to each trial.

With each step the robot took, it would generate a new, updated map of the environment, and re-plan its path. The planning time was restricted to fall within one step cycle, and if planning had not completed in that time, the best partial path was chosen to begin executing. For stability, we restricted the planner to only allow steps in which all six foot contact sensors were well-supported. In challenging environments, the planner would find partial paths of around 5 to 15 steps in the limited planning time. In simpler or clear environments, the planner would find paths all the way to the goal (generally within around 20 steps) within the allotted planning time.

In addition to autonomy through planning to a goal, we wanted ways to interact with and guide the robot while it was in motion. Toward this end, we implemented three different interfaces. (goal setting, path drawing, joystick).

A. Specifying an end goal

The first interface to directing the robot that we tested involved setting the desired goal position of the robot and letting the sensors, planner, and walking controller handle the rest. The robot would begin by scanning the environment and finding an initial path. As it walked, additional data about the environment would be gathered and the path would update and adjust to the new information. In this way, the robot was able to safely walk through a variety of environment configurations. During this planning a simple planner which searched out from the goal was used to provide a heuristic for the footstep planning process, as described in previous work[15].

This mode of operation requires little in the way of operator knowledge, and is the simplest for the human operator to specify. Figure 6 shows one such trial, where the user specified a goal past the two blocks nearest the camera. From that point on the robot's motion was generated autonomously with no further need for human input.

B. Drawing paths

A second interface used a "guide path," drawn by the operator on the environment, as a rough path for the robot to follow. The main idea of this interface is to enable an operator to specify, in a simple manner, commands such as "Walk around this way," or "Step over that," or "Go through

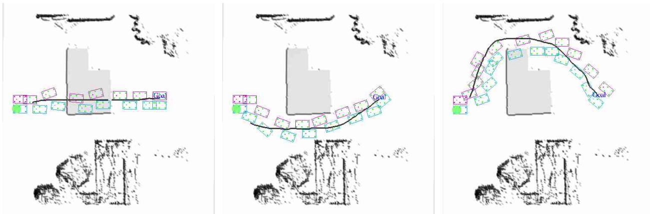

Fig. 7. Plans generated from different guide paths.

there." These commands all specify a little bit of information as to *how* to navigate to a goal, but still leave the details and specifics of the locomotion to the robot.

Figure 7 shows how drawing different guide paths in the same environment allows the user to choose amongst different possible footstep paths in a simple manner. In many environments, there are several paths which do not differ much in overall cost, and this interface allows the operator to easily specify how the planner should go about solving the problem.

The guide path is incorporated into the planning process as a heuristic for the search, replacing the reverse-planning heuristic. This guides the exploration of the space along the path specified by the operator, without forcing it to follow the guide path exactly. In the rightmost example of Figure 7, the guide path is drawn too high in the environment, where it comes close to some tall obstacles. The resulting footstep path stays lower in the environment in that area, deviating from the guide to keep the robot safe. For more details on the influence of the guide path on the planning of footsteps, please see our related work[16].

This interface allows the user to be more specific about how the robot approaches the problem, but requires a little more operator knowledge about the capabilities of the robot to use effectively. If the user draws good guide paths, planning occurs more quickly than with the reverse-planning heuristic, and partial paths are almost always of high quality. However, if the user draws guide paths through parts of the environment that are untraversable, the planning search will be guided in a poor manner. And due to the limited planning time, the planner may not have time to correct for the poor guide path and find a path which deviates sufficiently from the guide path to be executable.

C. Joystick control

A final control interface we used is the more direct method of specifying robot motion using joystick input. This control strategy is based on work previously applied to HRP-2[17]. In this mode of control the robot is not trying to plan a long range path to a goal, but instead trying to fit the user's directional commands from the joystick into the capabilities of the robot and its controller. This allows the user to use a simple interface to directly control the path the robot takes at each moment, without the need to also direct all the details of locomotion, such as foot placement among obstacles.

The joystick command entails 3 axes, which are mapped to forward velocity, lateral velocity, and rotational velocity. From these velocities and the robot's step cycle, we can compute a desired next step to best fulfill the user's command.

The algorithm we used to find the best step is different from the one used in the "Intelligent Joystick" work[17], but the results are very similar. In the previous work, all possible steps were enumerated, and search orders were precomputed for a discretization of joystick inputs. The algorithm we used worked in two parts: First, because we have an explicit representation of the limits of the robot's capabilities for adjusting actions during regular footstep planning, we use that representation to find the closest action to the user command that fits within the reachable region of the robot. We then use that closest action as a reference action and adapt it to the terrain, using the "blind local search" adaptation method described in our previous work[12].

Figure 8 shows one trial using this joystick control in which the operator commanded the robot to walk over a set of blocks, turn around, and walk over them in the reverse direction (only the first half is shown in the pictures). When walking over the blocks, the user only needs to push forward on the joystick, and the underlying system takes care of foot placement, swing-leg trajectories, and overall body motion to move the robot safely over the terrain.

This mode of operation requires the most operator knowledge of the capabilities of the robot. By directly commanding the walking direction, the system has little freedom to deviate from the user's commands. Thus, in order for the robot to make progress, the user must direct the robot along routes in which the robot can find safe footholds. However, in the case that the user commands the robot to move in a direction in which it cannot find safe footholds, it will only walk as far as is safe in that direction. So a lack of operator knowledge does not result in unsafe operation.

VII. Conclusion and Future Work

We have presented a system for autonomous navigation for a biped robot in previously unknown, complicated environments via a pivoting laser scanner and footstep planning. This method uses only on-board sensing, and does not require particular textures, shapes, or advance knowledge or the obstacles and terrain to traverse. It provides dense, filtered 3D data of the areas in front of the robot, and from that data a global path to the goal. Both sensing an planning operate constantly as the robot walks, allowing it to adjust and adapt to new data during the execution of its walking motion.

We also briefly presented three interfaces which can be used to operate the robot and direct it to the desired position in the desired manner, using either a final goal position, a rough guiding path, or joystick control.

There are several improvements that can be made and directions for further research. First, during the trials presented here, the robot's only sensor was a pivoting laser scanner. Additional sensors and additional sensing modalities can be combined into a more complete environment representation, and the registration of successive scans can also aid in reducing odometry error and keeping past observations valid for a longer period.

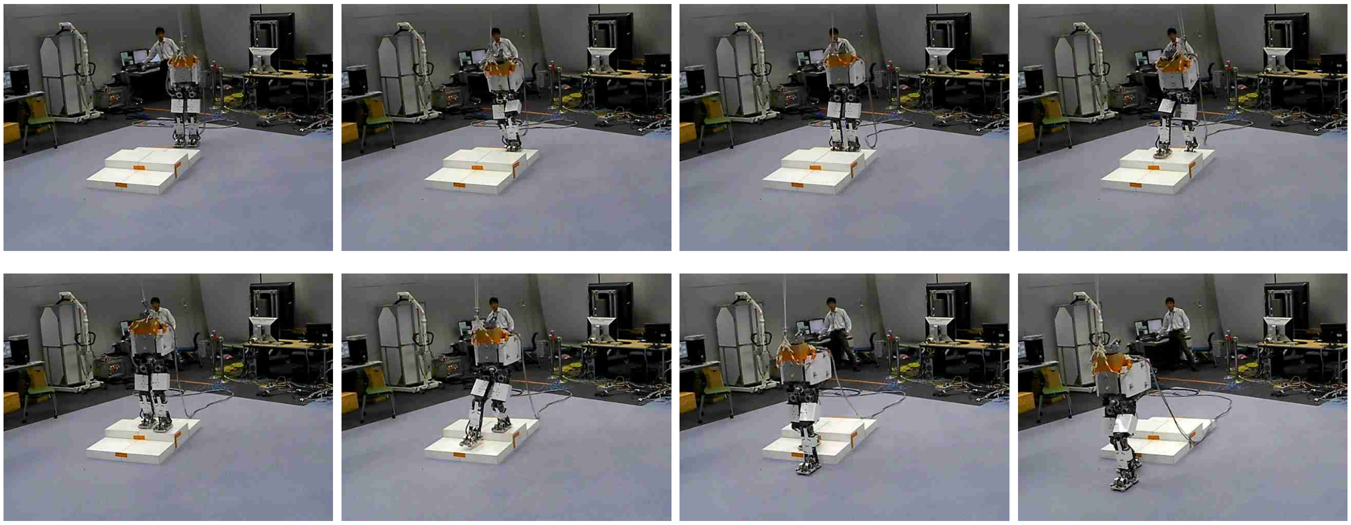

Fig. 8. The robot being controlled by a joystick operator (in the background) and commanded to walk straight forward over the steps.

Next, the method for generating swing leg trajectories for stepping over obstacles or onto different levels is rather simplistic. A better approach would take into account the leg geometry, as well as joint velocity limits, allowing for a safer trajectory which is guaranteed to be executable by the robot.

Also, the environments shown had multiple levels and obstacles of different shapes, but the areas the robot stepped were still flat and horizontal. To attain comparable mobility to a human, biped robots will need to be able to walk stable in even more difficult environments, such as forests, fields, or rocky terrains, where stepping locations are not as smooth or level.

This research is the result of a collaboration between AIST and TOYOTA, utilizing planning systems developed by AIST researchers and sensing techniques developed at TOYOTA.

REFERENCES

[1] K. Nishiwaki, S. Kagami, Y. Kuniyoshi, M. Inaba, and H. Inoue, "Toe joints that enhance bipedal and fullbody motion of humanoid robots," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, DC, USA, May 2002, pp. 3105–3110.

[2] J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba, and H. Inoue, "Footstep planning among obstacles for biped robots," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, pp. 500–505.

[3] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, "The development of Honda humanoid robot," in *Proceedings of the IEEE International Conference on Robotics and Automation*, May 1998, pp. 1321–1326.

[4] P. Michel, J. Chestnutt, J. Kuffner, and T. Kanade, "Vision-guided humanoid footstep planning for dynamic environments," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, December 2005.

[5] J. Chestnutt, P. Michel, J. Kuffner, and T. Kanade, "Locomotion among dynamic obstacles for the Honda ASIMO," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Deigo, CA, October 2007.

[6] O. Lorch, J. Denk, J. F. Seara, M. Buss, F. Freyberger, and G. Schmidt, "ViGWaM - an emulation environment for a vision guided virtual walking machine," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, 2000.

[7] O. Lorch, A. Albert, J. Denk, M. Gerecke, R. Cupec, J. Seara, W. Gerth, and G. Schmidt, "Experiments in vision-guided biped walking," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002.

[8] R. Cupec, O. Lorch, and G. Schmidt, "Vision-guided humanoid walking - concepts and experiments," in *Proc. of the 12th Int. Workshop on Robotics in Alpe-Adria-Danube Region (RAAD'03)*, Cassino, Italy, May 2003.

[9] J.-S. Gutmann, M. Fukuchi, and M. Fujita, "Stair climbing for humanoid robots using stereo vision," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, September 2004, pp. 1407–1413.

[10] ——, "A floor and obstacle height map for 3D navigation of a humanoid robot," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, April 2005.

[11] ——, "Real-time path planning for humanoid robot navigation," in *Proc. of the Int. Joint Conf. on Artificial Intelligence (IJCAI'05)*, Edinburgh, UK, July 2005, pp. 1232–1237.

[12] J. Chestnutt, K. Nishiwaki, J. Kuffner, and S. Kagami, "An adaptive action model for legged navigation planning," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, Pittsburgh, PA, November 2007.

[13] P. Michel, J. Chestnutt, S. Kagami, K. Nishiwaki, J. Kuffner, and T. Kanade, "GPU-accelerated real-time 3D tracking for humanoid locomotion and stair climbing," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2007, pp. 463–469.

[14] Y. Kida, S. Kagami, T. Nakata, M. Kouchi, and H. Mizoguchi, "Human finding and body property estimation by using floor segmentation and 3D labelling," in *Proceedings of 2004 IEEE International Conference on Systems, Man & Cybernetics(SMC04)*, October 2004, pp. 2924–2929.

[15] J. Chestnutt and J. Kuffner, "A tiered planning strategy for biped navigation," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, Santa Monica, California, November 2004.

[16] J. Chestnutt, K. Nishiwaki, J. Kuffner, and S. Kagami, "Interactive control of humanoid navigation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (submitted)*, St. Louis, MO, October 2009.

[17] J. Chestnutt, P. Michel, K. Nishiwaki, J. Kuffner, and S. Kagami, "An intelligent joystick for biped control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Orlando, FL, May 2006.