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

Interactive Control of Humanoid Navigation

Joel Chestnutt[†], Koichi Nishiwaki[†], James Kuffner[†‡], and Satoshi Kagami[†]

[†]Digital Human Research Center
National Institute of Advanced Industrial Science & Technology
2-41-6 Aomi, Koto-ku
Tokyo 135-0064, Japan
{joel.chestnutt, k.nishiwaki, s.kagami}@aist.go.jp

[‡]The Robotics Institute
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213, USA
kuffner@cs.cmu.edu

***Abstract*— We present a method for interactively guiding the navigation of a humanoid robot through complex terrain via an intuitive path-drawing interface. In contrast to full autonomy or direct teleoperation of the robot, the user suggests an overall global navigation route by "drawing" a path onto the environment while the robot is walking. The path is used by a footstep planner that searches online for a sequence of suitable footstep locations that follow the indicated path as closely as possible while respecting the robot dynamics and overall navigation safety. In this way, the planner provides the robot partial autonomy in selecting precise footstep sequences while the human operator retains high-level control of the global navigation route. We present experimental results of the complete system on the biped humanoid HRP-2 navigating on and around various platforms, chairs, and stairs. We use an augmented reality system so that interactively drawing paths on the world is intuitive and natural.**

I. INTRODUCTION

As legged robots gain the abilities to walk and balance on more than just flat, obstacle-free floors, they grow closer to fulfilling the potential of legged locomotion shown by biological systems. To truly fulfill this potential, these robots must successfully traverse complicated, rough terrain, requiring the robots to step onto or over various features of the environment. For humanoid robots to navigate through complex indoor environments designed for humans, the robots must deal with furniture, walls, stairs, doors, and previously unknown obstacles on the floor. For outdoor environments, they must have 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.

We are constantly striving towards full autonomy for our robots, to have them be able to act in intelligent ways and perform tasks with little input necessary from humans. At the same time, however, we want to have easy ways of interacting with our robots, to guide and direct them when needed. This interaction, should not only provide the human a means to more precisely specify desired behavior to the robot, but also take advantage of the human's superior sensing and knowledge in order to achieve higher levels of performance on a given task.

In the case of humanoid navigation, we would like to enable a robot to be guided safely through environments that they cannot sense well, or to areas beyond their current sensing or planning ability, relying on the more global

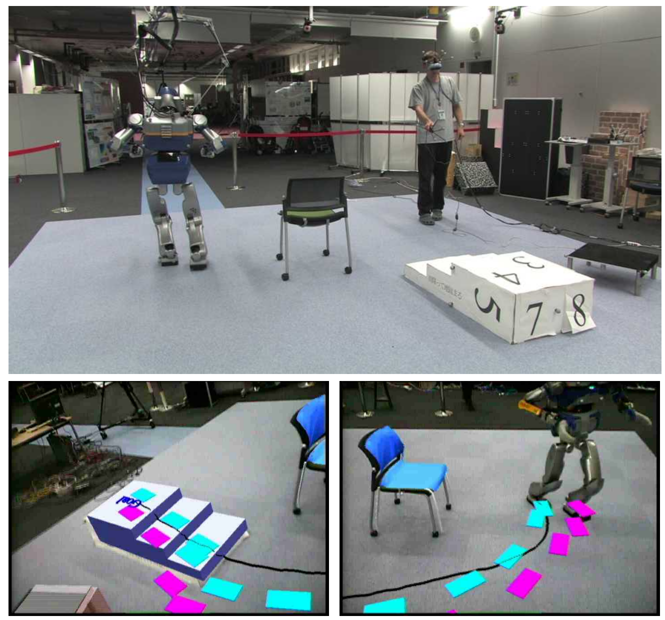

Fig. 1. *Top*: The humanoid HRP-2 being directed by a human operator through an environment with several obstacles. *Bottom*: The augmented reality user view of the scene displaying the guide path and the robot's computed future footstep locations.

domain knowledge of the human operator. Towards this end, we have developed a system for interactively guiding the navigation of a humanoid robot through complicated environments. The user simply draws the overall route that the robot should take using a pointing device. The robot then plans a footstep sequence along the indicated path and automatically generates motion which roughly follows the user's guide while maintaining balance and safety. By integrating this path-drawing with augmented reality systems, it is easy for the user to intuitively specify a desired path directly onto the environment.

Planning footstep sequences rather than whole-body trajectories allows the robot to both reason about the contact with the environment to ensure safe and stable support, as well as reduce the planning state space to a dimensionality that can be reasonably searched online while the robot is walking. With a compact representation of the capabilities of the robot and its walking controller, the planner can quickly find safe paths that take advantage of the robot's ability to step over obstacles or climb up onto stairs and platforms, as shown in Figure 1.

The rest of this paper is organized as follows. Section II describes related work in the field of navigation for legged robots. Section III describes the overall algorithm used to find safe paths through complicated terrains. Section IV explains how we guide the robot by drawing paths into the environment. Section V details the interfaces we use for drawing into the world.

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 navigation strategies is still an ongoing area of research.

Current planning approaches for legged robots can be placed along a spectrum based on how much of the robot's underlying details are considered during the planning process. At one end of the spectrum, every detail is considered, and solving the navigation problem involves solving a giant motion planning problem for all degrees of freedom of the robot. This approach is used for short-term motions, such as whole-body manipulation [1], but can quickly become too computationally expensive for locomotion problems. However, planning the details for the whole body has been used to connect different configurations as part of a locomotion plan [2], [3]. Other systems have used local planning on a step-by-step basis, allowing the robot to adjust its gait locally in response to the sensed terrain, usually in a statically stable manner [4]–[7].

At the opposite end of the spectrum are planners which ignore all the details of the legs, and instead treat the robot as if it were a wheeled robot and "steer" it through the environment. Global navigation strategies for mobile robots can usually be obtained by searching for a collision-free path in a 2D environment. Because of the low-dimensionality of the search space, very efficient and complete (or resolution-complete) algorithms can be employed [8]. Human guidance, via path-drawing and gestures, has been integrated into this kindo of system to provide an intuitive interface for robot navigation [9]. These planar planning techniques have been applied to biped humanoid robots, resulting in conservative global navigation strategies obtained by choosing an appropriate bounding volume (e.g. a cylinder), and designing locomotion gaits for following navigation trajectories computed by a 2D path planner [10], [11]. However, this always forces the robot to circumvent obstacles rather than using the ability to traverse obstacles by stepping over or onto them. For the QRIO robot, this approach has been augmented with additional actions such as stair climbing and descending, allowing the robot to use some more of its capabilities [12], [13]. Other applications of this approach use heuristics to generate a 2D body path for the environment, and then fill in the details along that path with local planning for the legs [14]. Another approach planned ways to adjust HRP-2's body posture to fit into the available free areas along a path [15].

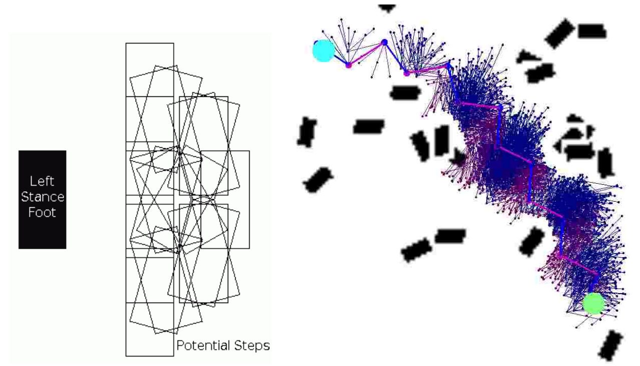

Fig. 2. *Left:* A set of possible footsteps the robot can take with its right foot. *Right:* The planning tree exploring an example environment.

Our approach lies in the middle. We are not planning for all the degrees of freedom of the robot, but we are planning safe stepping motions that take advantage of the robot's legged abilities.

III. Footstep planning

The methods presented in this paper are an extension of our previous research into legged navigation planning [16], and interactive control in complex environments [17]. This section gives a brief high-level overview of the action model and state space that we have used in that work.

Due to the complexity of planning for all the degrees of freedom in a humanoid, we reduce the dimensionality of our search space to keep the problem manageable, while still utilizing the robot's ability to step onto or over obstacles. In our work, this dimensionality reduction is accomplished by reasoning about the foot placement of the robot, discretizing our search along the discrete changes in the hybrid dynamics of the legged systems. This provides a natural way of breaking up the problem, allowing us to apply discrete planners to find a safe sequence of foot placements through the environment. The robot's locomotion controller can then take the robot through each stage of support, taking the robot from its start configuration to the goal.

In this way, our action set becomes the set of possible footsteps the robot can make. One set of possible actions is shown in Figure 2. Collision checking becomes a matter of evaluating footholds and stances at the border between actions, as well as the motion of the connecting footsteps. The planning process with this action model breaks the full motion planning problem into a planning problem in the reduced dimensionality of relevant stances, and then the problem of generating footstep motions, paths through those constraint surfaces.

To re-use existing control strategies with this action set, we can use one of many previously developed locomotion controllers to solve the problem of connecting various stances. For our walking biped example, we can use the stances from the planner as input to a walking controller which then generates a dynamically stable motion taking the robot from one foothold to the next.

A. Goal-directed navigation

Using the footstep planning framework, we can interact with the robot by setting the desired final position of the robot in the world. This is a very high-level approach to navigating the robot, which does not require much user knowledge about the limitations of the robot.

In this approach, the robot manages all the remaining details of planning a path and generating motions to safely reach the goal. We use an A* search through the possible footsteps the robot can make, evaluating them for safety and stability. One example of the search tree grown when planning a path through an environment can be seen in Figure 2. This method has been applied to several robots navigating through complicated environments [16], [18], [19]. In these cases the user would move the goal location during the robot's travel to interactively change the robot's footstep path.

B. Joystick control

To achieve more precise control, we use a low-level interface where the human operator specifies the desired velocities of the robot via a joystick. This form of interaction is useful when the operator needs more direct control over the exact velocities or position of the robot. This is not currently integrated with the goal-directed navigation, but rather is an alternate mode of operation.

Joystick control has previously been used in positioning and navigation for current humanoid robots [20], although this work does not account for obstacles and amounts to "steering" the robot. More direct control of individual legs has been implemented as far back as the GE truck [21], which used force feedback for the operator to individually control the legs. The Adaptive Suspension Vehicle used several different operating modes [22], one allowing the operator to directly control foot placement, a low-speed mode which progressed by "feeling" the terrain, a more autonomous mode where the joysticks control body position and orientation and foot locations are determined more autonomously, and a higher speed walk which did not account for obstacles.

We wish to maintain the simplicity of the joystick interface for specifying velocity, but still have the robot account for obstacles, in a mode of control we refer to as an "intelligent" joystick. The idea of an intelligent joystick can be compared to riding a horse: the rider provides high-level control inputs about which direction to travel, but the horse handles all of the details of locomotion, including the complexities of selecting suitable foot placements and the overstepping of obstacles along the way. In the case of a legged robot, the joystick controls the overall movement direction of the robot, but the system autonomously selects foot placements and trajectories which best conform to the user's command given the constraints of balance and terrain characteristics. For more information about applying this method to a biped robot, see our previous work [17].

This form of interaction is useful when the operator needs more direct control over the exact speed or position of the robot, but it also requires more operator knowledge about the limitations of the robot, so that the user can drive the robot along a direction in which it can find valid steps.

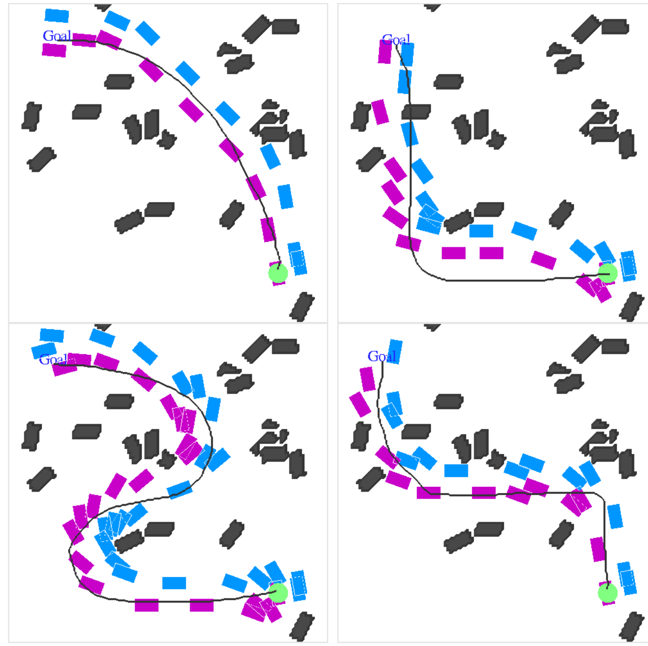

Fig. 3. Specifying different guide paths results in very different paths for the robot to follow.

IV. Directing the Path of a Robot

The main idea of the method proposed in this paper 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 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. This provides a middle ground to the more direct control of using a joystick and the high-level control of only specifying a final position.

Our approach functions by having the operator draw a rough path of where they want the robot to go. Given this rough path, the robot plans out which footsteps it will take in order to safely reach the goal along that path, and generates stepping motions to maintain balance and move to the goal. Because there are multiple paths in use in this system, for clarity we will refer to the rough path that the user draws as a *guide path*, and the path of steps that the robot plans as a *footstep path*. Figure 3 shows the resulting footstep paths that are planned from different guide paths drawn by the user.

Due to the added control being accorded to the operator, extra knowledge of the robot's limitations is required to direct the robot correctly. If the user specifies a guide path that the robot cannot follow, the robot may fail to find any valid path in the time allotted. For example, HRP-2 can only climb the stairs seen in Figures 5 and 7 by approaching them from the front. It cannot step directly up or down 30 cm to reach the top of the stairs from the sides or the back. The operator must be aware of these sorts of limitations in order to draw a path that the robot can reasonably follow.

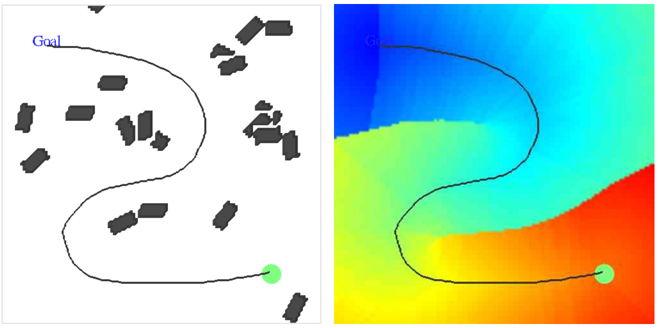

Fig. 4. A guide path through the environment, and the heuristic generated from that guide path.

A. Path-based heuristic

To incorporate the user's guide path into the planning process, we build a heuristic for the A* planning search based on that guide path. This heuristic will guide the planning search along the guide path provided by the user, while still allowing the search the freedom to deviate to find safe stepping locations. This is different from "steering" the robot along the guide path, in that we still plan a set of footsteps out to the goal. If the guide path was sloppy or not able to be followed exactly, the planning process can still find paths close to it using the stepping capabilities of the robot. The heuristic we use is the weighted sum of two distances:

$$h(x) = w_p d_p(x) + w_g d_g(x). \quad (1)$$

The first distance, $d_p(x)$, is the distance from the state x to the nearest point on the user-specified path. The second distance, $d_g(x)$, is the remaining distance along the guide path from the state x to the goal at the end of the guide path. The weights w_p and w_g can be used to fine-tune how much the planning process will deviate from the guide path during its search to the goal, but in the examples shown in this paper, we keep $w_p = w_g$. One example guide path and the resulting heuristic are shown in Figure 4.

In our implementation, the guide path is represented by a linear spline, with vertices sequentially added as the user draws. We compute $d_p(x)$ by using a k-d tree to find the nearest vertex in the spline, and then find the nearest point on either of the linear segments connected to that vertex. It should be noted that this can give an incorrect location for the closest point if the vertices of the spline are sparse, and a segment passes near the state x without a nearby vertex. However, due to many vertices being added during the drawing process, we have a dense set of vertices in our splines. Once the nearest point on the guide path has been found, $d_g(x)$ is easily computed as just the remaining length in the guide path.

B. Planning time

Because the user is providing the planner with extra guidance, the process of generating the footstep path is often faster than when only specifying a goal to the planner. In the simple environment shown in Figure 3, planning to the goal with no guide path takes $0.045s$, of which $0.025s$ is spent building its own heuristic for the terrain. When using a guide path, the planning time will vary depending on the guide path the user draws. For the upper left path in Figure 3, the planner took $0.01s$ to find a footstep path. For the upper right path, the planning process took $0.04s$.

However, for more difficult terrains that involve tight passages in the state space, the gains can be more significant. For the terrain shown initially in Figure 1, planning a footstep path all the way to the goal takes $11.3s$ with no guide path. With a guide path drawn, the planner can find a complete safe footstep path in only $0.71s$.

V. Interfaces

We performed our experiments inside of a motion-capture area. As a result of this setup, we can track the robot and obstacles with a high degree of accuracy in order to test out various algorithms. From the 3D shapes of the objects in our scene and their positions and orientations from the motion-capture system, we build maps of the shape of the environment. Figure 5 shows one such map in the middle of an interaction trial. As the robot moves through the scene, the user draws paths on the scene for the robot to follow. The use of a motion capture setup enabled fairly robust and accurate global tracking of the robot, obstacles, pointing device, and user viewpoint for our experiments. However, note that the general idea and concept of interactive specification of navigation guide paths does not require a motion capture system. For example, if the robot were equipped with accurate human body tracking or other devices to sense the operator gestures and recover the indicated desired navigation route, similar results could be obtained.

A. Augmented reality

In addition to the GUI display shown in Figure 5, we can include this path-drawing interface into an augmented reality system. The augmented reality system operates by localizing cameras within the experimental environment, and overlaying sensor data or planning information onto the camera image to aid in debugging or visualizing the planning process [23], [24].

In addition to displaying the obstacles and path of the robot in the world (as seen in Figure 7), we can outfit a control device with motion capture markers for use as an input device, shown in Figure 6. From the motion capture system, we know its position within the environment very precisely, which allows us to intersect a ray drawn from the frame of the pointer with the geometry of the scene. This intersection point tells us where the user is pointing, acting as a kind of free-floating mouse. With this setup, we can draw guide paths directly into the world, in a very natural manner. Figure 7 shows this system in action. While the robot is walking along a previously specified guide path, the user sketches out a new guide path to follow, ending at the top of the stairs. Once the new guide path is drawn, the robot generates a footstep path to walk along it and climb the stairs.

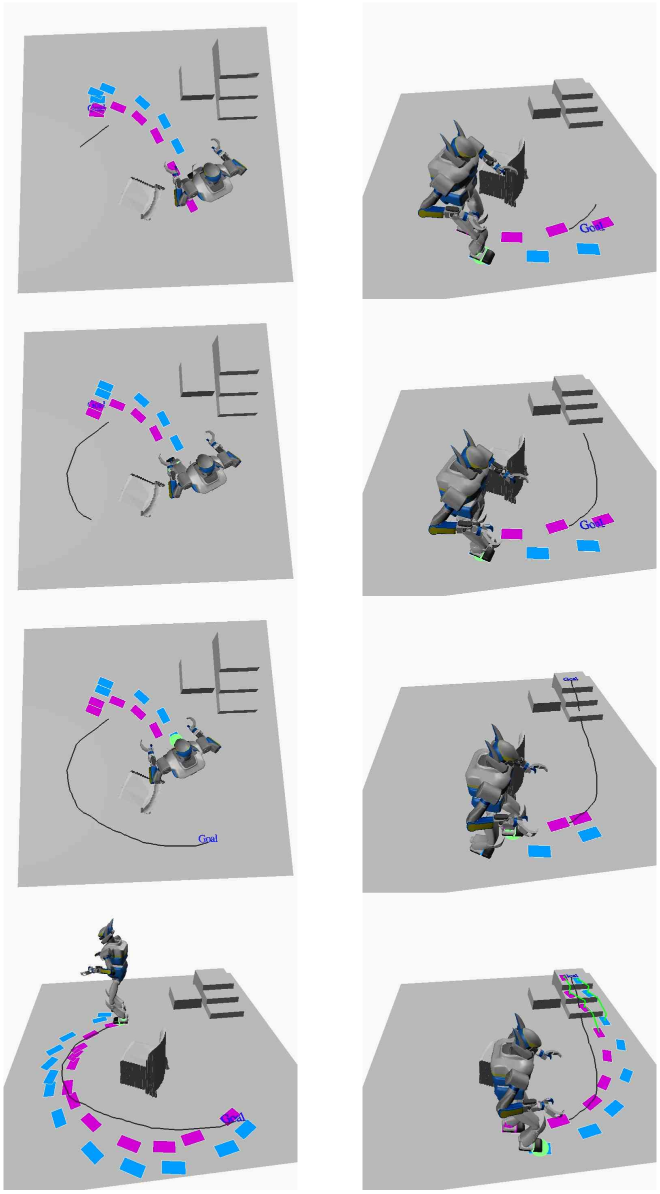

Fig. 5. Drawing paths for the robot while it walks through the environment.

VI. Discussion and Future Work

In this paper we described a method for guiding the planning process of a navigating robot interactively. This guidance enables the operator to direct the robot in a quick and easy manner to safely travel a desired route. In addition, using an augmented reality system to draw guide paths directly onto the world provides a simple and intuitive method for directing the robot. Also, in complicated environments, the guidance provided by the human operator can aid the planner to find safe paths more quickly than it could in a fully autonomous fashion.

While use of the augmented reality system provides an intuitive setup for directing the robot, it is also a complicated, expensive, and non-mobile system. A future refinement to the interfaces discussed in this paper would be to use alternative means of tracking the operator gestures (ideally the robot's own on-board sensors) to detect where a user is pointing, and infer guide paths from the pointing gestures.

Fig. 6. Pointing device which can be tracked via motion-capture, and its appearance in the augmented reality system.

Finally, for a more complete range of control, we are currently exploring methods to provide the operator with a simple way of switching between different modes of command: from specifying goals or tasks, to drawing guide paths, and to more direct joystick control or teleoperation.

References

[1] J. Kuffner, S. Kagami, K. Nishiwaki, M. Inaba, and H. Inoue, "Dynamically-stable motion planning for humanoid robots," *Autonomous Robots*, vol. 12, no. 1, pp. 105–118, January 2002.

[2] K. Hauser, T. Bretl, and J.-C. Latombe, "Non-gaited humanoid locomotion planning," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, Tsukuba, Japan, 2005.

[3] M. Kallmann, R. Bargmann, and M. Mataric', "Planning the sequencing of movement primitives," in *Proceedings of the International Conference on Simulation of Adaptive Behavior*, 2004.

[4] J. E. Bares and D. S. Wettergreen, "Dante II: Technical description, results, and lessons learned," *International Journal of Robotics Research*, vol. 18, no. 7, pp. 621–649, July 1999.

[5] E. Krotkov, J. Bares, T. Kanade, T. Mitchell, R. Simmons, and W. R. L. Whittaker, "Ambler: a six-legged planetary rover," in *Fifth International Conference on Advanced Robotics, 1991, Robots in Unstructured Environments (ICAR '91)*, vol. 1, June 1991, pp. 717 – 722.

[6] R. Hodoshima, T. Doi, Y. Fukuda, S. Hirose, T. Okamoto, and J. Mori, "Development of TITAN XI: a quadruped walking robot to work on slopes - design of system and mechanism," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, 2004.

[7] D. Wettergreen and C. Thorpe, "Developing planning and reactive control for a hexapod robot," in *Proceedings of ICRA '96*, vol. 3, April 1996, pp. 2718 – 2723.

[8] A. Stentz, "Optimal and efficient path planning for partially-known environments," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 1994, pp. 3310–3317.

[9] D. Sakamoto, K. Honda, M. Inami, and T. Igarashi, "Sketch and run: A stroke-based interface for home robots," in *Proceedings of the 27th Annual SIGCHI Conference on Human Factors in Computing Systems*, April 2009.

[10] J. Kuffner, "Goal-directed navigation for animated characters using real-time path planning and control," in *Proc. CAPTECH '98 : Workshop on Modelling and Motion Capture Techniques for Virtual Environments*, 1998, pp. 171–186.

[11] J. Pettre, J.-P. Laumond, and T. Simeon, "A 2-stages locomotion planner for digital actors," in *Proc. SIGGRAPH Symp. on Computer Animation*, 2003.

[12] K. Sabe, M. Fukuchi, J.-S. Gutmann, T. Ohashi, K. Kawamoto, and T. Yoshigahara, "Obstacle avoidance and path planning for humanoid robots using stereo vision," in *Proceedings of the IEEE International Conference on Robotics and Automation*, New Orleans, April 2004.

[13] J.-S. Gutmann, M. Fukuchi, and M. Fujita, "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.

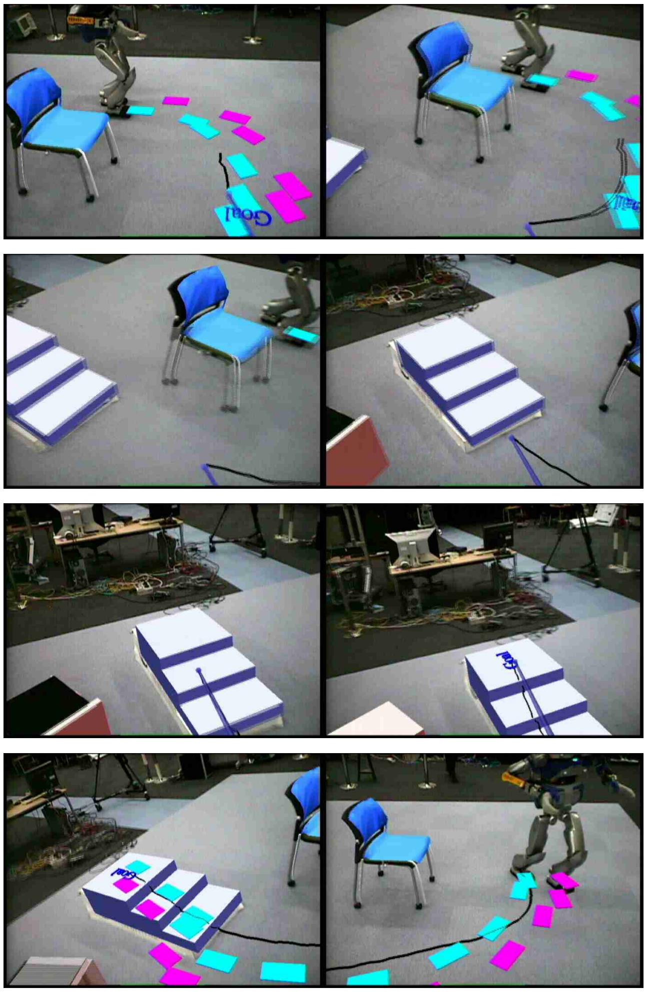

Fig. 7. Drawing a path in the augmented reality environment while the robot walks. The robot's current path is displayed as rectangles.

[14] T.-Y. Li, P.-F. Chen, and P.-Z. Huang, "Motion planning for humanoid walking in a layered environment," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2003.

[15] F. Kanehiro, T. Yoshimi, S. Kajita, M. Morisawa, K. Fujiwara, K. Harada, K. Kaneko, H. Hirukawa, and F. Tomita, "Whole body locomotion planning of humanoid robots based on a 3d grid map," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, April 2005.

[16] 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.

[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.

[18] J. Chestnutt, M. Lau, G. Cheng, J. Kuffner, J. Hodgins, and T. Kanade, "Footstep planning for the Honda ASIMO humanoid," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, April 2005.

[19] J. Chestnutt, "Navigation planning for legged robots," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, December 2007.

[20] J. Kuffner, K. Nishiwaki, S. Kagami, Y. Kuniyoshi, M. Inaba, and H. Inoue, "Self-collision detection and prevention for humanoid robots," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, D.C., May 2002.

[21] R. S. Mosher, "Exploring the potential of a quadruped," International Automotive Engineering Congress, January 1969.

[22] K. J. Waldron and R. B. McGhee, "The adaptive suspension vehicle," *IEEE Control Systems Magazine*, vol. 6, pp. 7–12, 1986.

[23] M. Stilman, P. Michel, J. Chestnutt, K. Nishiwaki, S. Kagami, and J. Kuffner, "Augmented reality for robot development and experimentation," Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-05-55, November 2005.

[24] K. Kobayashi, K. Nishiwaki, S. Uchiyama, H. Yamamoto, and S. Kagami, "Viewing and reviewing how humanoids sensed, planned and behaved with mixed reality technology," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, 2007.