In Situ Motion Capture of Speed Skating: Escaping the Treadmill

Jeffrey E. Boyd Andrew Godbout Chris Thornton
Department of Computer Science
University of Calgary
Calgary, Alberta, Canada
{*jboyd,agodbout,chris.thornton*}*@ucalgary.ca*

***Abstract*—The advent of the Kinect depth imager has opened the door to motion capture applications that would have been much more costly with previous technologies. In part, the Kinect achieves this by focusing on a very specific application domain, thus narrowing the requirement for the motion capture system. Specifically, Kinect motion capture works best within a small physical space while the camera is stationary. We seek to extend Kinect motion capture for use in athletic training – speed skating in particular – by placing the Kinect on a mobile, robotic platform to capture motion in situ. Athletes move over large distances, so the mobile platform addresses the limited viewing area of the Kinect. As the platform moves, we must also account for the now dynamic background against which the athlete performs. The result is a novel, visually-guided robotic platform that follows athletes, allowing us to capture motion and images that would not be possible with a treadmill. We describe the system in detail and give examples of the system capturing the motion of a speed skater at typical training speeds.**

***Keywords*-Mobile robots; Robot vision systems; Robot sensing systems**

I. Introduction

Motion capture is an established technology with applications in computer games, computer animation, human motion research and clinical motion analysis. The leading technologies for motion capture are:

1) video-based marker-tracking systems (e.g., those built by Vicon [1]),
2) inertial measurement unit (IMU) systems (e.g., the motion capture suits made by Xsens [2]), and
3) exoskeletons (e.g. the Meta Motion Gypsy 7 system [3]).

Video-based systems offer excellent speed and accuracy but the acquired motion must be performed within the fields of view of multiple cameras simultaneously. Capturing motion over large areas – the size of a 400*m* track for example – is prohibitively complex and expensive. IMU systems capture motion by integrating acceleration and angular velocity data from IMUs fastened to body parts. They can operate over much larger areas but can suffer from drift in the dead-reckoning performed by the IMUs and there are no video images innately captured with the motion. Exoskeleton systems work by attaching instrumentation to the exterior of a subject's body to measure joint angles. These systems tend to be more affordable and operate over large areas, but the exoskeleton can be bulky and for some applications (e.g., athletic movements performed at high speeds) even dangerous.

Beyond the established applications of motion capture, there is a growing interest in capturing human motion to produce feedback for training athletes. For examples, see the work of Stienestra et al. [4], Schaffert et al. [5], Vogt et al. [6], Effenberg [7], and Godbout and Boyd [8]. In work where full-body motion acquisition is required, video-based capture works well if the motion can be performed in a small area or on a treadmill. However, for sports like speed skating, small areas and treadmills are not an option. A speed skating track is 400*m* long and while skating treadmills exist, they cannot duplicate turns which account for half of a track. Exoskeletons are not an option because of their bulk and safety concerns. Consequently, existing examples of motion capture for sonification (audio display of data, e.g., to provide feedback of performance data to athletes) do not capture full-body motion, and instead focus on data from a small set of sensors [4], [8]. IMU systems would work, but they do not provide video images of the skater during capture, something that is useful to coaches. Thus, there remain niches for development in the field of motion capture.

The recent introduction of the MicroSoft Kinect has provided a new option for motion capture that is cheap and available, with an abundance of information available to experimenters [9], [10]. As an example of a structured light system (e.g., Yeung [11]), the Kinect measures depth/disparity in a scene in a manner analogous to stereo vision systems. Software exists for skeletal pose estimation from Kinect depth data [12], [13]. Its prime advantage over stereo vision systems is the *active eye* that produces the structured light guarantees that the *passive eye* will see a known pattern that is ideally suited for correlating disparities. However, its intended use is for interaction with video games, and as such, is not always ideal for other applications. For example, for athletics *off the treadmill*, its viewing area is small.

This paper describes a novel robotic system that follows a speed skater on the ice with a Kinect to capture full-body motion. It offers the ability to capture motion *in situ*, on the track under real training conditions that cannot be duplicated on a treadmill. The moving, robotic platform addresses the limited performance area issues associated with visual motion capture systems and avoids safety issues surrounding exoskeletons in this application. An IMU system should also be able to capture this motion, but will not have the added advantage of providing simultaneous video of the athlete. We describe the system in details including how we address issues of speed, traction, sensing, tracking, and control for following distance and angle.

II. BACKGROUND

Although following a human appears to be a simple robotic task, there continues to be active interest in the problem from two applications areas:

- human-robot interaction (HRI) in which following a human becomes part of an interaction, and
- military applications in which a robotic assistant follows a person, or vehicles move in a convoy.

In the HRI realm, Gockley et al. [14] describe a person-following robot that uses a laser-based tracker. They compare a direction-following strategy (the robot drives toward the person) versus a path-following strategy (the robot follows the person's path) from the perspective of human interaction. They conclude that while the direction-following appears more *natural*, a hybrid approach may be best where a robot uses direction following, switching to path following in a cluttered environment. Gigliotta et al. [15] describe a follower system that locates and follows people in a domestic environment. In their design, the leader is active, i.e., it emits an infra-red beacon that the follower can see and follow. They describe their solutions to following in a cluttered, domestic environment. Sonoura et al. [16] fuse laser range data with vision data to implement a person-following robot. As with most HRI applications, their control handles both following the person and obstacle avoidance. Satake and Miura [17] describe a stereo vision system for detecting and tracking people for the purpose of following. They use a person-shaped template applied to a dense disparity/depth map. Multiple people can be tracked by associating template matches to trajectories formed by an extended Kalman filter. They describe a control strategy for a robot to follow a person based on the estimated trajectory of the tracked person.

Robot followers are useful in military applications. Giesbrecht [18] describes the development of a vision-based robotic follower system for military convoy applications. Borenstein et al. [19] look at following systems for *robotic mules* (load-carrying assistants) and convoys. They describe a system for correcting the leader's heading in the case where the follower has the most reliable navigation instrumentation. Ng et al. [20] describe a following system for armoured vehicles in jungle environments. It combines sensing, tracking and path planning to navigate in difficult terrain.

There has been some work on sensing and control issues for robotic following. Cowan et al. [21] describe a robot follower system using visual servoing from omnidirectional cameras and evaluate their system and two control strategies in simulation. Mariottini et al. [22] investigate the localization problem for robot-mounted cameras in a leader-follower configuration. They use an extended Kalman filter for position estimation and evaluate in simulation. Chen and Birchfield [23] use binocular vision to identify a person for following. They match features from a Luca-Kanade feature detector to find disparity/depth. A three-step process based on inter-frame consistency, and background and figure and motion estimates extracts the features from the moving human figure. The depth/disparity and horizontal position of the figure drive two proportional feedback loops for control.

Not surprising, the surge of interest in the Kinect has inspired a person-following robot example [24]. The system uses ROS [10] and is demonstrated at very low speeds and uses rudimentary control.

III. PHYSICAL DETAILS

A. Requirements

A robot for *in situ* speed-skating motion capture has some requirements that are out of the ordinary for person-following robots. Among these are the following.

1) The robot must be faster than most. A typical training lap of 35*s* over 400*m* corresponds to 42*km/h*. Wotherspoon's world record time of 34.03*s* for 500*m* corresponds to an *average* speed of 54*km/h*. Most small robotic development platforms are not capable of these speeds.
2) The robot must have excellent traction on ice. The turn radius on a speed skating track is approximately 32*m*. In a 35*s* lap, the robot experiences a centrifugal acceleration of approximately 0.4*g*. The robot must have traction to handle this acceleration without excessive damage to the ice surface.
3) While following directly behind the skater may be useful at times, it will also be useful to view the skater from the sides or rear quarters. Our control system must allow for this.
4) At these speeds, safety is paramount. Any system that may endanger an athlete cannot be used. It is essential to not only have manual overrides, but the sensing and control must be aware of failure states so that the robot can shutdown automatically.

B. Platforms

Advances in radio-control (RC) vehicles for hobbyists address the first of our requirements, speed. We chose a *Traxxas Slash* [25] $1/10$-scale four-wheel-drive racing truck. The manufacturer advertises speeds in excess of 55*km/h*

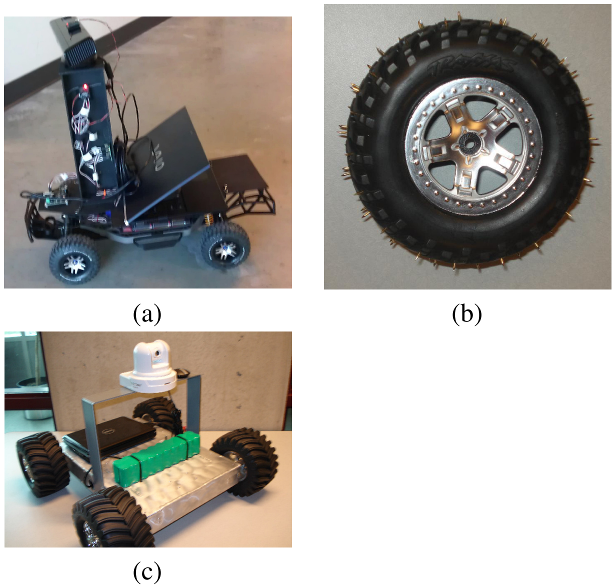


Figure 1. Robot platform for high-speed, mobile motion capture of speed skating: (a) The modified RC four-wheel-drive truck platform, (b) close-up view of spiked tires for traction, and (c) a slower, but easier SuperDroid platform for development phases.

in stock configuration with a mass of 2.3*kg*. Figure 1(a) shows the vehicle after configuration for motion capture and following.

Testing revealed that stock tires were not sufficient to hold turns beyond approximately 10 to 15*km/h*, so we took advice from hobbyists who race RC vehicles on ice and added custom spikes to a set of tires by inserting thumbtacks into the tires from the inside (see Figure 1(b)). The results were satisfactory and we confirmed both the manufacturer's claim of speeds near 60*km/h* in stock configuration, and that traction with the thumbtacks was sufficient to hold corners on the ice at these speeds.

As an additional precaution for handling on ice, we added a gyroscope control system to the vehicle's steering. With the gyroscopic control, steering commands to the vehicle set desired turn rates versus setting the wheel angle. Should the vehicle loose its traction and begin to drift (rotate too fast), the gyroscopic control adjusts the steering to compensate automatically.

Obviously, the vehicle was not designed to carry a Kinect and other equipment so further modification was required. To get the speed performance we require, it was essential that these modifications be as light as possible. We placed a plywood base reinforced with carbon-fibre rods on top of the chassis, fastened to the stock body mounts. A plywood platform holds the Kinect about 12*mm* above the base (Figure 1(a) shows the Kinect on an alternate platform, about 30*mm* above the base). A light-weight laptop computer to control the vehicle sits on the base just rear of centre.

Note that during development, it was essential to build in-

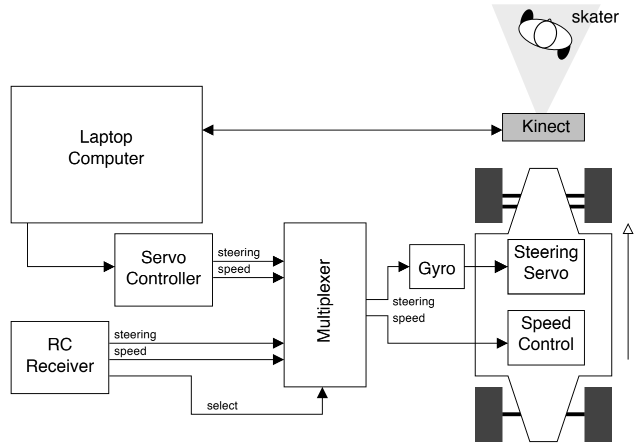


Figure 2. Schematic diagram for robotic Kinect motion acquistion system. A multiplexer switches between manual control via an RC receiver and the onboard laptop computer. The laptop computer tracks the skater with the Kinect to derive control signals for following.

crementally. To that end, we implemented and tested most of this system on a slower, but more powerful SuperDroid [26] vehicle shown in Figure 1(c). This allowed us to develop with less concern for safety and weight before progressing the the faster and lighter platform.

C. Electronics

Figure 2 shows a block diagram of the electronic and computer systems added to our robotic vehicle for following athletes. The robot requires two signals for control: steering and speed. A conventional RC servo motor operates the parallel steering mechanism on the robot. An electronic speed controller (which also uses conventional RC servo signals) drives a brushless motor to propel the robot and control speed. As mentioned previously, the robot uses a gyro to stabilize the steering when drift occurs.

The robot allows for both manual operation by a human via a radio control link and computer control. An RC servo signal multiplexer switches control between the two as dictated by the *select* channel from the RC receiver. This allows a human operator to take control of the robot at any time by operating a switch on an RC transmitter. This effectively acts as a *kill* switch when the transmitter controls are left in their neutral positions – essential for safe operation. Manual control is also handy for positioning and retrieving the robot during testing.

A laptop computer onboard the robot provides computer control. The computer uses a USB/serial controlled RC servo motor controller to provide signals that are compatible with the multiplexer and servo motor and speed controller. The computer also communicates with the Kinect to retrieve disparity maps for tracking the athlete and control. We used a particularly low-mass (approximately 1*kg*) laptop computer with a quad-core i5 processor to have sufficient, but light-weight computing power on board.

D. Kinect Mount

Mounting the Kinect camera is mostly trivial. A simple velcro attachment holds the Kinect to a mounting platform and allows us to orient the camera for different following angles. We fabricated two mounts for the Kinect camera: one 100*mm* tall and the other 300*mm* tall. While the taller platform gives us a better viewing angle, the lower platform reduces sway in the robot's suspension and is better for higher speeds. Power for the Kinect comes from a three-cell lithium-polymer battery (nominally $11.1V$). While this is slightly below the $12V$ provided by the Kinect's wall-power adapter, it does not appear to affect its operation. 1000*mAh* capacity batteries give several hours of operation between charges.

IV. Person-Following System Details

Our system uses visual servoing to enable the robot to follow an athlete. The primary advantage of this approach is that it relies on the athlete for what might otherwise be a challenging navigation task. Furthermore, in the context of an athletics track, following at close range eliminates the need for any obstacle avoidance – the athlete does this for us too.

Visual servoing has two basic components: segmentation and tracking of the person, and control of robot. These are described in the following subsections.

A. Segmentation and Tracking

Segmentation and tracking for our person-following robot has the following steps.

1) Segment the Kinect disparity image to form connected components that correspond to objects.
2) Use a multiple-target tracking system to track all objects.
3) Manually identify (e.g., by mouse click) which object trajectory to follow.
4) Use the trajectory state as input to the control system to drive the robot.

Segmentation is a variation on well-known raster-based algorithms for labelling connected components. We cannot assume that we have a stationary background as we expect the Kinect to be moving continuously. This means that any segmentation based on background subtraction is not likely to work. Instead, we form components connected by adjacent pixels with a depth difference below a specified threshold, Δz. The relationship between depth and disparity values in a Kinect *depth* image is (within the necessary precision) [10]:

$$z = \frac{348}{1090 - z_k}, \text{or} \quad (1)$$

$$z_k = 1090 - \frac{348}{z}, \quad (2)$$

where z is depth in *meters* and z_k is Kinect disparity. Obviously, a constant Δz translates to varying changes in Kinect disparity, Δz_k. To derive Δz_k we first differentiate Equation 1 to get

$$\frac{dz}{dz_k} = \frac{348}{(1090 - z_k)^2}, \quad (3)$$

then assuming

$$\frac{\Delta z}{\Delta z_k} \approx \frac{dz}{dz_k}, \quad (4)$$

gives

$$\Delta z_k = \frac{\Delta z}{348}(1090 - z_k)^2. \quad (5)$$

Thus, when forming connected components our system considers adjacent pixels with disparity values differing by less than Δz_k to be part of the same object. Examples in this paper use $\Delta z = 0.15m$.

Segmentation by depth is not sufficient for our application because the objects we want to detect and follow are almost always connected to the floor and will appear to be part of the same connected component. To deal with this, we use the following *ad hoc* process to eliminate *floor* pixels.

1) For each row in the image, perform a linear regression to find the best linear model for disparity values versus column.
2) If the linear model fits the row well, we assume that the row consists mostly of floor pixels and discard as potential object pixels all pixels that are close to the model.

A best-hypothesis multiple-target tracker tracks objects from the centroid of the segmented objects. The tracker is a simplification of the multiple-hypothesis tracker of Cox and Hingorani [27], in which the best association hypothesis is accepted in every frame, rather than defer decisions to resolve ambiguities. Accepting the best hypothesis works well here because the control system requires current estimates of object state, and there is little ambiguity while tracking a single athlete.

Each trajectory is formed from by a Kalman filter with state vector $[\ x \quad y \quad z\ ]^T$, where x and y are image coordinates and z is pixel depth. The Kalman filter uses a constant-state model which is reasonable if we assume that the control system will be servoing on the object's position. Any unassociated objects spawn new tracks. Tracks without recently associated data are discarded after a fixed number of frames (typically 30 frames, or $1s$).

An interactive display (Figure 3) shows the segmented image with all current trajectories. We engage the control system by clicking with a mouse over the desired object. The system will then follow that object as long as the trajectory is active, i.e., following stops when the tracker discards the trajectory due to lack of updates. This gives us some measure of safety – the tracking system has a recognizable failure state when it loses a trajectory that can precipitate an automatic shut down. We also provided a wireless link

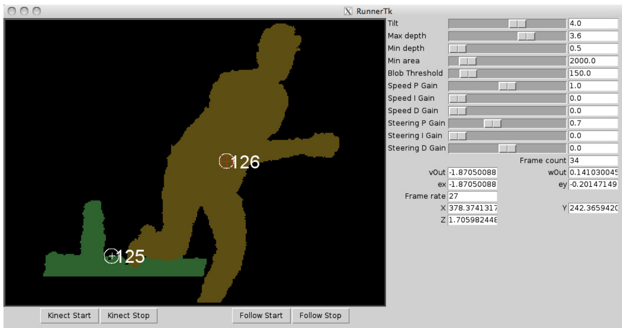


Figure 3. Screen capture of our tracking/control system in operation. Different colours indicate the different segmented objects. Crosshairs and circles indicate active trajectories from the tracker. The numbers are trajectory identifiers enumerated by the tracker. The red crosshair indicates the trajectory selected manually for following.

to the computer to allow us to monitor and tune the control system while in use.

B. Control

Following a person on an athletic track in close quarters does not require obstacle avoidance. As such, it is sufficient to follow the athlete, rather than the athlete's path. The common approach is to use a proportional control system that keeps the person centred in the field of view and at a desired distance. We elaborate slightly by allowing the robot to follow not only from behind, but from the rear quarter, and use two (steering and speed) proportional-integral (PI) controllers [28] for greater accuracy in the desired position and following distance.

Figure 4 shows schematically the geometry used to compute a position error signal to drive the controller. The input to our controllers is the desired following distance, r_{set}, and orientation θ_{set}. Via the object tracking system described in the previous subsection, the Kinect measures the range, r and azimuth, θ to the person. Referring to the robot's inertial frame of reference, the robot moves in the positive x direction, with positive y on the left-hand side. The position error in the robot reference frame is then

$$e_x = r\cos(\theta) - r_{\mathrm{set}}\cos(\theta_{\mathrm{set}}), \text{and} \tag{6}$$

$$e_y = r\sin(\theta) - r_{\mathrm{set}}\sin(\theta_{\mathrm{set}}). \tag{7}$$

Let v be the speed PI controller output and ω be the steering PI controller output. The output of the two controllers is then:

$$v = k_{v_1} e_x + k_{v_2} \int_0^t e_x dt, \text{and} \tag{8}$$

$$\omega = k_{\omega_1} e_y + k_{\omega_2} \int_0^t e_y dt, \tag{9}$$

where k_{v_1}, k_{v_2}, k_{ω_1}, and k_{ω_2} are manually tuned control gains.

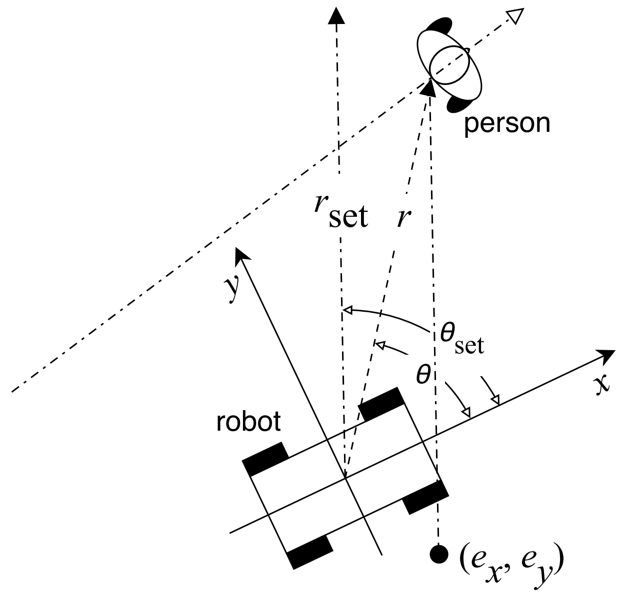


Figure 4. Geometry for computing error signals for Kinect-based person-following control.

In our implementation of the PI controllers, we constrain the integral term to stay within finite bounds for stability. We also force the integral of the steering term to be zero when robot is not moving. Note that PI control does not guarantee stability so tuning of the controller gains is essential. It is also necessary that the robot follow from behind, i.e., $|\theta_{\mathrm{set}}| \leq \pi/2$, for stable following.

V. Results

We developed the system starting with the Superdroid platform to debug our system at slower speeds, and progressed to the high-speed platform testing first in a gymnasium, then finally on ice. Figure 5 shows our robotic motion capture system in gymnasium trials. The Superdroid platform (Figure 5(a)) was limited to the speed of a brisk walk but was ideal for early testing because of its ability to carry a heavier payload than the RC vehicle, and because slower top speeds make for safer testing. Figure 5(b) shows gymnasium testing with the RC vehicle. Running speeds are easily obtained by this platform and caution is necessary because the braking performance of the vehicle is much poorer than a human runner – a human runner can sprint toward a wall and stop short while the robot cannot stop before hitting the wall. Images in Figure 5(c) through (e) show Kinect disparity image captured while following from different angles, verifying the control geometry.

Figure 6 shows on-ice testing conducted at the University of Calgary Olympic Speed Skating Oval. We started with low-speed tests with a non-athlete skater on hockey skates (Figure 6(a)) to verify traction, control, and speed capabilities up to approximately 15*km/h*. Finally, we tested with a

(a)

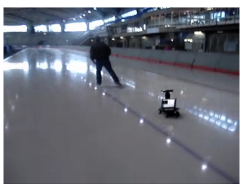
(b)

(c)

(d)

(e)

Figure 5. Motion capture system in use in a gymnasium: (a) at walking speeds and (b) running speeds. The bottom row shows Kinect disparity images for walking from (c) rear left quarter, (d) rear centre, and (e) rear right quarter, $\theta_{\text{set}} = -\pi/4$, 0, and $\pi/4$ respectively.



(a)

(b)

Figure 6. Motion capture system in use on-ice: (a) at low speeds (approximately 15*km/h*), and (b) near training speeds (approximately 30*km/h*).

trained athlete, a former Canadian national team skater as the subject (Figure 6(b)), reaching speeds over 30*km/h*.

Figure 7 shows Kinect disparity images from on-ice trials with the trained speed skater. The top two rows show part of a lap of the oval covering one straight section and one corner at approximately 30 to 35*km/h*. The bottom row shows acceleration from a start position through a straight section, accelerating up to approximately 40*km/h*. To our knowledge, this is the first example of visual motion capture of speed skating *in situ*. Moreover, it would not be possible to acquire comparable data on a treadmill for the corner and start sequences.

VI. Discussion

To date, we have achieved the individual objectives for our robotic motion capture system, although not simultaneously. That is, we can follow from arbitrary viewing angles, and we can operate at the desired speeds, but it is still difficult to do both at the same time. The normal motion of a speed skater in a straight section is side-to-side. For the trial shown in Figure 7, the skater helped by skating a straight line, something he does not ordinarily do. Furthermore, we observed that the robot could lose lock in transitions between corner and straight due to accelerations in motions we had not anticipated. Thus, the remaining obstacles to achieving our requirements fully lie in the properties of the Kinect, and improving control.

The Kinect field of view is small when moving at over 30*km/h*, especially when view from the side or rear quarters – the subject can easily leave the field of view. For example, the rightmost images of the corner and start sequence in Figure 7 show the skater at the far limit of depth measured by the Kinect. This happens due to oscillations in the speed control loop. We cannot change the properties of the Kinect, so the solution lies in the tuning of the control gains to eliminate oscillations and keep the robot close to the athlete.

In steady-state conditions, with the skater at constant velocities and minimal lateral motions, the PID controller has worked well. However, we have seen that during large accelerations, this controller is not adequate. For example, as the skater exits a corner, he makes a stride towards the inside of the corner followed by a hard stride to the right to straighten out while maintaining speed. These large left-to-right accelerations are beyond what the PID controller can handle while keeping the skater in the field of view. Future development will require that we move to better controller and possible allowing the Kinect to rotate on the platform to keep the skater in view.

VII. Conclusion

We have developed and demonstrated a system for *in situ* motion capture of speed skating. The system has produced unique data that will be valuable to coaches and athletes. Ultimately, the data will be analyzed in real-time to give skaters auditory feedback to improve their skating. While there are still some limitations, the system is ready as a tool for training.

References

[1] "Motion capture systems from vicon," Retrieved January 23, 2012, from http://www.vicon.com.

[2] "Xsens: 3d motion tracking," Retrieved January 23, 2012, from http://www.xsens.com.

[3] "Gypsy 7 electro-mechanical motion capture system," Retrieved January 23, 2012, from http://www.metamotion.com/-gypsy/gypsy-motion-capture-system.htm.

[4] J. Stienstra, K. Overbeeke, and S. Wensveen, "Embodying complexity through movement sonification: case study on empowering the speed-skater," in *9th ACM SIGCHI Italian Chapter International Conference on Computer-Human Interaction: Facing Complexity*, Alghero, Italy, May 2011, pp. 39–44.

[5] N. Schaffert, K. Mattes, and A. O. Effenberg, "A sound design for the purposes of movement optimization in elite sport using the example of rowing," in *International Conference on Auditory Display*, Copenhagen, Denmark, May 2009.

[6] K. Vogt, D. Pirró, I. Kobenz, R. Höldrich, and G. Eckel, "Physiosonic-movement sonification as auditory feedback," in *International Conference on Auditory Display*, Copenhagen, Denmark, May 2009.

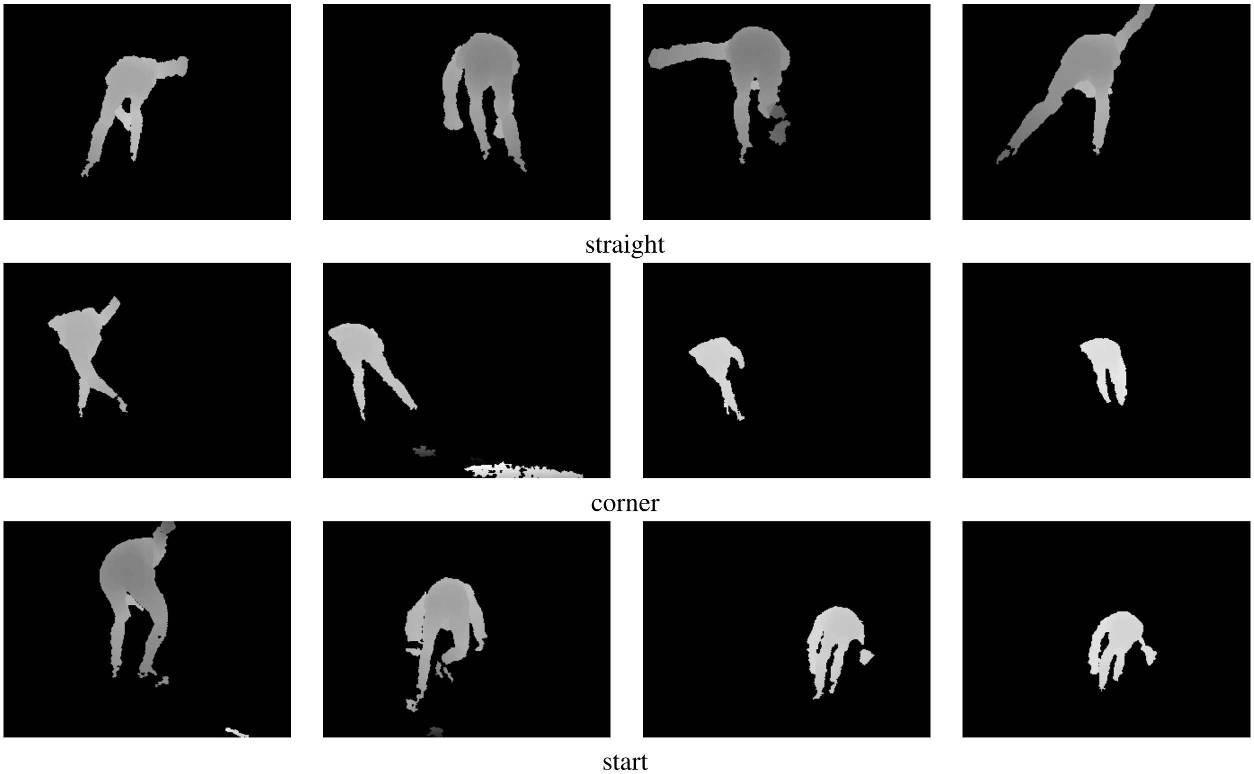


Figure 7. Kinect disparity images from on-ice testing. The top two row shows images from straight and corner track sections, images at 0.5*s* intervals, speed approximately 30 to 35*km/h*. The bottom row shows an example race start with skater accelerating along a straight section of track up to approximately 40*km/h*.

[7] A. O. Effenberg, "Movement sonification: Effects on perception and action," *IEEE Multimedia*, vol. 12, no. 2, pp. 53–59, 2005.

[8] A. Godbout and J. E. Boyd, "Corrective sonic feedback for speed skating: a case study," in *International Conference on Auditory Display*, Washington, DC, June 2010, pp. 23–30.

[9] "Open kinect project," Retrieved Januray 19, 2012, from http://openkinect.org/wiki/Main_Page, 2012.

[10] "ROS (robot operating system)," Retrieved Januray 19, 2012, from http://www.ros.org/wiki, 2012.

[11] K. K. Yeung, "A low cost three-dimensional vision system using space-encoded spot projection," Master's thesis, Department of Electrical Engineering, University of British Columbia, Vancouver, BC, Canada, 1985.

[12] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011, pp. 1297–1304.

[13] "OpenNI," Retrieved Januray 23, 2012, from http://openni.org/, 2012.

[14] R. Gockley, J. Forlizzi, and R. Simmons, "Natural person-following behavior for social robots," in *Human Robot Interaction*, Arlington, VA, March 2007, pp. 17–24.

[15] O. Gigliotta, M. Caretti, S. Shokur, and S. Nolfi, "Toward a person-follower robot," in *Second Robocare Workshop*, vol. 2, 2005, pp. 65–68.

[16] T. Sonoura, T. Yoshimi, M. Nishiyama, H. Nakamoto, S. Tokura, and N. Matsuhira, "Person following robot with vision-based and sensor fusion tracking algorithm," in *Computer Vision*, X. Zhihui, Ed. Vienna, Austria: InTech, 2008, ch. 27, pp. 519–538.

[17] J. Satake and J. Miura, "Robust stereo-based person detection and tracking for a person following robot," in *ICRA Workshop on People Detection and Tracking*, Kobe, Japan, May 2009.

[18] J. Giesbrecht, "Development of a vision-based robotic follower vehicle," Defence Research and Development Canada - Suffield, Tech. Rep. TR 2009-026, February 2009.

[19] J. Borenstein, D. Thomas, B. Sights, L. Ojeda, P. Bankole, and D. Fellars, "Human leader and robot rollower team: correcting leaders position from followers heading," in *Unmanned Systems Technology XII*, vol. SPIE Vol. 7692, Orlando, FL, April 2010.

[20] T. C. Ng, J. Ibaez-guzmn, J. Shen, and Z. Gong, "Vehicle following with obstacle avoidance capabilities in natural environments," in *International Conference on Robotics and Automation*, Sendai, Japan, September 2004, pp. 4283–4288.

[21] N. Cowan, O. Shakerina, R. Vidal, and S. Sastry, "Vision-based follow-the-leader," in *International Conference on In-*

telligent Robots and Systems, Las Vegas, NV, October 2003, pp. 1796–1801.

[22] G. L. Mariottini, G. Pappas, D. Prattichizzo, and K. Daniilidis, "Vision-based localization of leader-follower formations," in *Decision and Control 2005 and 2005 European Control Conference*, Seville, Spain, December 2005, pp. 635–640.

[23] Z. Chen and S. T. Birchfield, "Person following with a mobile robot using binocular feature-based tracking," in *International Conference on Intelligent Robots and Systems*, San Diego, CA, October 2007, pp. 815–820.

[24] G. Gallagher, "Follower robot," Retrieved Januray 22, 2011, from http://www.ros.org/wiki/openni/Contests/ROS 3D/Follower Robot, 2011.

[25] "Traxxas - the fastest name in radio control," Retrieved January 23, 2012, from http://traxxas.com/.

[26] "Super droid robots," Retrieved January 23, 2012, from http://www.superdroidrobots.com/.

[27] I. J. Cox and S. L. Hingorani, "An efficient implementation of reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 2, pp. 138–150, 1996.

[28] K. Ogata, *Modern Control Engineering*. Englewood Cliffs, NJ: Prentice-Hall, 1970.