Inexpensive Ground Truth and Performance Evaluation for Human Tracking using multiple Laser Measurement Sensors

William Shackleford
shackle@nist.gov

Tsai Hong
Tsai.Hong@nist.gov

Tommy Chang
Tommy.Chang@nist.gov

National Institute of Standards and Technology(NIST)
100 Bureau Dr.
Gaithersburg, MD 20878

ABSTRACT

This paper will describe a flexible and inexpensive method of obtaining ground truth for the evaluation of Human Tracking systems. It is expected to be appropriate for evaluating systems used to allow robots and/or autonomous vehicles to operate safely around humans. It is currently focused on tracking people as they stand still or walk. It relies on multiple Laser Measurement Sensors(LMS) also called laser line scanners. The LMS's are mounted to scan in a horizontal plane. A method for quickly calibrating the relative position and orientation of each of the sensors to each other is described. A basic human tracking algorithm using the LMS's is described along with how the algorithm can be combined with a priori knowledge of the walkers intended path during the test. A graphical user interface(GUI) displays both the data obtained directly from the LMS and the output of the tracking algorithm. The GUI allows the user to verify and adjust the tracking algorithm without needing to annotate every frame, and therefore at a lower cost than systems that require extensive annotation. Tests were performed with people walking or running though several patterns, while data was simultaniously recorded by a more expensive system require individual receivers on each participant for comparison.

Categories and Subject Descriptors

I.2.10 [**Vision and Scene Understanding**]: 3D/stereo scene analysis

General Terms

Performance, Measurement

Keywords

Human Tracking, Laser Measurement Sensor(LMS),Ground Truth

This paper is authored by employees of the United States Government and is in the public domain.
PerMIS'10, September 28-30, 2010, Baltimore, MD, USA.
ACM 978-1-4503-0290-6-9/28/10.

1. INTRODUCTION

While robots and autonomous vehicles have found applications in numerous extreme environments from Mars to the bottom of the ocean, one of the most difficult environments for robots to work in are those where the safety of people moving within the environment needs to be addressed. In an industrial setting, robots are usually required to be fenced off from people and shutdown completely when people need to breach the fence. [7] [6] In order for robots and autonomous vehicles to operate in closer proximity to people, tests of their perception systems need to be devised that will thoroughly prove the system's safety. The evaluation of these tests requires ground truth knowledge of the location of all humans in the environment during the test. In [15], thermal imaging is used to track humans for real-time mobile robot applications. In [5], laser scanners are combined with video to track humans in a scene. Ultra-WideBand (UWB) technology is used in [4]. However most of these systems were designed to be used within a fully autonomous system while others require expensive support infrastructure. Our approach is to employ multiple LMS's mounted to scan horizontally at a fixed height between the foot and knee with a Graphical User Interface(GUI) for annotation and post-processing to develop ground truth data to be used in the evaluation of other systems.

2. LMS MOUNTING

Figure 1 shows our apparatus for mounting each LMS. The roll and pitch can be adjusted to level by adjusting the height of each of the three feet. It is light weight enough for one person to carry easily. The height of the sensor is adjustable.

The height may be a crucial factor for which future research may be needed to provide additional guidance. The following factors should be considered in selecting the height.

- Occlusions – At very low heights, grass may occlude the sensor.
- Size of a person in the horizontal plane – At longer distances it will be more important to use a height where the person is wider in order to get more data points on the person.
- Person to person variability – People are expected to have more similarly shaped scans at ankle height than

Figure 1: Adjustable and portable apparatus for mounting the LMS

at a height that might correspond with one persons shoulders and another persons chest.

- Separation between people – People might be more likely to brush shoulder to shoulder than ankle to ankle.

3. DATA COLLECTION

Each LMS sensor was read using tools in the Mobile Open Architecture Simulation and Tools(MOAST) framework. [3] [1] [2] This provides a real-time live connection for the new programs for display and analysis described in this paper via the Neutral Messaging Language(NML) [13] [9] [10] as well as offline access through NML Packed Message Files corresponding to the MOAST defined SensorData1D data structure recorded by the MOAST Data Logger. [8] The advantages of this approach have already been tested in [12] and [11]. [1]

4. MANUAL VISUAL CALIBRATION

We prefer to be able to combine the output of multiple sensors. This allows us to see both sides of a person's legs, provides redundancy in case one sensors view is obstructed, and allows us to extend the area in which the experiments may be conducted. One of the first tasks to be performed before the output of multiple sensors can be combined is to determine the relative position and orientation of each of the sensors so that data can be represented in a common coordinate system. One simple approach is to use a graphical interface that allows the output of one scan to be overlaid on another and dragged into position while recording the position and orientation offset needed for alignment. A screen shot of this graphical interface is shown in Figure 2. The display shows an overhead view of the scene. While Z values are carried along in the computation, they have no effect on the display. In this figure the data from one sensor is plotted in blue while the data from the other sensor is plotted in red. Two items in this scene are useful for doing the calibration, a poster board held up by the corner of a table and a cylinder.

[1]All the source code for the programs described in this paper are also now available from the MOAST repository. http://sourceforge.net/projects/moast/

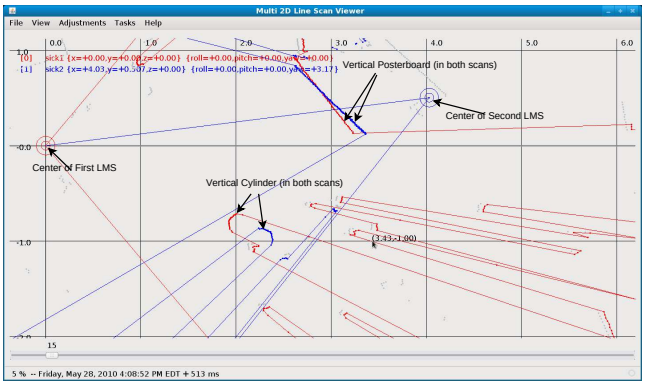

Figure 2: Screenshot of Graphical User Interface(GUI) for Manual Calibration before calibration. Points from LMS1 are in red. Points from LMS2 are in blue.

The GUI has a mode for adjusting each degree of freedom independently. In each mode, the point the mouse is pressed down on is dragged with the mouse position by adjusting the degree of freedom associated with that mode. When the user presses the mouse down, the screen coordinates of the mouse pointer are recorded ($D_{screen} = \{D_{ScreenX}, D_{ScreenY}\}$) as well as the current offsets of the selected sensor ($O = \{x_o, y_o, z_o, roll_o, pitch_o, yaw_o\}$). While dragging the position offset x,y and orientation offset roll,pitch,yaw are updated from the current mouse position ($U_{screen} = \{U_{ScreenX}, U_{ScreenY}\}$). Both the mouse down position and mouse up position are converted from screen to world coordinates using Equations (1) and (2). The scale depends on the zoom level and is the ratio of meters to pixels on the display. The XOffset and YOffset variables are determined by the amount the user has scrolled the view. The corresponding sensor coordinates of the mouse down and up positions are converted from world coordinates to the selected sensor's coordinate by multiplying by the inverse of the pose created from ($O^{-1} = Posemath.inv(\{x_o, y_o, z_0, roll_o, pitch_o, yaw_o\})$) using the Real-Time Control System(RCS) Posemath Library.[14] Updating the x and y offsets is trivial as seen in Equations (3) and (4). $roll_f$ and $pitch_f$ are the roll and pitch fractions, temporary variables to reduce the complexity of Equations (5) and (6) used to update roll and pitch. Equation (7) updates the yaw value.

$$x_{world} = Scale * (x_{screen}) + XOffset \quad (1)$$

$$y_{world} = Scale * (y_{screen}) + YOffset \quad (2)$$

$$x = x_o + (U_{WorldX} - D_{WorldX}) \quad (3)$$

$$y = y_o + (U_{WorldY} - D_{WorldY}) \quad (4)$$

$$D_{Sensor} = \{D_{SensorX}, D_{SensorY}\}$$
$$= Posemath.multiply(O^{-1}, D_{World})$$

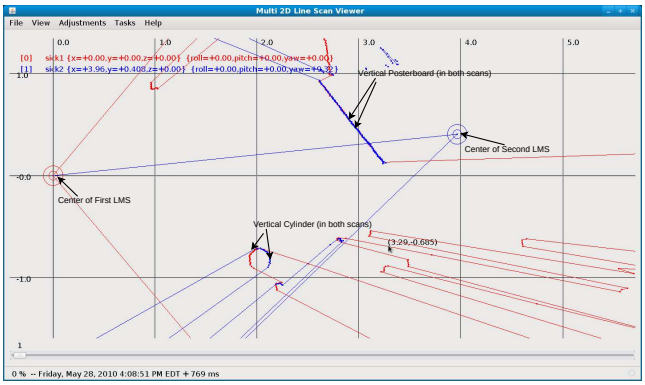

Figure 3: Screenshot of Graphical User Interface(GUI) for Manual Calibration after calibration. Points from LMS1 are in red. Points from LMS2 are in blue.

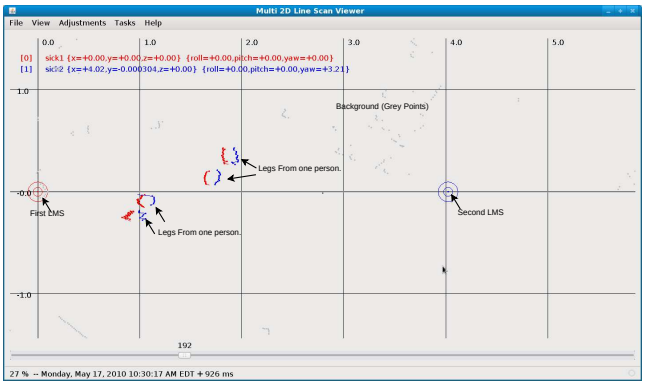

Figure 4: Screenshot of Graphical User Interface(GUI) showing the use of Background Subtration. Points in grey are part of the background and not processed further.

$$U_{Sensor} = \{U_{SensorX}, U_{SensorY}\}$$
$$= Posemath.multiply(O^{-1}, U_{World})$$

$$roll_f = cos(roll_0) * \frac{U_{SensorY}}{D_{SensorY}}$$

$$roll = \begin{cases} \pi & \text{if } roll_f \leq -1 \\ acos(roll_f) & \text{if } -1 < roll_f < 1 \\ 0 & \text{if } roll_f \geq 1 \end{cases} \quad (5)$$

$$pitch_f = cos(pitch_0) * \frac{U_{SensorX}}{D_{SensorX}}$$

$$pitch = \begin{cases} \pi & \text{if } pitch_f \leq -1 \\ acos(pitch_f) & \text{if } -1 < pitch_f < 1 \\ 0 & \text{if } pitch_f \geq 1 \end{cases} \quad (6)$$

$$yaw = atan(\frac{U_{SensorX}}{U_{SensorY}}) - atan(\frac{D_{SensorX}}{D_{SensorY}}) + yaw_o \quad (7)$$

The advantage of manual visual calibration is that no automatic algorithm is needed to segment the data to find points of correlation in the two or more line scan data sets. No particular calibration target is needed although simple objects with flat or circular vertical surfaces make identifying them in the display easier. The sensors were mounted on adjustable and portable frames which provides flexibility in collecting data but means that frequent calibrations are required. The GUI was intuitive enough that recalibrations could be completed in only a couple of minutes. Figure 3 shows the result of one such recalibration.

5. BACKGROUND SUBTRACTION

After the relative positions of each of the sensors have been calibrated, the next step is to separate possible humans from the background objects such as walls and furniture. One approach to doing this is to collect scans that will only contain the relatively fixed objects in the scene. The sensor values are collected over some period of time and the minimum range value from each sensor at each angle is stored in a background file. The background is then removed from the data by removing any point whose range is greater than the minimum recorded when establishing the background minus a threshold(10cm). This is shown in Figure 4.

6. LEG AND HUMAN GROUPING

Each frame of data whether read from the live sensor or from data logs contains the same SensorData1D data structure. It contains the field of view, angular resolution, a time stamp and an array with a series of range values. The sensor has different modes that can change the field of view and angular resolution. Each range value is tested against the background and values within 10cm of the background are removed from further processing. The 10cm tolerance was found to reliably remove the background. If detecting people deliberately pressing themselves up against a wall or other background object were necessary, the tolerance could be reduced. Range values that differ from the previous or next by more than 10cm are also removed. Range values of the edges of objects may have a range that represents an averaging over the beam width of the object and the farther off background. This is sometimes called the mixed-pixel effect. Eliminating points before and after changes eliminates points likely to be on edges. This tolerance can also be adjusted. The program maintains a list of humans being tracked. For each human being tracked, the program keeps a history of that person's positon over time, the current centroid of the points associated with each leg, the centroid of the two leg centroids and a list of data points from the current frame associated with each leg. Each range value is converted to a 2D cartesian point using the calibrated offset for that sensor. An object for all possible pairs of points less than twice leg tolerance apart is created and the set of pairs sorted based on the distance between pairs. Each pair is evaluated in order starting with the closest.

There are three cases:

- If neither point in the pair belongs to a leg group then create a new group and add both points.
- If exactly one of the two points belongs to a leg group

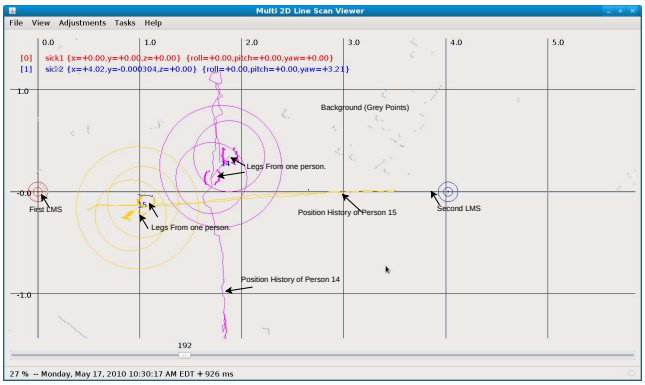

Figure 5: Screenshot of Graphical User Interface(GUI) showing the grouping of each person and the track of that person's movement.

check if the unassociated point can be added to the other points group.

- If both points belong to different groups check to see if the two groups can be merged. (If both points belong to the same group there is nothing to do.)

Once the points have been grouped into leg groups, the legs are associated with humans. Each leg group is tested against the set of human groups created from the previous frame of data in order based on the distance to the closest leg. The leg is associated with that human group if all of the following are true.

- The human does not already have two legs associated with it from the current frame.
- The leg is not farther than the tolerance in the expected path of that human.
- The leg is not farther than the leg tolerance away from the leg positions of the human in the previous frame.

If the leg is not closer to any human from the previous frame than the configurable threshold then a new human is added to the set.

This is shown in Figure 5.

7. TEST SETUP

In order to evaluate the effectiveness of this system a series of tests were performed where the tracks of humans were simultaneously recorded with this system and a system that uses four laser transmitters on tripods in each corner of the room and battery powered receivers mounted on hard hats to be worn by each participant. The transmitter/receiver system is a more expensive system that was expected to also have higher fidelity. Since each receiver has a unique identifier it is not as subject to errors where two people coming close together may be mistaken for one person or each other. The test consisted of seven patterns.

- One person walking in a circle.
- One person walking in a zigzag pattern with sharp turns.
- Two people walking side by side parallel to an imaginary line between the laser measurement sensors.
- Two people walking side by side parallel to a line orthogonal to an imaginary line between the laser measurement sensors.
- Two people crossing the center of the test area from opposite sides parallel to an imaginary line between the laser measurement sensors.
- Two people crossing the center of the test area from opposite sides parallel to a line orthogonal to an imaginary line between the laser measurement sensors.
- Two people crossing the center of the test area from corners ninety degrees apart.

Each pattern was repeated at fast and slow speeds and with the height of the laser scanners at 0.14 m (0.45 ft), 0.4m (1.3 ft), and 0.6 m (1.96 ft). At the lowest height we also collected data a medium speed.

8. TEST RESULTS

Figure 6 shows overlapped plots of 2D overhead positions of each person in each test scenario showing both the data from the transmitter/receiver system and from the laser measurement scanners. In Figure 7 the maximum and average error for each test is plotted along with variables affecting/identifying the test. Although the computer controlling the laser scanner and the computer controlling the transmitter/receiver system were synchronized using the Network Time Protocol(NTP), analysis showed strong evidence that the transmitter/receiver system had delays up to 3 seconds that varied even during the course of a single test. Therefore errors were computed by finding the distance between each position reported by the transmitter/receiver system and the closest point reported by the human tracking algorithm using the line scanner data taken in the previous three seconds. In addition some data reported by the transmitter/receiver system was ignored due to large temporary jumps in position. The maximum error between the two systems over the test as shown in the top plot of Figure 7 varied between 0.10 m (0.32 ft) and 0.76 m (2.49 ft). The mean error as shown in the second to top plot of Figure 7 varied between 0.04 m (0.13 ft) and 0.23 m (0.75 ft). Both the mean and maximum errors peaked during the fast circle tests. There are two theories for the large errors during the fast circle tests. The first is that as a person runs around a circle they lean into the circle such that the receiver on a person's head travels a smaller radius circle than a system detecting their feet. The other theory is that the transmitter/receiver system may be doing time averaging that also might tend to reduce the size of the circle traveled. Some tuning was required after the data were collected. The default human threshold of 0.9 was chosen but was raised for two of the tests and lowered for four of them as shown in the second to bottom plot of Figure 7. When people are walking faster, especially when the sensor was mounted at the lowest height, their legs tended to move farther away from each other, which led the algorithm to mistake one person for two. This was corrected by increasing the threshold for human radius. Tests where people are closer to each other caused one leg of one person to be grouped with the wrong

person. This was corrected by reducing the threshold for the human radius. For four of the tests the tracking algorithm was provided with a priori expected paths, which prevented the algorithm from associating a leg with the person on the other track in error. While the software supports expected paths of any number of line segments the expected paths provided for this test only consisted of a single line segment for each person. This meant the effort to provide the expected path was minimal. The paths were only needed for the tests where two people crossed paths at right angles and there was a potential to swap paths.

9. CONCLUSIONS

While some ground-truth systems are costly because they require expensive hardware and infrastructure and others require extensive hand segmentation, it is possible to produce ground-truth from less expensive systems. The key is that the ground-truth system does not need to be perfect, it only needs to be much less subject to the types of errors expected from the system under test in the particular experiment. For example while the system under test might only have views available from a moving autonomous vehicle, the ground truth system has multiple views available from fixed locations. The system under test may be required to handle occlusions while the experiment is designed to ensure the ground truth system is never occluded.

Specifications of the error range for systems for human tracking require a better definition of the position of a human. While some systems specify millimeter accuracy or better, this is typically for the position of a target on a human. The relationship of the target to the person's center of mass or a bounding volume containing the human is left unknown. The range of errors on each data point are easily obtainable from the manufacturer of the laser measurement scanner. These errors are typically only a centimeter or two and may even be in the millimeter range. However the error introduced by assuming a person's center of mass or that the center of their head is directly above the center of the cluster of points on the legs or ankles is likely greater than this. For robot safety applications, even knowing the center of mass to millimeter accuracy is not useful. To avoid hurting a person, the robot must either have the full bounding volume of the person including things that they are carrying, or assume a radius large enough to include these.

Further work will be done to determine the suitability of this system outdoors and at much larger ranges.

10. REFERENCES

[1] S. Balakirsky. MOAST on sourceforge.net. `https://sourceforge.net/projects/moast/`.

[2] S. Balakirsky. MOAST wiki. `http://tinyurl.com/2ahr2qg`.

[3] S. Balakirsky, F. Proctor, C. S. Jr, and T. Kramer. A mobile robot control framework: From simulation to reality. In *International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR 2008)*, November 2008. `http://tinyurl.com/28925aw`.

[4] J. A. Corrales, J. A. Corrales, F. A. Candelas, and F. Torres. Hybrid tracking of human operators using imu/uwb data fusion by a kalman filter. In *Proceedings of the 3rd ACM/IEEE international conference on*

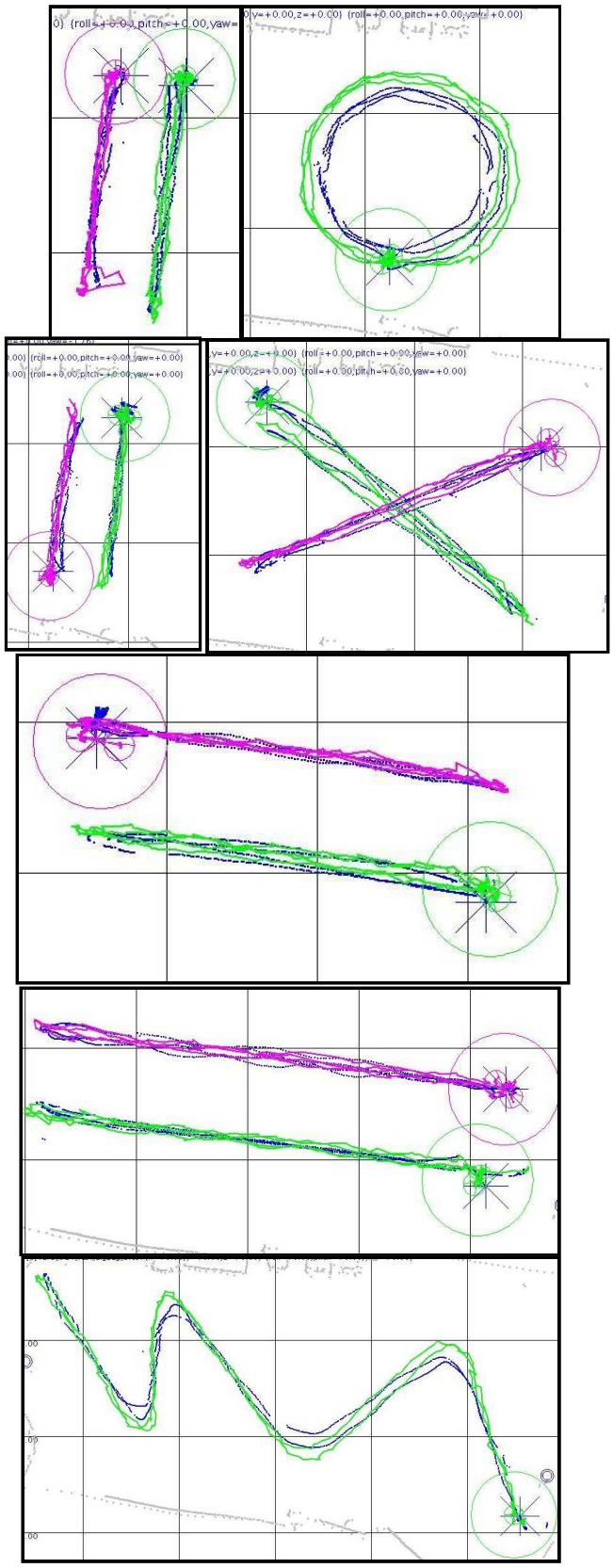

Figure 6: Results for the 0.14m height fast tests for each walking pattern. data points from the transmitter/receiver system in dark blue, background range scanner data in grey, track from person 1 in as detected by laser scanners in green, track from person 2 in purple. The light grid marks 4 m (13.1ft) squares. The darker lines border each test plot. Each test has three laps back and forth or around.

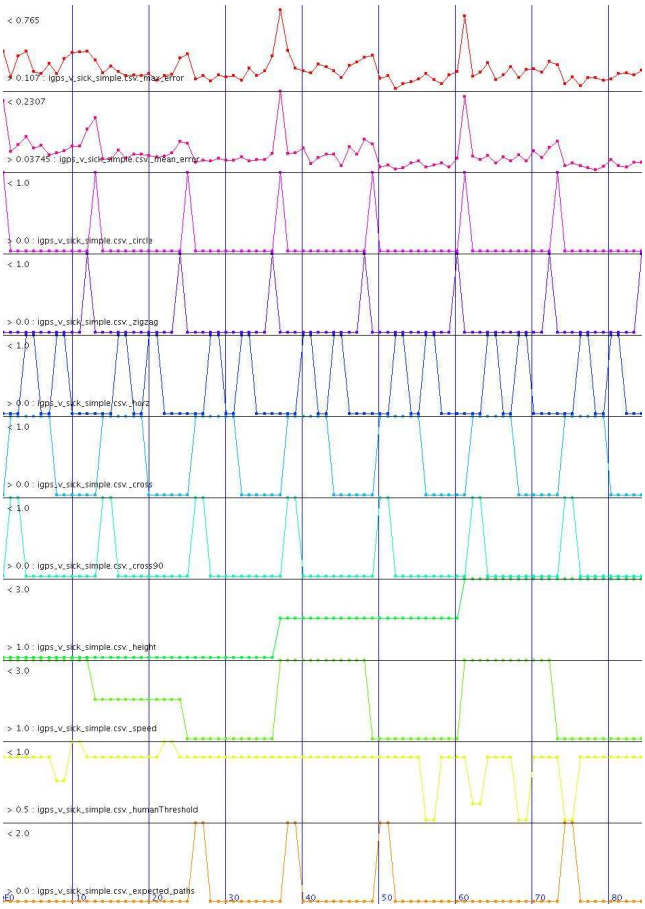

Figure 7: Plots of the summary results for each test. X axis is the test number, each variable is plotted with a different min and max scale shown.

Human robot interaction, 2008. `http://portal.acm.org/citation.cfm?id=1349822.1349848`.

[5] J. Cui, H. Zha, H. Zhao, and R. Shibasaki. Multi-modal tracking of people using laser scanners and video camera. *Image and Vision Computing*, February 2008. `http://tinyurl.com/24u2crn`.

[6] I. T. S. D. F. (ITSDF). Itsdf b56.5 safety standard for guided industrial vehicles and automated functions of manned industrial vehicles. `http://www.itsdf.org`.

[7] R. I. A. (RIA). *ANSI/RIA R15.06-1999 American National Standard for Industrial Robots and Robot Systems - Safety Requirements (revision of ANSI/RIA R15.06-1992)*. Robotic Industries Association (RIA), 1999.

[8] W. Shackleford. NML message files. `http://www.isd.mel.nist.gov/projects/rcslib/message_files.html`.

[9] W. Shackleford. The NML programmer's guide (c++ version). `http://www.isd.mel.nist.gov/projects/rcslib/NMLcpp.html`.

[10] W. Shackleford. The NML programmer's guide (java version). `http://www.isd.mel.nist.gov/projects/rcslib/NMLjava.html`.

[11] W. Shackleford. Performance evaluation of the primitive mobility execution of an automated guided vehicle within the mobility open architecture simulation and tools environment. In *Proceedings of the 14th World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI 2010 Orlando, FL, United States*, June 2010.

[12] W. Shackleford and R. Bostelman. Data collection test-bed for the evaluation of range imaging sensors for ANSI/ITSDF b56.5 safety standard for guided industrial vehicles. In *Proceedings of the Performance Metrics for Intelligent Systems Workshop*, October 2009. `http://tinyurl.com/26w96wd`.

[13] W. Shackleford, F. Proctor, and J. Michaloski. The neutral message language: A model and method for message passing in heterogeneous environments. In *Proceedings of the World Automation Conference*, June 2000. `http://tinyurl.com/2bwokte`.

[14] W. Shackleford, F. Proctor, and K. Murphy. RCS pose mathematics library. `http://www.isd.mel.nist.gov/projects/rcslib/posemathdoc/`.

[15] A. Treptowa, G. Cielniakb, and T. Duckett. Real-time people tracking for mobile robots using thermal vision. *Robotics and Autonomous Systems*, September 2006.