Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems
September 28 - October 2, 2004, Sendai, Japan

Stair Climbing for Humanoid Robots Using Stereo Vision

Jens-Steffen Gutmann, Masaki Fukuchi, and Masahiro Fujita
Intelligent Systems Research Laboratory
Sony Corporation
Tokyo, Japan
Email: {steffen,fukuchi,mfujita}@pdp.crl.sony.co.jp

Abstract— **For the fully autonomous navigation in a 3 dimensional world, a humanoid robot must be capable of stepping up and down staircases and other obstacle where a sufficient large flat surface can support the robot's feet. This paper presents methods for the recognition of stairs and a control architecture that enable the humanoid robot QRIO to safely climb up and down in its environment. The approach is based on data captured by a stereo vision system and segmented into planar surfaces. From the segmented planes, stairs that can be climbed by the robot are extracted and fed to a control system which decides the action to be taken next. Experimental results on a staircase as well as climbing up and down a sill are presented.**

I. Introduction

The autonomous navigation of mobile robots is a long and well studied topic with numerous publications on the subject [3]. The majority of approaches restrict the problem to a 2 dimensional world and in many cases a wheeled robot is used as the target platform.

Recently, the development of legged robots such as humanoids robots enable the agent to not only move in a 2 dimensional world, but to also change its elevation, e.g. by changing its posture from sitting or lying to standing, or by stepping up and down staircases or flat obstacles.

In our previous research, we introduced the robot QRIO, a small biped humanoid robot for entertainment applications [4]. QRIO is able to communicate with humans verbally, and through visual input and gestures. The robot is also able to navigate in a 2 dimensional world by recognizing floor and obstacles using stereo vision, and to plan collision-free paths [16].

While there have been impressive demonstrations of biped robots (including QRIO) walking up and down a staircase, the executed motions are in many cases pre-programmed without any visual feedback. For the fully autonomous navigation in the 3 dimensional world, this is not acceptable and a robot should be able to recognize steps of a staircase or other flat obstacles it can step on.

In this paper, we present methods enabling QRIO to recognize flat surfaces with specific shapes, i.e. stairs, that can be exploited by the robot for 3 dimensional navigation. Each stair is extracted from stereo data segmented into planar regions. Furthermore, a parametrized control system is presented, that enables the robot to autonomously climb up and down the recognized stairs without taking any assumptions about the size, height, direction (up/down) or number of stairs.

This paper is the consequent continuation of our previous work presented in [5]. While the latter mainly focused on precise plane segmentation from range data, the contribution of this paper is the recognition of stairs from the segmented data, and the parametrized control system for stair climbing.

The organization of this paper is as follows. The next section gives an overview of QRIO and our architecture for stair recognition and climbing. Section III briefly describes our method of plane segmentation from stereo data. The recognition of stairs is presented in Section IV, followed by the motion control system in Section V. Section VI presents experimental results; and related work is discussed in Section VII. We conclude in Section VIII.

II. System Overview

QRIO (see Fig. 1) is small biped humanoid robot with 38 degrees of freedom. A real-time body-stabilizing system enables the robot to walk on inclined and irregular terrain adaptively, and to re-stabilize immediately even when external forces affect its balance. Step patterns are generated in real-time for realizing various walking behaviors ranging from active and stable biped walking to dance performances [12].

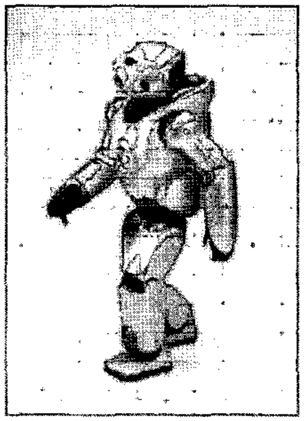

Fig. 1. The small biped entertainment robot QRIO.

0-7803-8463-6/04/$20.00 ©2004 IEEE

For visual perception, QRIO is equipped with a stereo camera enabling the robot to sense and understand the environment in 3 dimensions. The stereo system consists of 2 cameras as the robot's eyes, each with a field of view of 47° horizontally and 39° vertically, and a FPGA module in the robot's head for stereo processing. Using its stereo cameras, QRIO can compute the distance to objects and segment data into planar regions. We use this system for floor extraction and obstacle detection; and plan collision free paths around the obstacles [16].

In this work, we further utilize this stereo system for the recognition of stairs and develop a control system for climbing up and down stairs. Fig. 2 shows an overview. Stereo data is first segmented into multiple planes from which stairs are extracted. The stair information is then fed to a stair-climbing module that decides the action to be taken next and sends commands to the robot's actuators.

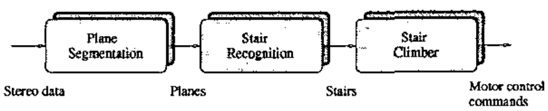

Fig. 2. System for stair recognition and climbing.

For the convenience of explanation, we introduce a robot coordinate system in which data is presented. The origin is at the center between the feet with the x-axis going forward, the y-axis going left, and the z-axis going up (see Fig. 3). When the robot is walking, this coordinate system changes in a discrete way, flipping to the side of the foot that currently touches the ground.

Note that usually the origin is placed in one of the two feet [16] which is a useful representation in a footstep planning system [1], [13]. Thus, our approach is not a footstep planning system, but a single-step planner where the robot can stand on both feet after each step.

Range data captured by the stereo cameras can be transformed into this coordinate system by following the kinematic chain from the camera to the foot and adding an offset for centering. Due to noise in the encoders of the joints or uneven floor, this transformation usually introduces an error of up to 2-5° in rotation[1] and 1-2 cm in translation.

III. Plane Extraction

For the safe walking of QRIO, the robot should prefer to step on flat and even surfaces while tolerating disturbances up to a certain level. By looking for horizontal planar regions, we can effectively direct the robot to terrain where it can walk safely.

We take the (optimistic) assumption that everything that exhibits a flat and even surface can be stepped on by the robot given the difference in height to a neighboring flat surface doesn't exceed a certain threshold. This is true for the floor and objects such as stairs and sills. However,

[1]Experiments revealed that the high rotational error is due to a calibration problem in one of the tilt joints connecting body and head.

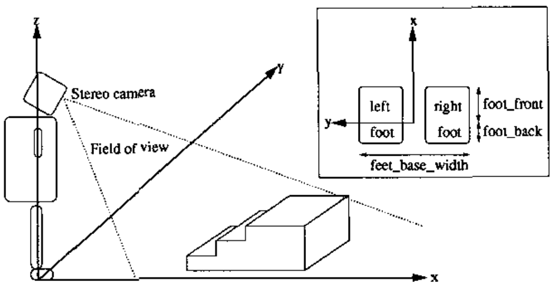

Fig. 3. Coordinate system used in our approach.

it might also trap the robot when placing a flexible flat object such as a paper box where the the robot's weight significantly deforms the surface.

Extracting planar regions from range data is a well-studied problem [9]. In our obstacle avoidance and path-planning system [16] we employed the randomized Hough transformation for fitting planes to the stereo data, an approach that was first presented by Okada *at al.* [15]. While this has been a successful method for finding the ground plane, it turns out that for the precise segmentation of multiple planes (e.g. when observing a staircase), the method is prone to under segmentation, i.e. several steps are merged into one (diagonal) plane.

One way to improve the Hough transformation approach is to only search for planes that are approximately horizontal. However, this is complicated by the fact that the camera-to-robot coordinate transformation is affected by noise, thus a sufficiently large angular interval is needed that might also include the diagonal planes of under-segmented results.

Instead, we employ a variant of the *scan-line grouping* approach by Jiang and Bunke [10], a method that explicitly searches the range data for *break points* where planes intersect. The algorithm first fits line segments to rows (or columns) of the stereo image and then performs region growing using the line segments as primitives. By refining the original line segmentation and region growing approach using adaptive thresholds, we could further improve this method resulting in a more precise segmentation with less under-segmented results. Details about this approach can be found in our previous work [5].

In the plane-segmented data, we search for a horizontal plane close to the ground plane as predicted by the kinematic transformation (we use a threshold of about 7° for orientation and 2 cm for distance). When found, planes and range data are projected to this ground plane. This greatly eliminates the errors introduced in the kinematic transformation.

The result of plane segmentation is a set of planes each represented by the number and the mean of supporting points, a unit-length normal vector and signed distance to the origin (the plane parameters), and a polygon describing the boundary.

Fig. 4 shows a typical result of plane-segmented data

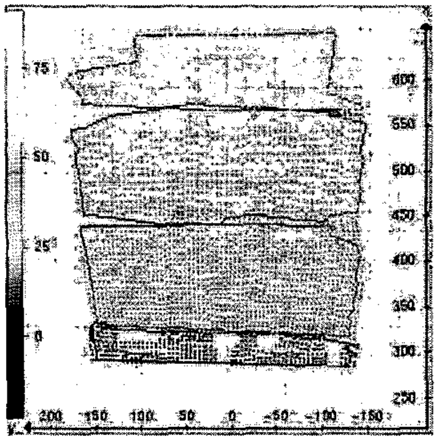

Fig. 4. Example of plane-segmented data extracted from stereo images.

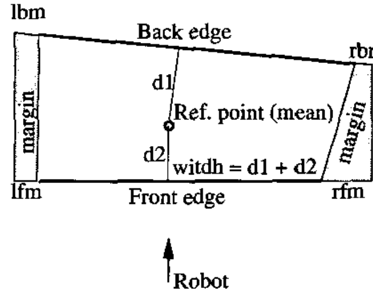

Fig. 5. Representation of a single stair. The robot is assumed to stand approximately in front of the stair.

provided by our multi-plane segmentation method. The intensity of pixels indicate the z-coordinate of points, ranging from dark (0 mm) to light (75 mm). Polygons around points show the segmentation result. Non-horizontal planes are omitted.

Please note that this data shows a rather good example since two steps are fully visible. In many cases the robot can observe only one step or only a part of it due to the limited field of view of our cameras. Thus, further processing is necessary for merging the results of observations over time. This is the task of the stair recognition approach described in the next section.

IV. Stair Recognition

From the plane-segmented range data, the recognition of stairs is straight-forward. We first introduce our representation of stairs and then divide the recognition problem into two phases: stair extraction and stair merging.

A. *Stair Representation*

We represent a stair s by a front and a back edge, a reference point near the center of the step, a step height, and a left and a right margin (see Fig. 5). A staircase $S = \{s_1 \ldots s_n\}$ is then a set of stairs.

Using this data structure we can model rectangular steps and steps of a spiral stair-case. As reference point we choose the mean of supporting points from the plane segmentation. The z-coordinate of the mean also defines the height of the step. The 'width' of the stair at the reference point can be computed as the sum of distances from the reference point to the front and back edges. Front and back edge together with the width define the minimal area the robot can step on. The (optional) left and right margins define extra space for safety without making any particular assumption about their shapes.

B. *Stair Extraction*

Each planar region reported by the multiple-plane extraction method that is approximately horizontal is examined as a stair candidate. The first step is to find a front and a back edge in the polygon describing the region boundary. Usually, however, this boundary contains many *fringes* making it impossible to indicate a front or back edge directly (see Fig.4). Therefore, a smoothing of the boundary polygon becomes necessary.

Due to outliers in the stereo data, the range data is usually not completely dense, which means the polygon reported by plane segmentation does not span the full stair. Therefore, we compute a convex polygon containing the border and use this as a smoothed boundary. Note that this is an optimistic assumption and can cause an overestimation of the size of stairs.

If we assume the robot is approximately standing in front of the stairs, we can identify the front and back edge by drawing a line from the robot position to the reference point. This line intersects the convex polygon at exactly two points which lie on the front and the back edge. Furthermore, a left and right margin can be computed by projecting all vertices onto front and back lines and taking the minimum and maximum coordinates on the lines.

After extracting a stair, we verify that its size is large enough for QRIO to step on. We discard stairs, if the length of one of the two edges is less than a threshold (15 cm), or the width falls below a minimum value (7 cm). This still includes stairs that are smaller than the robot's footprint (16 cm x 10.5 cm).

For the plane segmented data of Fig. 4 our stair recognition approach produces the result shown in Fig. 6. From the four planes, three were classified as stairs. The width of the lowest plane did not exceed the threshold, thus it was not recognized as a valid stair. Note that the topmost stair was only partially recognized since its back end is out of the field of view. In order to obtain a more complete recognition result, we therefore develop a stair merging approach in the next paragraph.

C. *Stair Merging*

In general, a single observation might not give enough information about a stair. Over- and under-segmented results can discard large portions of a step or completely discard it. In particular, when searching for stairs by swinging the robot's head, information from several time frames

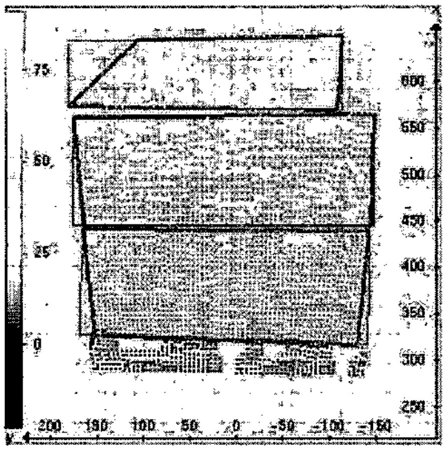

Fig. 6. Result of extracting stairs from the data shown in Fig. 4. Horizontal solid lines refer to the front and back edge of stairs while thin lines indicate the margins.

need to be integrated in order to get a more comprehensive model of the environment.

We maintain a set S of stairs initialized as the empty set. Each time a new set O of stairs is observed, it is merged to S by computing the union of both sets and repeatedly merging and replacing pairs of overlapping stairs until no more pairs can be found. Fig. 7 outlines the algorithm.

Checking for an overlap is performed by verifying that the heights of the stairs are similar and that there is a sufficiently large intersection of the 2D shapes. The merging of two stairs computes the smallest quadrangle enclosing both stairs and adjusts the margins accordingly.

We note that our merging approach is (again) optimistic and can produce over-sized models of stairs. Therefore, it is important in the plane segmentation and stair extraction steps to rather reject dubious results than allowing over-optimistic estimations. Furthermore, the integration of observations is only feasible as long as the coordinate transformation between different time frames are precisely known. We therefore, reset the set S of stairs to the empty set, whenever the robot is performing a step.

Initialize:
    $S \leftarrow \emptyset$
For each set O **of extracted stairs do:**
    $S \leftarrow S \cup O$
    do
        $change \leftarrow false$
        for all $(s, t) \in S \times S,\ s \neq t$ **do**
            if $overlap(s, t)$ **then**
                $S \leftarrow S \cup \{merge(s, t)\} - \{s, t\}$
                $change \leftarrow true$
                exit loop
            endif
        endfor
    while $change$

Fig. 7. Algorithm for merging stairs

V. Robot Motion Control

Stair climbing is implemented as a finite state machine (FSA) composed of the following states:

- Search: look down and swing head for gathering 3D range information.
- Align: Move to a preset position in front of the first stair and rotate in order to align with the stair.
- Approach: walk directly in front of the first stair.
- Climb: step up or down based on the observed data from stair recognition. After stepping, look down for finding new steps. Repeat until no more steps can be found.
- Finish: walk towards a safe area on top (or bottom) of stairs and switch back to normal operation.

Our motion control for stair climbing is parametrized using the variables shown in Fig. 8. We maintain two sets of these parameters, one for stepping up and one for down. Details about the individual variables are described in the different paragraphs below.

Throughout all states of our FSA, the robot examines the height (i.e. z-value) of all steps in the set of stairs. If the height z of one stair falls into the thresholds $step\_min\_z < z < step\_max\_z$ of a parameter set then this stair together with the corresponding parameters are used for further computations. If there are multiple matches, the step closest to the robot is taken.

This approach allows the robot to decide autonomously whether it has to ascend or descend a stair and excludes steps whose height is out of the limitations of the robot.

A. Search and Align

When swinging the head in the search phase, stair information is integrated over time by our stair merging approach described in the previous section. This enables the robot to virtually enlarge its field of view.

The robot then computes a point with distance $align\_distance$ in front of the first step, walks towards it, and upon arrival rotates for aligning with the found step.

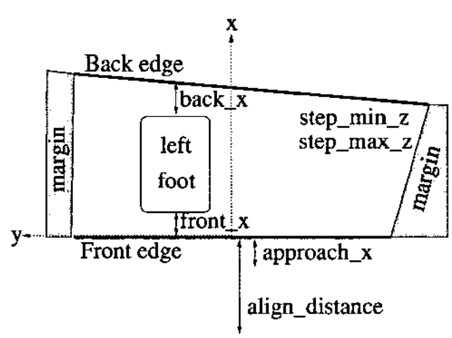

Fig. 8. Parametrization of stair-climbing. See text for explanation.

B. Approaching first Step

Before approaching the first step, the robot changes its posture such that it observes the ground in front of the step. This ensures that the multi-plane extractor can detect the ground plane which eliminates errors introduced by the kinematic transformation (see Sec. III) and provides a better height estimate for the first stair.

The robot then moves into an upright position and walks in front of the first step based on the estimated distance and the *approach_x* parameter. The position of the approached stair is shifted according to the executed motion and the stair is memorized for later evaluation.

C. Climb

The input to the climb state of our FSA is the memorized stair of either the approach state or a previous execution of the climb state. Memorizing the last stair is necessary as the robot is unable observe the immediate next step due to kinematic limitations in the used control mode.

The robot first moves its head down for observing subsequent stairs. Let's denote the memorized step by $step_1$ and the next subsequent one, if found, by $step_2$. By examining the height of $step_2$ relative to $step_1$ we can also choose a corresponding parameter set for $step_2$. This allows, for example, to step on a sill whose width is smaller than the robot's foot sole.

Before climbing $step_1$, the robot verifies that it can actually step onto it. For this the following conditions have to be met (see Fig. 9):

- the front edge of $step_1$ must be close enough and the robot be aligned to it,
- sufficient space on the left and right of $step_1$ exists for stepping up or down (including margins),
- both feet can be placed such that $front\_x$ and $back\_x$ are within given limits of the parameter sets (a preferred value for $back\_x$ should be chosen when possible). For $front\_x$ the parameter set of $step_1$, for $back\_x$ the one of $step_2$ is taken.

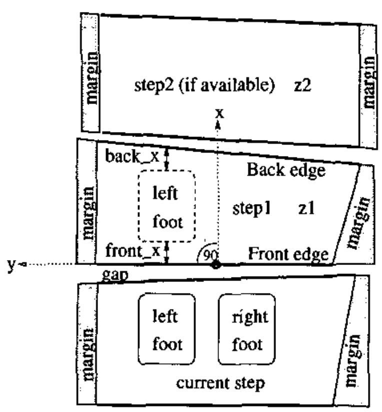

Fig. 9. Climbing up or down a stair.

If all of conditions are met, QRIO steps up or down $step_1$. If $step_2$ was found prior to climbing $step_1$, its pose is projected according to the executed motion and used as the memorized step in the next execution of the climb state. Otherwise, the robot reached the top or bottom of the stairs and proceeds to the finish state.

D. Finish

After successful climbing the stairs, the robot walks towards the center of the current stair. We let the robot celebrate its success by lifting its arms. From here, the robot can proceed using its normal operation.

VI. Results

The system described in this paper has been implemented and verified on QRIO using the parameter sets listed in Table I. All modules are running on the robot's on-board CPU, a MIPS R5000 processor clocked at 400MHz. Stereo data (88x72 pixels) is provided at 12.5 fps. The total processing time for plane segmentation, stair recognition, and motion control is less than 30 ms, enabling the system to perform at full frame rate while still leaving resources to other system components.

Fig. 10 shows several snapshots of QRIO climbing up a staircase of 4 stairs. Each stair if of size 33 cm x 12 cm with a height of 3 cm. The top stair is of size 32 cm x 33 cm. Initially the position of the stairs is unknown to QRIO. The only assumption made is that the robot is approximately in front of it. The robot searches the stairs for about 5 sec, aligns with the first step (5 sec), approaches it (5 sec), and climbs all four steps (24 sec). It then moves to the center of the top stair (5 sec). As can be seen from the figures, QRIO is able to successfully climb up the staircase only by using its stereo cameras for visual input. The geometry of stairs as well as their number are all estimated from the data provided by the multi-plane extractor. In the same way, QRIO can descend the staircase (see Fig. 11).

For verifying that the robot can take advantage of its autonomous recognition of stair transitions, we tested the system on a single stair where the robot has to step up and down the stair. This situation is common in household environments where rooms are often separated by a sill at the door. In order to make QRIO able to walk from one room to another, it has to climb over such a stair.

Fig. 12 shows the situation when QRIO steps onto the sill. The width of the sill is 9 cm, 1.5 cm smaller than

	Ascend	Descend
step_min_z	15 mm	-50 mm
step_max_z	50 mm	-15 mm
align_distance	100 mm	50 mm
approach_x	-15 mm	15 mm
front_x_limit	-25 mm	15 mm
back_x_limit	-5 mm	25 mm
back_x_desired	-15 mm	15 mm

TABLE I
PARAMETERS USED ON QRIO FOR STAIR CLIMBING.

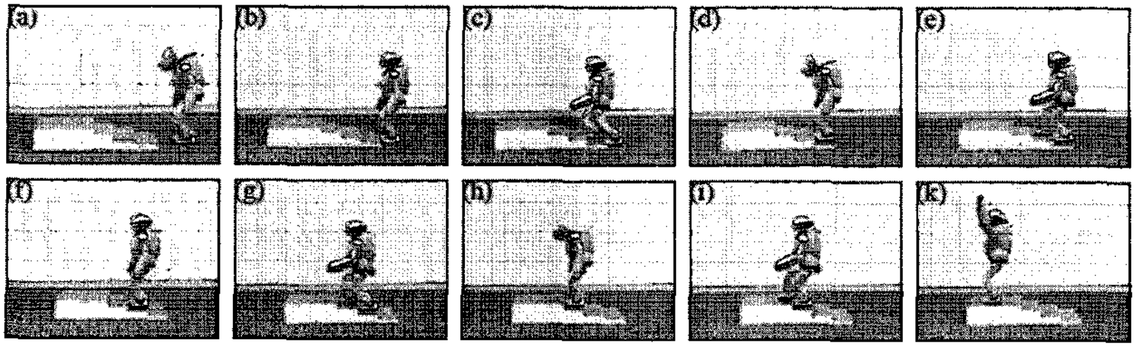

Fig. 10. QRIO climbing up a staircase: (a) search, (b) approach, (c-j) ascend, (k) on top.

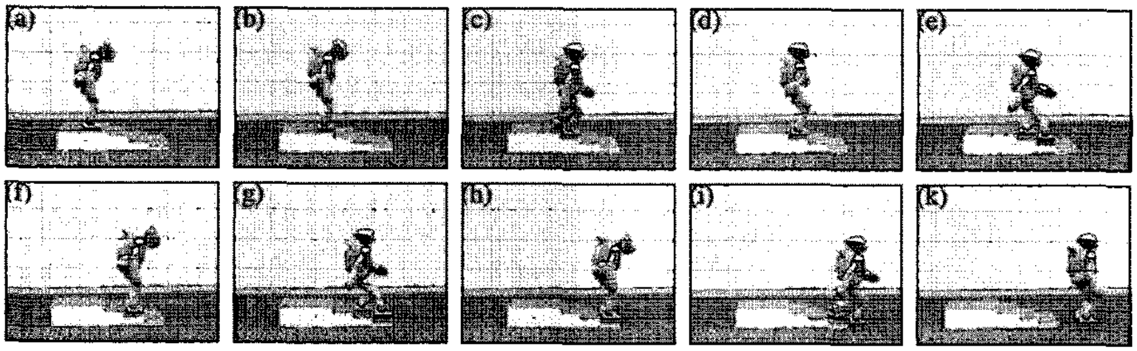

Fig. 11. QRIO climbing down a staircase: (a) search, (b) approach, (c-j) descend, (k) at bottom.

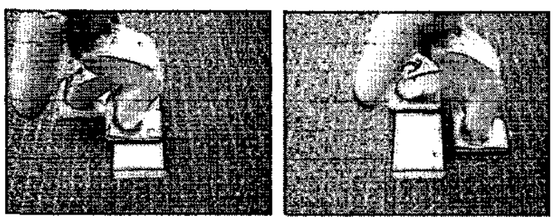

Fig. 12. QRIO stepping up on a sill

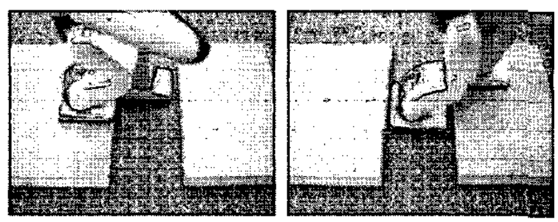

Fig. 13. QRIO stepping down on a basin

the foot size. Using our parametrized climbing approach, QRIO is able to step up and down the stair.

We further verified the situation where QRIO would have to step down on a stair and up again (a quadrangular shaped basin with flat surface). Unless the width of the basin exceeds a certain threshold, the robot refuses to step down on it. Fig. 13 shows a situation with a basin of size 13 cm (2.5 cm larger than the foot size) where QRIO was able to proceed.

VII. Related Work

Autonomous stair climbing has been a subject for many years. Early work concentrated on purely mechanical approaches, e.g. vehicles or wheelchairs that can climb up or down staircases without any input to infer about the size or shape of stairs [19].

For a legged robot, it is possible to develop a special leg design that enables the robot to climb stairs without visual feedback. One such example is the six-legged robot *RHex* as presented by Moore *et al.* [14].

Detecting the horizontal edges of stairs can be realized using monocular vision. This allows a robot driven by caterpillar to align itself with the found edges and move up and down stairs [7], [18].

Recognition of stair-cases by monocular vision has also been investigated by Se and Brady [17]. They use texture detection for finding distant stairs and edge detection when a staircase is closer. Using an *a priori* model they can also estimate the orientation and slope of the stairs.

If a complete *a priori* model of a staircase is given, stairs can be localized by extracting image features and matching them to the model. This approach enables Honda's Asimo to autonomously and continuously walk up and down stairs [6], [8]. The robot has been further equipped with sensors in its feet for detecting the foot position on a stair. This enables the estimation of the robot's stride length which can be used for adjusting future foot positions. [11].

A different model-based approach is taken by Cupec *et al.* [2], where objects with vertical and horizontal edges are extracted from monocular vision. By combining the edge extraction results, box-like obstacles and a staircase are recognized. Only objects near the ground plane are considered, thus it is questionable if a larger staircase could be recognized correctly.

Okada *et al.* were the first who presented stair-climbing for a humanoid robot using plane-segmented data from a stereo vision system. In their work, the robot was able to climb up one step only. For a staircase composed of several steps, it is unclear if their *plane segment finder* is able to precisely segment the planes.

The problem of motion planning for a humanoid robot in 3D is addressed by Chestnutt *et al.* [1]. They also address the case of stair climbing and develop a complete footstep planning system. However, their results are limited to a simulated world.

VIII. Conclusion

In this paper, we presented the autonomous stair climbing of a humanoid robot based on stereo vision. Multiple planes are extracted from range data from which in turn stairs are recognized and integrated over time. Stair climbing is performed by using two parameter sets depending whether the next step indicates an ascending or a descending step. A state machine enables QRIO to autonomously walk up and down a staircase or to walk over a sill. In principle any up and down of stairs can be handled by the approach.

The approach does not take any assumption about the size, height, or number of stairs. The only limitations are that stairs exhibit enough texture to ensure stereo data is dense, and that stairs have a quadrangular surface. To the best of the authors knowledge, this is the first approach that enables a humanoid robot to autonomously climb up and down a multiple of stairs without the need of an *a priori* model of the staircase.

Currently, our system does not take into account foot pad force sensors for verifying the position of feet on the stairs. Adding this information could improve the reliability of your system significantly.

In the future we will enlarge the field of view of our vision system such that the robot can easily observe several stairs, investigate the continuous climbing of stairs, and integrate the stair climbing approach into a full 3D navigation system.

Acknowledgment

The authors would like to thank all members of Sony Networked CE Laboratory, Sony Entertainment Robot Company and Sony Life Dynamics Laboratory for making this work possible.

References

[1] Joel Chestnutt, James Kuffner, Koichi Nishiwaki, and Satoshi Kagami. Planning biped navigation strategies in complex environments. In *Int. Conf. on Humanoid Robotics (Humanoids)*, 2003.

[2] R. Cupec, O. Lorch, and G. Schmidt. Vision-guided humanoid walking - concepts and experiments. In *Int. Workshop on Robotics in Alpe-Adria-Danube Region (RAAD)*, Cassino, Italy, May 2003.

[3] L. Feng, J. Borenstein, and H.R. Everett. "Where am I?" Sensors and methods for autonomous mobile robot positioning, 1996.

[4] M. Fujita, Y. Kuroki, T. Ishida, and T.T. Doi. A small humanoid robot SDR-4X for entertainment applications. In *Int. Conf. on Advanced Intelligent Mechatronics (AIM)*, Kobe, Japan, 2003.

[5] J.-S. Gutmann, K. Kawamoto, K. Sabe, and M. Fukuchi. Multiple plane segmentation with the application of stair climbing for humanoid robots. In *Symposium on Intelligent Autonomous Vehicles (IAV)*, Lisbon, Portugal, 2004. To appear.

[6] Y. Haikawa. Navigation control system for mobile robot. US patent 5,378,969, 1995.

[7] D.M. Helmick, S.I. Roumeliotis, M.C. McHenry, and L. Matthies. Multi-sensor, high speed autonomous stair climbing. In *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2002.

[8] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka. The development of Honda humanoid robot. In *Int. Conf. on Robotics and Automation (ICRA)*, Leuven, Belgium, 1998.

[9] A. Hoover, G. Jean-Baptiste, X.-Y. Jiang, P.J. Flynn, H. Bunke, D.B. Goldgof, K.K. Bowyer, D.W. Eggert, A.W. Fitzgibbon, and R.B. Fisher. An experimental comparison of range image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):673 – 689, 1996.

[10] X.-Y. Jiang and H. Bunke. Fast segmentation of range images into planar regions by scan line grouping. *Machine Vision and Applications*, 7(2):115 – 122, 1994.

[11] T. Kawai, T. Takenaka, T. Hasegawa, and T. Matsumoto. Apparatus for recognizing the landed state of foot of legged moving robot. US patent 6,177,776, 2001.

[12] Y. Kuroki, M. Fujita, T. Ishida, K. Nagasaka, and J. Yamaguchi. A small biped entertainment robot exploring attractive applications. In *Int. Conf. on Robotics and Automation (ICRA)*, 2003.

[13] O. Lorch, A. Albert, J. Denk, M. Gerecke, R. Cupec, J. F. Seara, W. Gerth, and G. Schmidt. Experiments in vision-guided biped walking. In *Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2484–2490, Lausanne, Switzerland, October 2002.

[14] E.Z. More, D. Campbell, F. Grimminger, and M. Buehler. Reliable stair climbing in the simple hexapod 'RHex'. In *Int. Conf. on Robotics and Automation (ICRA)*, 2002.

[15] K. Okada, S. Kagami, M. Inaba, and H. Inoue. Plane segment finder: Algorithm, implementation, and applications. In *Int. Conf. on Robotics and Automation (ICRA)*, Seoul, Korea, May 2001.

[16] 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 *Int. Conf. on Robotics and Automation (ICRA)*, 2004. To appear.

[17] S. Se and M. Brady. Vision-based detection of staircases. In *Asian Conference on Computer Vision (ACCV)*, 2000.

[18] S. Steplight, G. Egnal, S.H. Jung, D.W. Walker, C.J. Taylor, and J.P. Ostrowski. A mode-based sensor fusion approach to robotic stair-climbing. In *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2000.

[19] C.J. Watkins and S.R. Watkins. Stair vehicle. US patent 4,421,189, 1983.