

Development and Evaluation of OmniWalker for Navigating Immersive Computer Based Mine Simulations

Minghadi Suryajaya^{*}; Dr. Chris Fowler^{*}, Senior Research Fellow;
Dr. Tim Lambert[~], Senior Lecturer; Dr. Phillip Stothard^{*}, Senior Research Fellow;
Assoc. Prof. David Laurence^{*}; Dr. Chris Daly, Senior Lecturer

^{*}*School of Mining Engineering, UNSW, Sydney, 2052, Australia*

[~]*School of Computer Science and Engineering, UNSW, Sydney, 2052, Australia*

minghadi@student.unsw.edu.au, c.fowler@unsw.edu.au,

pmstothard@unsw.edu.au, lambert@cse.unsw.edu.au, d.laurence@unsw.edu.au, c.daly@unsw.edu.au

Abstract. The University of New South Wales, School of Mining Engineering is performing research using experimental OmniWalker platform for Navigating Immersive Computer Based Mine Simulations. This driver for this project is that many computer based simulations rely on conventional navigation methods such as a joystick and keyboard to enable the user to navigate around the virtual environment. In most cases, this constrains the user and prevents them from actually walking around in the synthesized environment as they would in a real environment. In some instances of safety oriented simulations this may create a false impression of the difficulty of the tasks to be undertaken and the complexity of the environment. The research has found that many state-of-the-art locomotion systems such as omnidirectional treadmills have a huge barrier to entry due to the high cost of ownership. In some cases, safety may also be an issue as the slow response of such mechanical devices renders them unable to adjust to with sudden changes in walking speed. In this paper, the University of New South Wales, School of Mining Engineering presents a preliminary study of our recently-developed OmniWalker in terms of its effectiveness in delivering improved VR simulation. The device itself is relatively inexpensive and very robust for regular usage. The paper presents details of system development and of a preliminary study on the effectiveness of the OmniWalker in underground coal safety training.

1. INTRODUCTION

Immersiveness is one of the key factors that influence the quality of the virtual reality experience. Some virtual reality training simulators, such as a driving simulator, can be built in a cabin-like form to immerse a user in the virtual environment. However, there are other varieties of simulation in which such an arrangement is not suitable; for example, Unaided Self Escape from an underground coalmine, where the user, in the real scenario would not be constrained within a cabin like environment. It has been demonstrated that real walking, rather than ‘fly-at-ease’ (joystick controlled motion), significantly improves the subject’s learning rate and understanding of the virtual environment[1, 2].

A device that allows user to perform actual walking is called a locomotion system. Most of these devices are very expensive to construct and maintain. One of the most well known of these is the Omnidirectional treadmill [3, 4]. They consists of two perpendicular treadmills woven together in a mechanical fabric. There are also other systems [5-10] where motorized platforms are used to compensate for the subject’s foot movement and hence keep the subject stationary.

Often, these devices are very speed-limited for safety reasons. Sudden acceleration and deceleration while walking is something we commonly do. However, active motors are inherently incapable of rapidly responding to unpredictable changes of acceleration.

Our system has adopted the basic design of the OBDP locomotion platform[11]. It has demonstrated the advantages of dish-shaped stroller platform which conforms to the kinesiology of the human body. However, the original OBDP system is very expensive to construct and maintain, since it uses about nine hundred custom-made ball bearing sensors to detect the user’s feet. Also, these ball bearing sensors are quite fragile in comparison to the significant forces exerted by the foot during walking.

Instead of custom-made stroller, our system uses standard ball transfer units commonly used on the factory floor to move heavy goods. The user motion is tracked using computer vision using off-the-shelf webcams, resulting in a robust and relatively inexpensive locomotion system.

This paper begins by describing the construction of our hardware platform and the development of the computer vision tracking mechanism and virtual mine environment where the user can undertake virtual exploration.

2. SYSTEM OVERVIEW

2.1 The OmniWalker Stroll-based platform

The platform is constructed in the shape of dish of diameter 1.5m. However, our earlier testing indicated that height-proportional hemispherical platform (that is where the radius is the distance between waist and foot) does not provide sufficient inclination for the feet to

slide naturally. A flat inclination angle was shown to be more suitable since it better accommodates the sliding action of the feet. Consequently, we use a flat 15 degree inclination angle. We also placed a rubber support under the dish to provide damp.

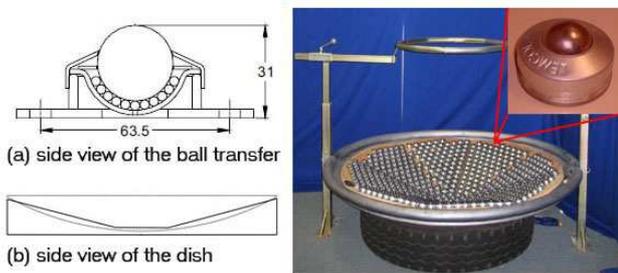


Figure 1: Walking platform

We use rugged heavy duty ball transfer units which can support a maximum capacity of 45 kg in a 36cm² space. A foot size of 20cmx8cm is therefore supported by a minimum of four of these ball-bearings. Consequently, the system can support a person with a maximum weight of 180kg standing on one foot. We have also added a circular support frame at waist-level height.

2.2 Tracking and visualisation hardware system

Tracking is a combination of image acquisition and processing. We used three PS3 EyeToy cameras, each of which is capable of acquiring very high-speed images of up to 120 frames per second. Image processing is done by mid-end PC in order to keep the cost to minimum. In the next sections, we describe the system architecture in more details.

3. SYSTEM DESIGN AND ARCHITECTURE

The ultimate aim of the system is for multiple users to be able to interact with each other in real-time, enabling the system to be used in remote delivery of virtual learning.

As shown in Figure 2, we developed our system into two separate modules; tracking and visualisation modules. Tracking module performs image acquisitions and computer vision computation. Visualisation modules rendered the images from either first, second or third person view. Second person view allows other user fitted inside (another) Omni Walker platform to interact with the others. Third person view allows external audiences to remotely observe the activity of the user(s).

VRPN library[12] is used as communication plug-ins between the modules. It allows us to develop the modules in a client-server model, where the client(s) can easily subscribe for the generated tracking data. Consequently, this allows other visualisation modules to have a third person view of the subject, enabling real-time interactions between multiple users and instructors in the virtual world.

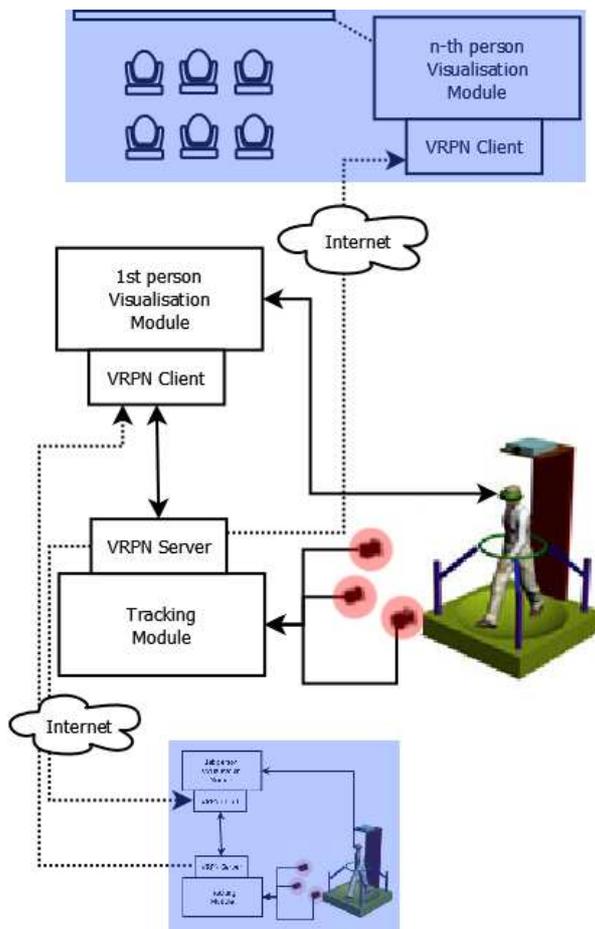


Figure 2: System Architecture

4. TRACKING MODULE

4.1 Camera calibration

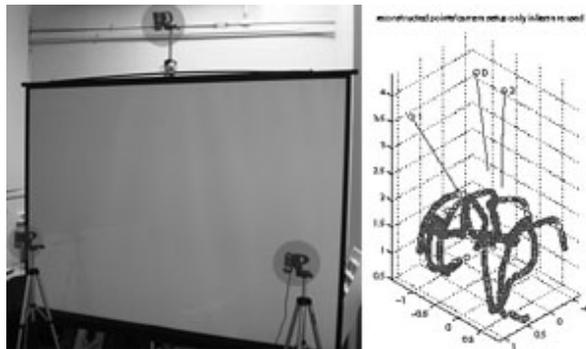


Figure 3: Camera positions and their calibration results from the Svoboda's self calibration toolbox

The cameras are placed in triangular formation to be able to see all the markers with minimum occlusion. Camera calibration is done using the multicamera self-calibration toolkit developed by Svoboda et al[13]. We use a small LED torch as the calibration wand. Scene calibration is done using three right-angled LED lights attached to the frame of the platform. It yields a fixed equation of the ground plane which is used later to reproject the triangulated 3D markers' position. Our previous system[14] used Bouguet's algorithm [15] which had a higher error and frequently had to be

repeated in order to obtain a more accurate calibration matrix.

4.2 Body markers and 2D Tracking

We use ten LED orbs (refer to Figure 4) as markers on the user's body joints. They are attached using velcro and flexible bands to make them comfortable to wear. Each LED orb consists of a diffuser and red, green, and blue LEDs. Hence, it can produce a range of colors. We used five different colors for upper and lower body parts. The choice of color for any particular joint is not important as long as they are different. This color scheme greatly reduces the complexity of the matching and searching procedure which will be described later. Our earlier system [14] used reflective markers that required a bright lighting environment which reduced immersiveness by "washing out" the display screen.



Figure 4: LED orbs and calibration wand

To initiate tracking, the subject adopts a T-stand stance. In order to improve performance, heuristic matching is only done on one camera view. Once a match is found, the intensity threshold is defined for each view and an array of histograms is regenerated for each marker in each view.

The 2D position of the markers is then tracked using a mean-shift algorithm[16]. The basic idea of the mean-shift algorithm is to search for the highest probable location of the markers in the neighbourhood of the previously found location. We use a combination of intensity thresholding and histogram backprojection[17] as the input to the mean-shift algorithm.

Intensity thresholding is used to obtain the most likely 2D position of the markers regardless of its color. It applies fixed level thresholding to the maximum value of the three-channel RGB image array. We applied intensity thresholding independently on each camera view.

Histogram backprojection generates a grayscale image based on color similarity to the tracked color. Consequently, the combination of these two methods yields a grayscale image, highlighting the most probable location of each tracker from each camera view (Figure 5).

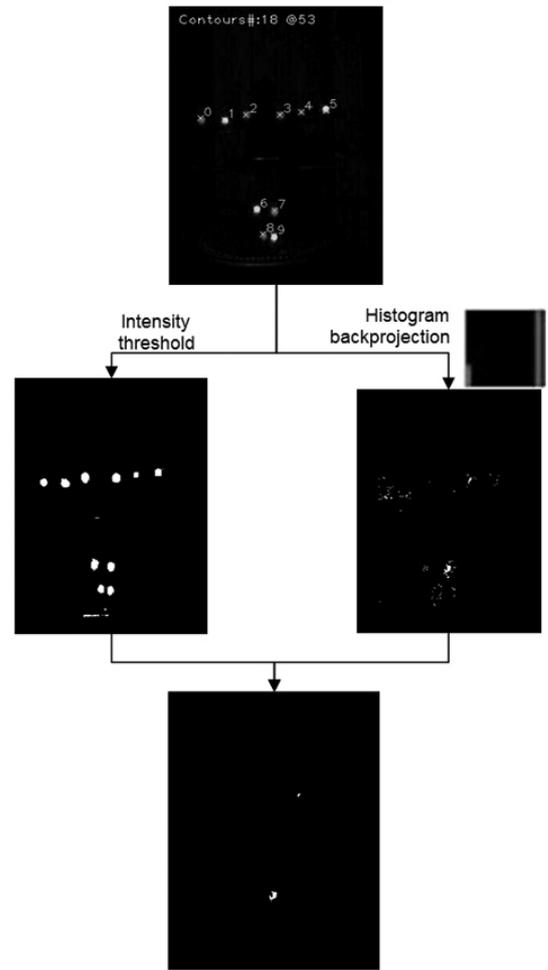


Figure 5: Mean-shift input from intensity thresholding and histogram backprojection for one of the markers

4.3 3D Triangulation and Gait recognition

We used Hartley and Zisserman's [18] method of Direct Linear Triangulation to get the 3D position of the markers. DLT allows us to obtain the 3D position of the markers from two 2D positions of the markers using the pre-calibrated camera matrix. The selection as to which pair of 2D features points to use is done sequentially. The system will always select from the first two views (left and top cameras). If any of the first two views has lost track of the markers, we then use the 2D feature points from the third view. We then use a Kalman filter[19] to reduce the noise in the reconstructed 3D data.

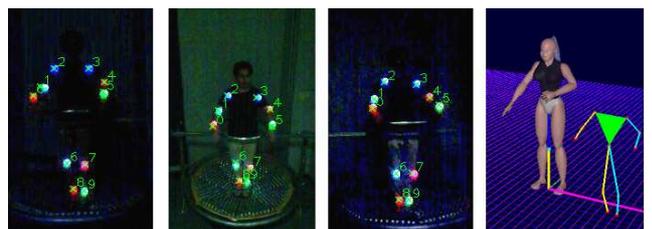


Figure 6: Three views capture (left, top, and right cameras) and the generated marker 3D position

The ultimate aim of the system is for multiple users to be able to interact with each other in real-time. This requires each user being able to control the avatar's body parts. We divide the control mechanism into two sections, upper and lower body model. Upper body model is controlled in similar fashion to 3 degree-of-freedom motion capture, where the avatar's limbs are translated and rotated according to the relative position of the tracked markers. However, the lower body parts of the avatar cannot be controlled in the same fashion since the movement of the legs (sliding down an inclined plane) is not the same as in a normal walk. In order to synthesize a robust and natural-looking walk, we use a similar mechanism to that of [20]. Like Raibert and Hodgins, we divided the walking animation into several stages and interpolated them.

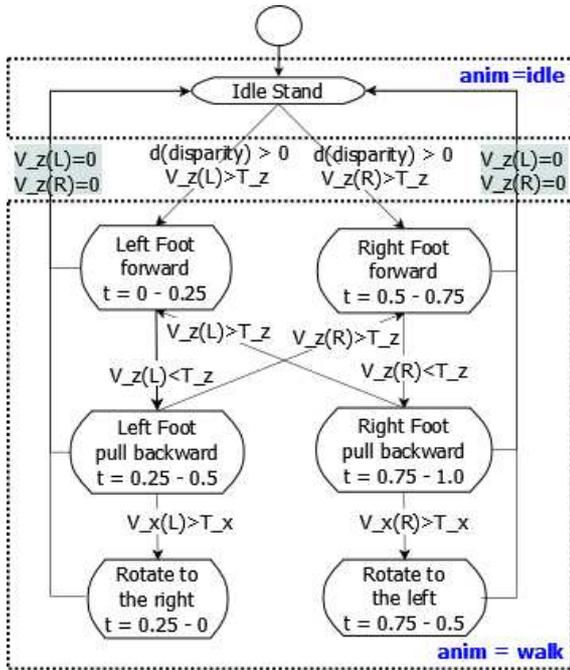


Figure 7: Walking animation states

Referring to Figure 7, the walking states diagram, $V_z(L)$ refers to the velocity of the left foot in the z-direction (forward and backward) and similarly for $V_z(R)$, $V_x(L)$, and $V_x(R)$. T_z and T_x are thresholds which are determined heuristically depending on the tracking accuracy which we currently put as 0.02. The walking animation time (t_c) is interpolated to be between 0 to 1.0. The current state of the walking animation is determined by the foot location relative to its maximum displacement. This yields the current animation time t_c . Immediately changing the animation time to t_c would result in jerky movement, so instead, the animation is allowed to run in accordance with the rendering update time until it reaches t_c , though this target will change as the foot moves. This introduces a slight lag, but users find this acceptable.

To change the walking direction, the user needs to adopt a sidesliding gait that is similar to the natural way of turning while walking. For example, for turning left the walker would swing forward the right foot then slide it to the right and finally slide it back again.

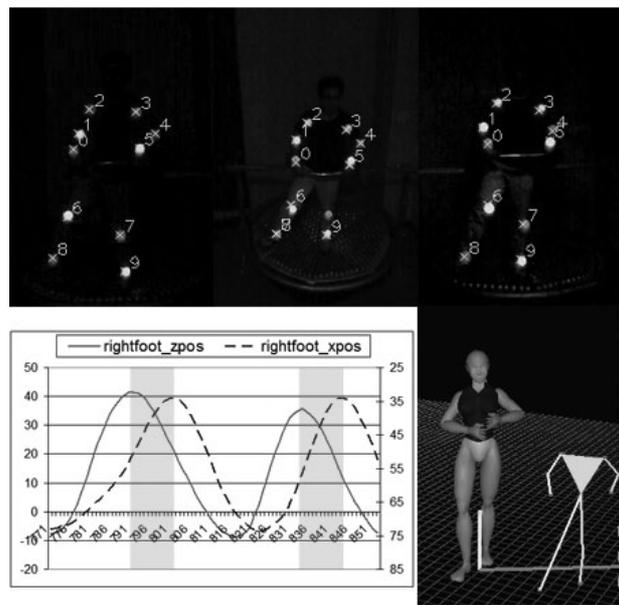
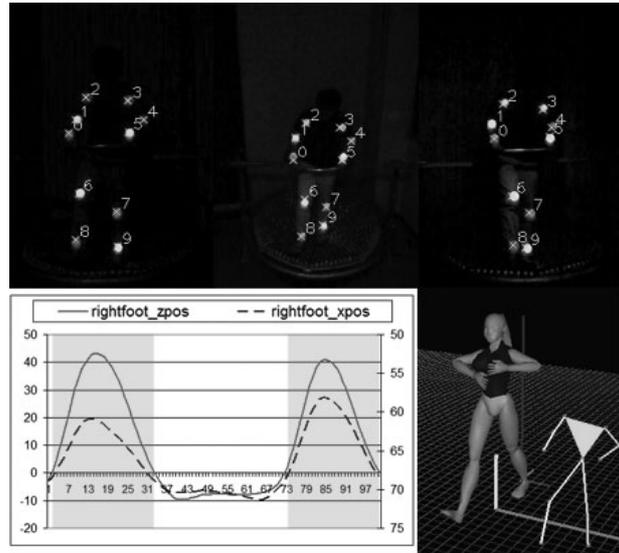


Figure 8: X-axis and Y-axis position of right foot during forward walking gait (top) and during left rotation gait (bottom). The X-axis is the image acquisition index (about 2ms in-between)

The system will track the sliding motion and rotate the avatar accordingly. Once a change of direction of gait is recognised, instead of continuing with the animation, we run the animation in reverse. Since when we are changing the direction of walk, the back foot usually acts as the pivot point to which the front foot will be pulled back. As shown in Figure 8, during rotation, the x-axis and z-axis movement of the foot are out-of-phase. When the z-position of the right foot is changing direction (from increasing to decreasing), the x-position does not follow the same trend as in the forward walking gait.

5. VISUALISATION MODULE

We model the coal mine worker avatar using the Blender software suite, which is an open-source 3D modelling package with a rich feature set. It has robust

supports for multiple file formats, physics engine, and plugins for exporting the model to other rendering engines. The virtual mine environment is developed using the OGRE 3D engine. The open source nature of these software suites allows us to be more flexible and produced significant saving in development cost.

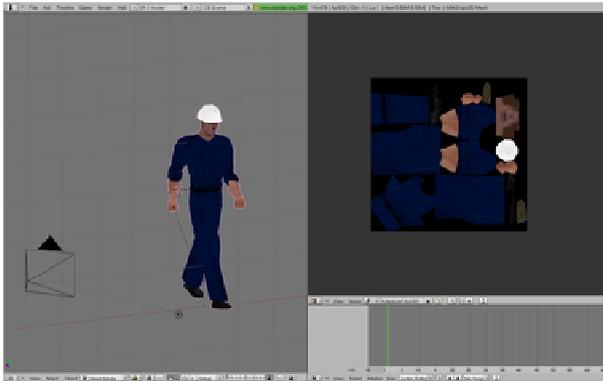


Figure 9: Blender design interface

6. RESULTS AND DISCUSSION

During a number of preliminary tests, we found the new technique delivers much better tracking result at a very low error rate, making the subject feeling much more in control of the avatar. The jitter generated in the previous prototype is almost non-existent with the new method.

In a few cases, the first view (left camera) lost track of one or two markers due to occlusion, but since the user does not need to rotate or move around, these markers are always visible from the other two cameras. Consequently, no loss of tracking ever occurs in term of 3D tracking. In term of 2D tracking, when the view loses track of the marker, it can recover instantly when the marker become visible again.

Figure 10 compares the 3D position of the right foot marker for the current and previous system. The X-axis is the frame counter (about 3 milliseconds in-between) and the Y-axis are its Z-axis disposition. The new method generates much less error in the triangulated position.

The system was tested in a single hardware with a 2.66GHz Intel Core i7-920, 6 GB DDR3, and Nvidia GeForce GTX 260+. On average, the tracking system can perform in real-time with average processing time of about 3ms per frame. Preliminary tests on a few users indicates user's acceptance on the range of walking gait and body's control. However, it was also found that other range of walking gait, such as side stepping and back stepping can be beneficial as well.

7. CONCLUSION

Walking is the most intuitive way to move around the real world. OmniWalker allows the subject to emulate real walking via the use of the stroller interface. Using the improved tracking mechanism the system allows more accurate tracking at slightly less computational cost compare to the previous prototype. The system is

still capable of real-time performance on mid-tier hardware.

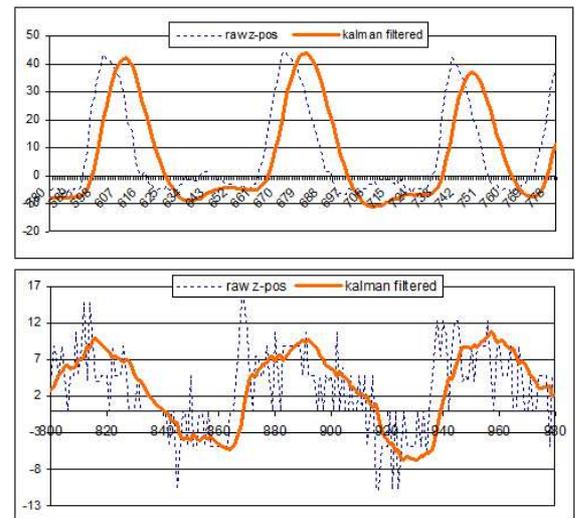


Figure 10: Kalman Filter output from the current prototype (top) and previous prototype (bottom)

We will continue to develop the system by improving the display system uses hemispherical screen instead of HMD in order to reduce motion sickness, and adding recognition of other gestures such as sidestepping and back stepping. Currently, more thorough assessment of the learning impact of the use of the system in underground coal mine safety training is in progress.

REFERENCES

- [1] M. Usuh, K. Arthur, M. C. Whitton *et al.*, "Walking > Walking-in-Place > Flying, in Virtual Environments," *SIGGRAPH 1999*, pp. 359-364, 1999.
- [2] C. A. Zambaka, B. C. Lok, S. V. Babu *et al.*, "Comparison of path visualizations and cognitive measures relative to travel technique in a virtual environment," *IEEE TVCG*, vol. 11, no. 6, pp. 694-705, 2005.
- [3] R. Darken, W. Cockayne, and D. Carmein, "The Omni-directional Treadmill: A Locomotion Device for Virtual Worlds," *Proceedings of UIST'97*, 1997.
- [4] H. Iwata, "The Torus Treadmill: Realizing Locomotion in VEs," *IEEE Computer Graphics and Applications*, vol. 9, no. 6, pp. 30-35, 1999.
- [5] J. M. Hollerbach, Y. Xu, R. R. Christensen *et al.*, "Design specifications for the second generation sarcos treadport locomotion interface," *In Haptics Symposium, Proc. ASME Dynamic Systems and Control Division*, vol. 69, 2000.
- [6] H. Iwata, H. Yano, H. Fukushima *et al.*, "Circular Floor," *IEEE Computer Graphics and Applications*, vol. 25, no. 1, pp. 64-67, 2005.
- [7] H. Iwata, H. Yano, and F. Nakaizumi, "Gait Master: A Versatile Locomotion Interface for Uneven Virtual Terrain," *Proceedings of IEEE Virtual Reality 2001 Conference*, pp. 131-137, 2001.
- [8] H. Iwata, H. Yano, and H. Tomioka, "Powered Shoes," *SIGGRAPH 2006 Conference DVD*, 2006.

- [9] H. Iwata, H. Yano, and M. Tomiyoshi, "String walker," *International Conference on Computer Graphics and Interactive Techniques*, pp. 20, 2007.
- [10] H. Noma, and T. Miyasato, "Design for locomotion interface in a large scale virtual environment. ATLAS: ATR Locomotion Interface for Active Self Motion," *Proc. ASME Dynamic Systems and Control Division*, pp. 111-118, 1998.
- [11] J.-Y. Huang, "An Omnidirectional Stroll-Based Virtual Reality Interface and Its Application on Overhead Crane Training," *IEEE Tran. Multimedia*, vol. 5, no. 1, 2003.
- [12] R. M. T. II, T. C. Hudson, A. Seeger *et al.*, "VRPN: A Device-Independent, Network-Transparent VR Peripheral System," *Proceedings of the ACM Symposium on Virtual Reality Software & Technology 2001*, 2001.
- [13] T. Svoboda, D. Martinec, and T. Pajdla, "A convenient multi-camera self-calibration for virtual environments," *PRESENCE: Teleoperators and Virtual Environments*, pp. 407-422, 2005.
- [14] M. Suryajaya, C. Fowler, and T. Lambert, "Camera-based OBDP Locomotion System," *Proceedings of the ACM Symposium on Virtual Reality Software & Technology 2009*, 2009.
- [15] J.-Y. Bouguet, "Camera Calibration Toolbox for Matlab," http://www.vision.caltech.edu/bouguetj/calib_doc/, 2008.
- [16] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Analysis Machine Intelligence*, 1995, pp. 790-799.
- [17] M. J. Swain, and D. H. Ballard, "Indexing via color histograms," *Computer Vision, 1990. Proceedings, Third International Conference on*, 1990, pp. 390-393.
- [18] R. Hartley, and A. Zisserman, *Multiple View Geometry in Computer Vision*: Cambridge University Press, 2003.
- [19] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transaction of the ASME - Journal of Basic Engineering*, pp. 35-45, March 1960, 1960.
- [20] M. H. Raibert, and J. K. Hodgins, "Animation of dynamic legged locomotion," *In Proc. ACM SIGGRAPH 1991*, 1991, pp. 349-358.