2015 IEEE International Conference on Robotics and Automation (ICRA)
Washington State Convention Center
Seattle, Washington, May 26-30, 2015

SNAPCAT-3D: Calibrating Networks of 3D Range Sensors for Pedestrian Tracking

Dylan F. Glas, Dražen Brščić, *Member, IEEE,* Takahiro Miyashita, and Norihiro Hagita, *Member, IEEE*

***Abstract*— The use of 3D range sensors for human position tracking has grown in recent years, especially for augmenting robotic sensing for human-robot interaction. However, extrinsic calibration of the relative positions of 3D range sensors is difficult, due to their limited range, narrow field of view, and distortion at large distances. 2D laser range finders have also been used for pedestrian tracking, providing greater accuracy and coverage at the cost of being more expensive and susceptible to occlusion. In this work, we present two novel techniques for calibrating the positions of 3D range sensors based on shared observations of pedestrians. The first technique uses 3D range sensors alone, and the second technique uses 2D and 3D range sensors together, using the high precision and long range of the 2D sensors to complement the short-range but richer sensing of 3D range sensors. We evaluate the accuracy of both automatic calibration techniques, and we furthermore show that the combination of 2D and 3D sensors gives more robust and accurate calibration than when using 3D sensors alone.**

Index Terms —person tracking, laser range finders, 3D range sensors, sensor calibration

I. Introduction

A. Pedestrian Tracking with Range Sensors

For robots operating in real social environments, accurate tracking of people is important for safety, motion planning, and effective human-robot interaction. Although short-distance tracking with on-board sensors such as [1] may be sufficient for basic robot safety, wide-area tracking provided by external sensors can help avoid deadlock in crowded environments [2], analysis and anticipation of pedestrian behavior [3], dynamic path planning to approach people [4], and other interactions with humans, such as "friendly patrolling" [5], and handing out flyers [6]. Finally, position tracking systems have been shown to be important even for stationary robots such as a seated android, in order to support tasks like gaze control and eye contact [7].

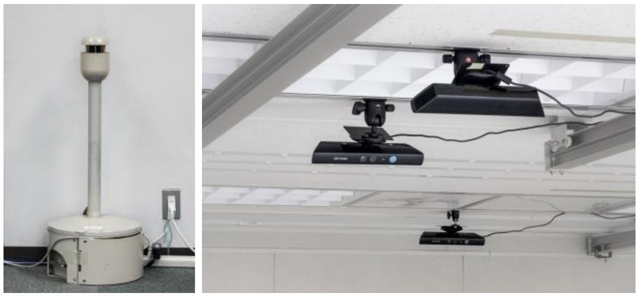

Figure 1. Left: Portable pole with laser range finder for 2D tracking system. Right: Ceiling-mounted 3D range sensors (Kinect) used for pedestrian tracking.

Although many techniques have been used for pedestrian tracking, such as video-based tracking, motion-capture systems, and others [8, 9], 2D laser range finders (LRF's) such as the Hokuyo UTM series and 3D range sensors such as the Microsoft Kinect or Asus Xtion PRO (Fig. 1) are growing in popularity for robust tracking of large numbers of pedestrians in real social spaces. 2D range sensors can scan large areas with high precision, but are more susceptible to occlusion in crowds, whereas ceiling-mounted 3D range sensors collect richer data and are less affected by occlusions, but their sensing range is quite limited.

Many systems have been developed for the study of human interactions using LRF's [10-13] and 3D range sensors both on-board the robot [14] and mounted in the environment in various configurations [15-17]. For any such multiple-sensor configurations, the problem of extrinsic sensor calibration is important, with direct impact on the tracking accuracy the system can provide. This work will focus on ceiling-mounted 3D range sensors and waist-height 2D LRF's [18, 19].

In this work we present two techniques developed for a calibration software tool we call "SNAPCAT-3D" (Sensor Network Automated Position Calibration and Alignment Tool). First, we present a practical technique for calibration of a network of ceiling-mounted 3D range sensors in 6 degrees of freedom (DOF) based on pedestrian observations. We then propose a second technique, in which the addition of 2D LRF's can be used to improve the 3D sensor calibration by greatly increasing the connectivity of the network and hence the number of shared observations.

B. Related work

The problem of calibrating sensor positions has been addressed for several kinds of sensors. For example, Senior et al. developed a visual technique for automatic video camera calibration [20], and reference [21] provides a survey of available techniques for sensor localization for ubiquitous computing applications. However, to our knowledge a 6-DOF calibration technique applicable to ceiling-mounted 3D range sensors has not yet been developed.

In previous work, we developed a technique for calibration of 2D sensors [22], wherein the optimal positions of sensors are computed to minimize the error between shared observations of pedestrians from different sensors. A similar

*This research was supported in part by the Ministry of Internal Affairs and Communications of Japan with a contract entitled "Novel and innovative R&D making use of brain structures," and in part by JST, ERATO.

D. F. Glas, D. Brščić, T. Miyashita, and N. Hagita are with ATR Intelligent Robotics and Communication Labs., Kyoto, Japan. D. F. Glas is also with the Ishiguro Symbiotic Human-Robot Interaction Project, ERATO, JST. (corresponding author's phone: +81-774-95-1405; fax: +81-774-95-1408; e-mail: dylan@atr.jp).

978-1-4799-6922-7/15/$31.00 ©2015 IEEE

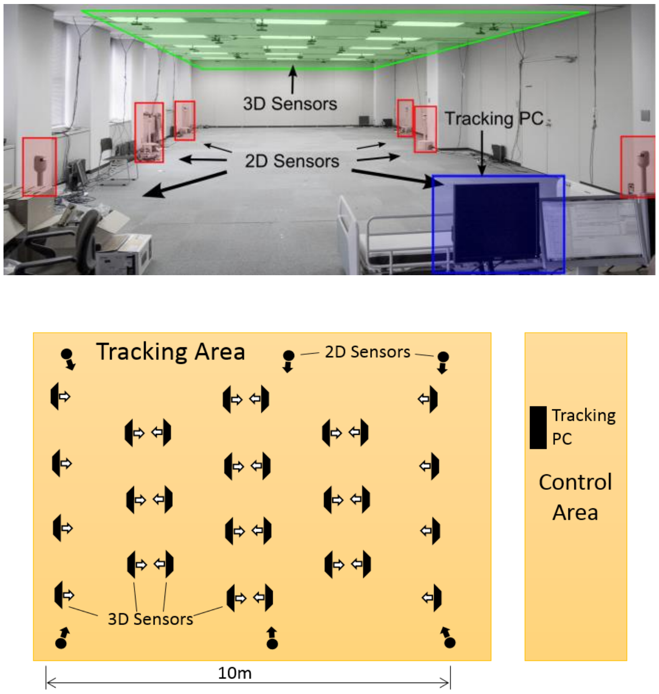

Figure 2. Photo and layout diagram of sensor room, showing approximate positions of 28 3D sensors and 6 2D sensors.

technique has been proposed by Schenk et al. [23]. In this work we build upon our previous technique, addressing new issues specific to the calibration of 3D sensors.

II. Design Considerations

Ceiling-mounted 3D range sensors pose several challenges preventing the use of common sensor calibration methods.

1) Point cloud matching

One approach for calibration might be to do a registration of the point clouds of static objects in the background, between all sensors. This approach would be similar to that taken with multiple 3D scans obtained using SLAM or other methods, *e.g.* [24]. However, there are several reasons why this approach cannot be used in the present scenario.

First, the point clouds themselves are distorted. Because our sensors are mounted on the ceiling to track people's heads, static background objects are typically located beyond the nominal range of the sensors. At such distances, there are many missing measurements, the measurements are more affected by noise, and range distortion occurs. Distortion of distant points varies between sensors, and even within one sensor over time [25].

Second, it is often difficult or impossible to identify shared features between the sensors. In part, this is because the sensors are fixed and have a relatively narrow field of view. In addition, the sensors mostly see floors and walls, which do not provide useful features for point cloud matching.

2) Floor plane extraction

Using the floor plane as a reference also appears promising, at least for calibrating pitch, roll, and height of the sensors, but as the floor is typically far beyond the nominal range of the sensor, distortion of measurements is again a problem. Furthermore, there is no way of resolving the ambiguity between observations of floors and walls.

3) Markers

Another option is to place markers in the environment, as 3 or more markers at known positions enable the 6D pose of one sensor to be calculated [26]. However, given the presence of distortion and noise and the large number of sensors to be calibrated, a large number of markers will be necessary to achieve good calibration. Using pedestrians as references achieves this goal with lower effort and less specialized equipment compared with fixed markers. Additionally, if the purpose of the tracking system is to observe a natural social environment such as a shopping mall or classroom, then it is preferable not to disturb the business or social environment with markers or calibration targets.

4) Hybrid sensing networks

In the case that 2D and 3D sensors are used together for tracking, it is necessary to calibrate sensors of both types in a shared coordinate system. This poses the problem of how to create shared observations visible to both kinds of sensors, as the nature of the data is quite different between sensor types.

5) Proposed solution

In this work, we focus on using observations of pedestrians for sensor calibration, because the pedestrian head observations are typically within the nominal range of ceiling-mounted 3D range sensors and thus do not suffer from distortion. Furthermore, because pedestrians move over time, it is easy to identify them as shared observations between sensors, and it is possible to generate shared observations even for sensor pairs which have a small overlap. Finally, by using passive detection of natural pedestrian motion as the input to the system, it is possible to perform calibration without being invasive to the environment.

Regarding the use of 2D and 3D sensors together, pedestrian observations are ideal for use as shared observations between the different types of sensors, as the systems are already designed to detect pedestrians. Furthermore, we will show that although the 3D sensors suffer from a low degree of overlap, the addition of 2D sensors, which have a much wider coverage area, can greatly improve the number of shared observations, resulting in a higher level of calibration accuracy.

III. Tracking Infrastructure

To provide a context for the proposed calibration techniques, we will first present some details of the tracking infrastructure.

A. Sensing System Architecture

An experiment room in our laboratory was used for all evaluations in this study. It was instrumented with 28 Asus Xtion PRO LIVE 3D range sensors mounted in rows on the ceiling at a height of 2.6 m, and 6 Hokuyo UTM LRF's mounted at a height of 86 cm, covering a tracking area 13 m long by 7.5 m wide. Fig. 2 shows the overall room layout.

Data from the 3D sensors was captured by 8 desktop PC's,

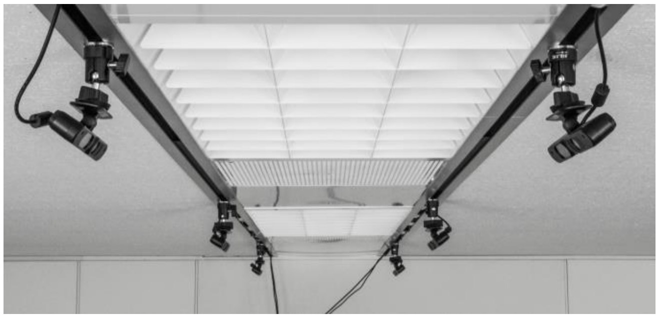

Figure 3. Arrangement of ceiling-mounted 3D range sensors (Asus). Sensors are inverted and mounted at an angle to maximize the coverage of the tracking system.

with up to four sensors connected to each PC. Data from the 2D sensors was captured by Asus Eee PC netbooks, with one PC dedicated to each sensor.

All data was streamed over a wired network to a Core i5 PC running tracking software written in Java. The calibration software was also written in Java and run on a Core i7 PC. For purposes of documentation and repeatability of results, offline processing was used for this study, but in practice we also use the software online with live data to calibrate our sensor environments.

B. *Sensor Arrangement*

1) *3D range sensors*

As Fig. 2 shows, we typically arrange the 3D range sensors upside-down in rows facing in alternate directions, as can be seen in Fig 3. Sensors are placed in order to minimize interference and maximize coverage. Figure 4 shows the overlap between these coverage areas when the sensors are arranged to cover an entire room. The small degree of overlap makes calibration difficult, as there are only a few shared observations between adjacent sensors.

These sensors are not used for full-body skeleton tracking, but rather for detecting the tops of people's heads. Details of the head detection algorithm can be found in [18]. To attain optimal coverage of the head plane, the sensors are adjusted by hand to aim at an angle of approximately 30-60 degrees from the horizontal, with the precise angle chosen to fit the particular room and sensor configuration. If this angle is too shallow (near the horizontal), the sensor will not be able to see the tops of people's heads, but if it is too steep, the effective sensing area will be very small.

2) *2D laser range finders*

The 2D laser range finders are mounted atop metal poles at a height of 86 cm, as shown in Fig. 1 (left). This height was chosen for optimal visibility - the waist is a larger target than the legs and more easily resolved at greater distances. The sensors are rigidly mounted so that their scans cover a horizontal plane, making pitch and roll effectively fixed at zero, as long as the sensors are placed on a level floor.

The sensor poles are portable, so their (x, y) position and yaw angle ϕ must be calibrated each time they are manually placed in a new position.

C. *Tracking Algorithm*

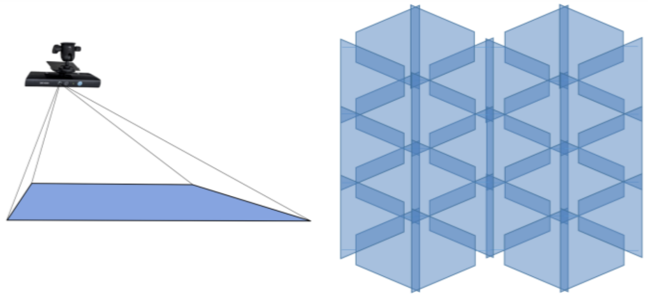

Figure 4. Left: The viewable area at head height for each 3D sensor is approximately trapezoidal. Right: Approximate coverage map for the 28 sensors in our sensor room, showing the regions of overlap between sensors.

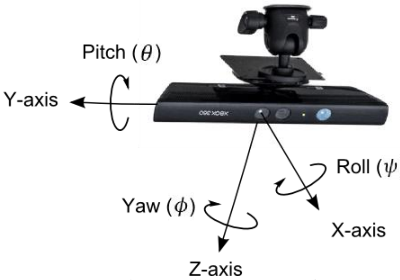

Figure 5. Sensor-centric coordinate system for 3D sensors.

The tracking algorithm used in this work combines the 2D tracking technique presented in [19] and the 3D tracking technique presented in [18]. For details of the tracking algorithms please refer to the original publications.

Both systems use a background subtraction technique to segment foreground data. From this data the top of the head (for 3D sensors) or the estimated body center at waist level (for 2D sensors) are extracted. Detections from multiple sensors are then combined using a particle filter to track each pedestrian.

Variants of this tracking system have been used in many environments, including shopping centers, an elementary school, and several laboratory settings, and the 2D version of the tracking system is currently available as a commercial product[1].

IV. Calibration with 3D Sensors

In this section, we will present our technique for 6-DOF calibration of 3D sensors based on pedestrian observations.

A. *Overview*

The problem we are concerned with here is that of calibrating a set of 3D sensors within 6 degrees of freedom: namely, spatial position (x, y, z), and angles ϕ (yaw), θ (pitch), and ψ (roll), as illustrated in Fig. 5.

We solve this problem in two steps as follows: first, we perform *pitch-roll-z calibration* for each of the 3D sensors individually, using the level plane formed by pedestrian head detections to find each sensor's θ, ψ, and z parameters, presented in Sec. IV-B.

The next step is to find a planar transformation which solves for x, y, and ϕ for all of the sensors. This *relative*

[1] The system is sold by ATR-Promotions under the name ATRacker. http://www.atr-p.com/products/HumanTracker.html (Japanese only)

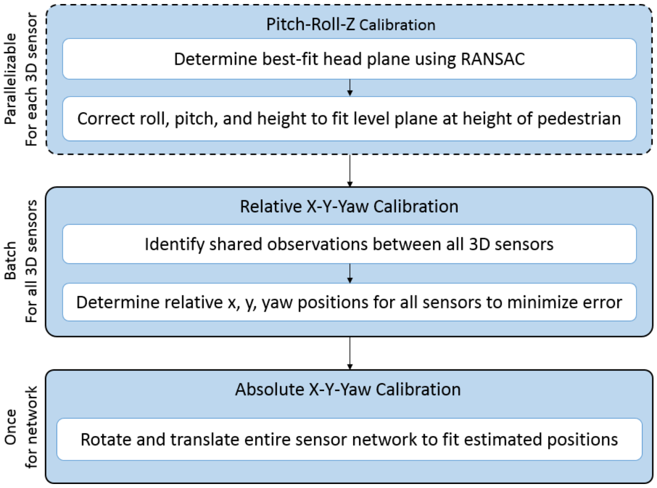
Parallelizable
For each 3D sensor
Pitch-Roll-Z Calibration
Determine best-fit head plane using RANSAC
Correct roll, pitch, and height to fit level plane at height of pedestrian
Batch
For all 3D sensors
Relative X-Y-Yaw Calibration
Identify shared observations between all 3D sensors
Determine relative x, y, yaw positions for all sensors to minimize error
Once
for network
Absolute X-Y-Yaw Calibration
Rotate and translate entire sensor network to fit estimated positions

Figure 6. Overview of calibration procedure. Dashed borders indicate process steps which can be conducted independently per sensor and thus parallelized.

x-y-yaw calibration, presented in Sec. IV-C, is achieved by identifying shared observations between the sensors and calculating the transformations which minimize the error between shared observations. Finally, the entire sensor network is aligned to a global coordinate system through *absolute x-y-yaw calibration*, described in Sec. IV-D. This procedure is summarized in Fig. 6.

B. *Pitch-Roll-Z Calibration*

To compute θ, ψ, and z for each of the 3D sensors, we record detections of a pedestrian walking in the space, and we use the fact that the observations of the top of the pedestrian's head will approximate a level plane – we assume the high frequency vertical head motion while walking will average out over time. Furthermore, if the pedestrian's height is also known, then we can specify the height of the plane.

It should be noted that our technique for detecting the top of a person's head, details of which are explained in [18], requires an *a priori* estimate of the sensor's pose, in particular its pitch. In practice, we do not consider this to be a problem, as approximate sensor poses can easily be estimated by eye. Approximate positions can be used as long as the head direction is up and the sensor is not too tilted in ψ. More tilting will give larger errors, as the top of the head will not be extracted correctly.

To begin the calibration procedure, we fit a plane to the head observations, as shown in Fig. 7, using the random sample consensus (RANSAC) technique [27] to find the best fit while ignoring outliers caused by poor reflectivity or mistaken detections.

Once the plane has been defined, we can determine the rotation angles in θ and ψ necessary to align it with a level ground plane. As illustrated in Fig. 8, the composite rotation angle Ω_r can be determined as the dot product of the normal vector $\overrightarrow{\boldsymbol{n_s}}$ from the head plane and $\overrightarrow{\boldsymbol{n_z}}$ from the level ground-truth plane. The rotation Ω_r can then be achieved by a composition of rotations θ_r and ψ_r.

After applying these rotations, each sensor's coordinate frame is level with respect to the ground plane. We next translate the head plane along the z-axis until its height matches the known height of the pedestrian. In this way, the parameters (θ, ψ, z) are defined for each of the 3D sensors. All head detections are then re-projected according to the updated pose of each sensor.

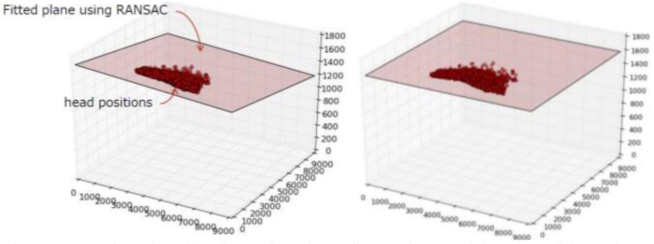

Figure 7. Left: Visualization of a plane fit to observed head positions for a misaligned sensor. Right: Plane fit after correction of sensor pitch, roll, and height (θ, ψ, z). All units are in mm.

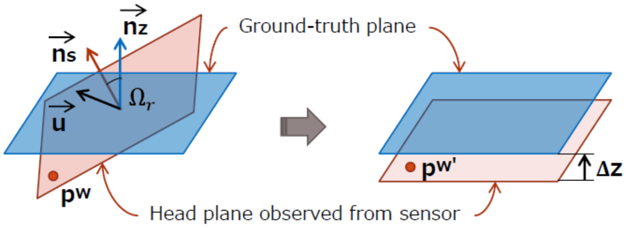

Figure 8. A transformation in pitch and roll is applied to the sensor so that the head plane coincides with the level ground-truth plane. The plane is then translated by a distance Δz so that its height matches the pedestrian's height.

C. *Relative X-Y-Yaw Calibration*

After the θ, ψ, and z parameters have been corrected such that all sensors are coplanar, we next translate and rotate all sensors relative to each other in the shared 2D plane to minimize the error between their shared observations.

1) *Procedure*

Figure 9 shows how a series of pedestrian positions (head detections) observed in time by two sensors can be used for calibration. First, correspondences are made between detections by separate sensors based on time of detection – in the figure, the numbers 1-5 represent time stamps (Fig. 9, left). The sensors are then rotated and translated to minimize the 2D Euclidean distance between those corresponding detections (Fig. 9, right).

To accommodate for small amounts of clock drift, we consider human detections from different sensors to be simultaneous shared observations if they occur within a given time threshold of each other (we used 20ms).

Note that in this work, we assume only one pedestrian to be visible at any time; however, various techniques are available to extend this to a multiple-pedestrian scenario. For example, individual pedestrians observed by different sensors can be associated based on trajectory shape, velocity profile, and membership in social groups [22, 28].

2) *Mathematical formulation*

The basic mathematical problem is to find a rigid 2D transformation matrix similar in some ways to the problem of "bundle adjustment" in computer vision [29]. However, there

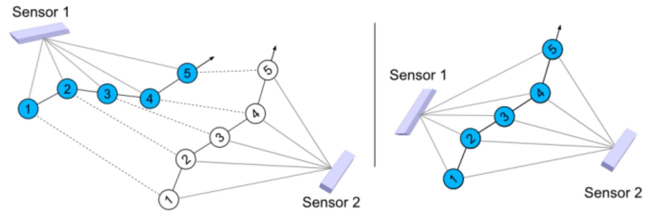

Figure 9. Example of calibration using shared observations. Left: Two uncalibrated sensors simultaneously observe the path of a person at times 1-5. Right: Sensor positions are adjusted to minimize the error between corresponding shared observations.

are a few key differences. Whereas bundle adjustment deals with 2D projections of 3D points on a camera image, the current problem deals with direct 3D measurements of the real spatial positions of the detected points.

More formally, consider a sensor network consisting of S sensors, in which human positions are observed at each time step $t \in \{1..\tau\}$ by zero or more sensors. The 2D pose of each sensor of index $i \in \{1..S\}$ is parameterized by a vector $\boldsymbol{s}_i = (x_i, y_i, \phi_i)$ and represented collectively as the parameter vector $\boldsymbol{\beta} = \{\boldsymbol{s_1}, \boldsymbol{s_2}, \dots, \boldsymbol{s_S}\}$. The observation of the human position at time t from sensor i is parameterized in sensor-relative 2D coordinates by a vector $\boldsymbol{p}_t^{(i)} = \left(x_t^{(i)}, y_t^{(i)}\right)$, and its positions can be transformed into global coordinates by a rigid transformation matrix $\boldsymbol{T_i}(\boldsymbol{s}_i)$, which for generality can be written $\boldsymbol{T_i}(\boldsymbol{\beta})$**.** Finally, let v_{it} denote a binary variable equaling 1 if sensor i can see the human at time t and 0 otherwise.

The procedure for finding the optimal sensor positions is thus represented by Eq. 1, that is, for each pair of sensors at each time step, we minimize the Euclidean distance between the 2D projections of their simultaneous observations of the pedestrian into real space by adjusting the sensor poses. We solve for the optimal parameter vector $\boldsymbol{\beta}$ by least-squares minimization, using the Levenberg-Marquardt method.

$$\underset{\boldsymbol{\beta}}{min} \sum_{i=1}^{S} \sum_{j=1}^{S} \sum_{t=1}^{\tau} v_{it} v_{jt} \left\| \boldsymbol{T}_i(\boldsymbol{\beta}) \boldsymbol{p}_t^{(i)} - \boldsymbol{T}_j(\boldsymbol{\beta}) \boldsymbol{p}_t^{(j)} \right\| \tag{1}$$

Note that since no absolute position reference is included in this calculation, the problem is underconstrained, and an infinite number of valid solutions are possible. As we are only concerned at this point with relative sensor positions, we can choose to set the first sensor as fixed at (0,0,0) and solve for the remaining $S - 1$ sensor positions.

As a final note, if large numbers of shared observations are available, random sampling can be used to reduce the size of the matrix to be solved. We tend to employ this practice when the data contains more than 10,000 shared observations.

D. Absolute X-Y-Yaw Calibration

Once the relative poses of the sensors have been determined, the final step in the proposed algorithm is to align the sensor network with a known coordinate system. Since the sensors need to be manually mounted on the ceiling for the purpose of tracking, we can expect that approximate positions of some or all the sensors are known *a priori*.

To obtain the best alignment of the calibrated sensor network to the external coordinate frame, we perform another least-squares minimization, using the sensor positions only.

In this calculation we only need to consider the 2D positions of the sensors, not their orientations (which are fixed relative to the rest of the network). We define a network of S sensors, where the 2D position of each sensor of index $i \in \{1..S\}$ is parameterized by a vector $\boldsymbol{s}_i = (x_i, y_i)$ in the network-relative frame. We then consider a set of vectors $\hat{\boldsymbol{s}}_i = (\hat{x}_i, \hat{y}_i)$ representing the *a priori* estimate of the position of sensor i in the global coordinate frame. We define a parameter vector $\boldsymbol{\beta} = (x_G, y_G, \phi_G)$ to represent the global transformation parameters to be applied to the sensor network, and let matrix $\boldsymbol{T}_G(\boldsymbol{\beta})$ define the rigid transformation between the network-relative frame and the global coordinate frame. In case position estimates for some sensors are not available, we define a binary variable v_i equaling 1 if the absolute sensor position estimate $\hat{\boldsymbol{s}}_i$ is defined and 0 if it is not.

We then perform the procedure represented in Eq. 2, finding the optimal global transformation which minimizes the error between the calibrated sensor positions and the *a priori* estimated sensor positions. Sensors with no *a priori* estimate are not used as constraints, but are translated and rotated with the rest of the network.

$$\underset{\boldsymbol{\beta}}{min} \sum_{i=1}^{S} v_i \| \boldsymbol{T}_G(\boldsymbol{\beta}) \boldsymbol{s}_i - \hat{\boldsymbol{s}}_i \| \tag{2}$$

We have also developed a graphical user interface allowing manual alignment. The sensor positions are projected on a map of the tracking area, and a user can drag and rotate the entire sensor network by hand to register it with the absolute coordinate frame. In practice, we find manual alignment most useful when the sensors are not rigidly fixed and we are trying out different configurations, whereas once sensors have been installed, the mathematical technique is more convenient.

V. Calibration using 2D and 3D sensors

The techniques presented so far can be used with 3D range sensors alone, and as of the time of writing, they have successfully been used by us for calibrating sensors in seven different experimental environments.

However, depending on the sensor configuration, it can sometimes be quite difficult to obtain shared observations between some sensors. This can lead to low calibration accuracy, or even make it impossible to calibrate the network.

To remedy this problem, we propose the use of 2D laser range finders in conjunction with the 3D sensor network. The greater sensing range of the 2D LRF's can ensure a much larger degree of overlap between sensors. As Fig. 10 shows, a single 2D sensor can track a person over a very large area, while the tracking area of the 3D sensors is quite limited.

As Fig. 11 (left) shows, the connectivity of 3D sensor

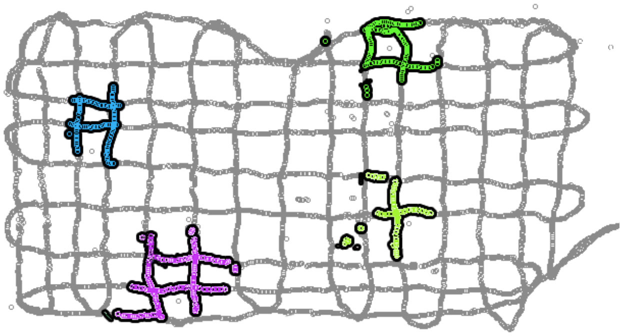

Figure 10. Detections of a pedestrian walking in a grid pattern in our sensing room. Gray circles show detections from a 2D LRF, and groups of colored circles outlined in black show detections of the same pedestrian from four 3D sensors.

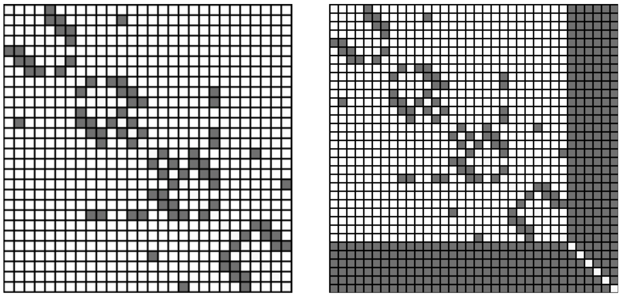

Figure 11. Illustration of network connectivity. Gray squares indicate the existence of shared observations between two sensors. The matrix on the left shows 3D sensors only, while the matrix on the right also includes 2D sensors which, in this case, each share observations with all of the 3D sensors.

network in terms of shared observations can be sparse. The 28-sensor network shown in the diagram has four major groups of highly-interconnected sensors, but very few connections between these groups. Fig. 11 (right) shows how connectivity is dramatically increased by including the 2D sensors. This technique can even make it possible to calibrate sets of 3D sensors sharing no overlap with each other.

1) Technique

In order to calibrate 2D and 3D sensors on the same coordinate frame, we model the head detections from the 3D sensors as being directly above the waist detections from the 2D sensors. In this way, the human detections from all sensors can be projected onto a shared 2D plane.

The problem thus becomes one of calibrating a set of 3D sensors in 6 DOF $(x, y, z, \phi, \theta, \psi)$ together with a set of 2D sensors with only 3 DOF (x, y, ϕ), since we consider the 2D sensors to be level and fixed at a given height.

The simplest way to perform calibration using the two kinds of sensors is to add the 2D sensors to the network in the relative x-y-yaw calibration step. This does work, but doing so greatly increases the size of the matrix to be optimized, and thus slows down the computation.

To perform the calibration more efficiently, we consider that when calibrating 3D sensors alone, only a small fraction of the detections from any given sensor constitute shared observations with other sensors, and the rest of the data are not useful for calibration.

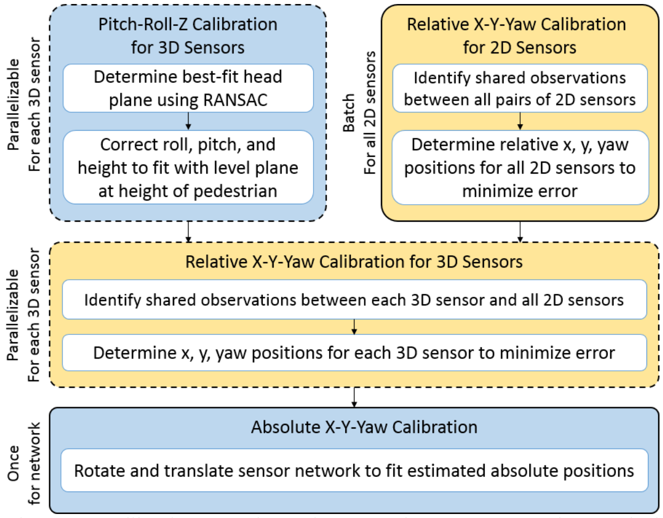
Parallelizable
For each 3D sensor
Pitch-Roll-Z Calibration for 3D Sensors
Determine best-fit head plane using RANSAC
Correct roll, pitch, and height to fit with level plane at height of pedestrian
Batch
For all 2D sensors
Relative X-Y-Yaw Calibration for 2D Sensors
Identify shared observations between all pairs of 2D sensors
Determine relative x, y, yaw positions for all 2D sensors to minimize error
Parallelizable
For each 3D sensor
Relative X-Y-Yaw Calibration for 3D Sensors
Identify shared observations between each 3D sensor and all 2D sensors
Determine x, y, yaw positions for each 3D sensor to minimize error
Once
for network
Absolute X-Y-Yaw Calibration
Rotate and translate sensor network to fit estimated absolute positions

Figure 12. Calibration procedure using 2D and 3D sensors together. Dashed borders indicate process steps which can be conducted independently per sensor and thus parallelized. Yellow highlighting indicates steps that differ from the 3D-only calibration procedure.

However, observations of the 2D sensors overlap with nearly 100% of the data from the 3D sensors. As a result, the small number of shared observations between adjacent 3D sensors makes only a minor contribution, and the 3D sensors can be calibrated directly to the 2D sensors.

Thus, the relative x-y-yaw calibration for the 3D sensors can be parallelized, and we organize the process as shown in Fig. 12. First, in parallel with the pitch-roll-z calibration of the 3D sensors, the set of 2D sensors are calibrated with each other. Next, each of the 3D sensors are independently adjusted such that their observations align with the observations from the calibrated 2D sensors. Finally, the entire network is aligned with the global coordinate system.

VI. Evaluation

A. Evaluation Procedure

We evaluated the performance of our two proposed techniques in two ways: first, we measured the accuracy of estimated sensor positions against manually-measured ground truth, and second, we measured the absolute tracking accuracy of pedestrians at known locations. For each of these measurements, we evaluated both the proposed calibration technique using 3D sensors only, and the second technique using both 3D and 2D sensors. This evaluation was conducted in the environment described in Sec. III-A.

1) Sensor position accuracy

To obtain ground truth of the sensor positions, we used a Bosch DLE 150 precision laser range measurement device to measure the location of each sensor in an (x, y, z) coordinate system relative to the walls of the room. We then recorded 2D and 3D range data from one pedestrian walking in the room for 300 seconds, and we generated sets of estimated sensor positions using each of the two proposed techniques. For each set of sensor positions, the error between the estimated and ground-truth positions of each sensor was computed.

2) *Pedestrian Tracking Accuracy*

We then evaluated the accuracy of pedestrian tracking by marking 18 reference points at 2m intervals on a precisely-measured 10m x 4m grid on the floor and asking participants to stand over those points. We repeated each measurement four times, with the participant facing in a different cardinal direction each time. The mean position over two seconds of data was recorded for each reference point.

In total, five participants were measured four times each, facing in different directions each time, yielding 20 measurements at each of the 18 reference points, for a total of 360 measurements overall.

B. Results

The results of these two evaluations are shown in Table I. For both evaluations, we computed the root-mean-squared (RMS) error in x, y, and z directions, as well as total RMS error in 3D. Additionally, we calculated the 2D error in the x-y plane, because we consider this to be a more relevant error metric for 2D tracking applications.

The sensor position results (considering 2D RMS error) show a sensor position accuracy of 130 mm with the 3D-only technique and 52 mm with the 3D+2D technique. Tracking accuracy shows a slightly higher error, at 159mm with 3D-only and 104mm with 3D+2D.

The results for z error were similar across both techniques, because the same process of pitch-roll-z calibration was used. Along the z-axis, sensor positions were calibrated within 17mm, and tracking was accurate to within 71mm.

C. Discussion

Results showed the tracking error to be larger than the sensor position error. This is partly due to the fact that the human figure is an irregular shape, with no clear center. Even the top of the head is not well-defined, for example, if the head is inclined. Reflectivity errors from hair and clothing also contribute to uncertainty in defining the human shape.

Similarly, the difficulty of detecting the exact top of the head leads to z errors in real-time tracking, whereas during sensor calibration the RANSAC plane fit is successfully able to reject many such misdetections as outliers, correctly determining the sensor z positions to high precision.

Finally, the data demonstrate that greater accuracy is obtained via the addition of 2D sensors during the calibration process. The 3D+2D technique provided a 60% improvement in sensor position accuracy and a 35% improvement in tracking accuracy over the 3D-only technique.

VII. Future Work and conclusions

A. Future Work

1) *Multiple Anonymous Pedestrians*

Ideally, it would be desirable to perform calibration based on multiple, anonymous pedestrians, such as customers in a shopping mall. If we fix the z position of the sensors based on measured ceiling height rather than using the pedestrian's height for z calibration (a reasonable choice, given that the sensors are typically mounted on a ceiling of fixed and known height), then calibration can be performed using anonymous pedestrians. Pitch and roll can still be calibrated based on the head plane, even if z is unknown.

Table I. Evaluation results

	3D Only	**3D + 2D**
Sensor position accuracy		
x Error	109 mm	35 mm
y Error	71 mm	39 mm
z Error	16 mm	17 mm
Total Error (3D)	131 mm	55 mm
Total Error (2D Projection)	130 mm	52 mm
Pedestrian tracking accuracy		
x Error	137 mm	80 mm
y Error	80 mm	66 mm
z Error	71 mm	71 mm
Total Error (3D)	174 mm	126 mm
Total Error (2D Projection)	159 mm	104 mm

Furthermore, in a previous study we have presented techniques for using multiple pedestrians to calibrate 2D sensor systems, including techniques for identifying shared observations among large numbers of pedestrians in public spaces [22]. In future work, we hope to integrate such methods with the techniques proposed here, enabling 3D sensor networks to be calibrated using data passively collected from pedestrians naturally walking through the space being monitored.

2) *Minimum Required Data*

One question that remains to be resolved is how to determine the minimum amount of data required for calibration. Currently, we have a reference pedestrian walk in a grid pattern through the room to ensure sufficient data, but it would be helpful to develop an automatic way of identifying when the system is ready for calibration, and specifying which areas need more data. For plane-fitting, it is necessary to have a sufficient 2-dimensional spread of points, as a single straight-line trajectory does not uniquely define a plane, and for inter-sensor calibration, it is important to have shared observations with adjacent sensors, meaning the pedestrian needs to walk through all overlapping sensing areas. Currently, the decision of when to calibrate is manual, and operators of the system are given an option to collect more data if there is a failure in plane extraction or insufficient connectivity for network calibration.

3) *Zero-knowledge Calibration*

As mentioned earlier, the technique we have presented requires rough initial estimates of pitch and roll for each sensor, in order to detect the tops of people's heads. While these estimates can usually be obtained easily in practice, it would be more elegant to create a calibration algorithm that could calibrate the entire network starting from an entirely zero-knowledge state. In future work we hope to develop techniques to make this possible.

4) *Hybrid tracking networks*

In this study, we mainly considered the 2D sensors to be a temporary aid for the calibration of the 3D sensor network, but it is also reasonable to consider making the 2D sensors a permanent part of the system. The strengths and weaknesses of each sensor type could complement each other, resulting in more robust tracking overall. Furthermore, the additional information gained from using two kinds of sensors could help in tasks such as recognizing gestures or activities. We consider this to be an exciting area for future work.

B. Conclusions

We have developed a set of techniques enabling precise 6-DOF extrinsic calibration of large numbers of ceiling-mounted 3D range sensors based solely on observations of moving pedestrians. This is an important achievement for enabling the practical deployment of sensor networks in active public spaces, where traditional point-cloud matching approaches cannot be used, and where it is undesirable to disturb the social environment for the purpose of sensor calibration.

In this work, we have introduced techniques for calibrating pitch and roll of each sensor based on mapping pedestrian head detections to a level plane, and for determining relative sensor positions based on shared observations of pedestrians. Furthermore, we have shown the benefit of using 2D sensors to assist in the calibration of 3D sensors, resulting in measurable improvements in both sensor position estimation and pedestrian tracking performance. Precise calibration can lead to more accurate pedestrian tracking, an important consideration for robots operating in proximity to humans.

ACKNOWLEDGMENT

We would like to thank Florent Ferreri and Kuanhao Zheng for their contributions to the development of this system.

REFERENCES

[1] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers, "People Tracking with Mobile Robots Using Sample-Based Joint Probabilistic Data Association Filters," The International Journal of Robotics Research, vol. 22, pp. 99-116, February 1, 2003 2003.

[2] P. Trautman and A. Krause, "Unfreezing the robot: Navigation in dense, interacting crowds," in Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, 2010, pp. 797-803.

[3] T. Kanda, D. F. Glas, M. Shiomi, and N. Hagita, "Abstracting People's Trajectories for Social Robots to Proactively Approach Customers," Robotics, IEEE Transactions on, vol. 25, pp. 1382-1396, 2009.

[4] S. Satake, T. Kanda, D. F. Glas, M. Imai, H. Ishiguro, and N. Hagita, "A Robot that Approaches Pedestrians," IEEE Trans. Robotics, 2012.

[5] K. Hayashi, M. Shiomi, T. Kanda, and N. Hagita, "Friendly Patrolling: A Model of Natural Encounters," in Robotics: Science and Systems VII, Los Angeles, CA, 2012.

[6] C. Shi, M. Shiomi, C. Smith, T. Kanda, and H. Ishiguro, "A Model of Distributional Handing Interaction for a Mobile Robot," in Robotics: Science and Systems, 2013.

[7] G. Balistreri, S. Nishio, R. Sorbello, and H. Ishiguro, "Integrating Built-in Sensors of an Android with Sensors Embedded in the Environment for Studying a More Natural Human-Robot Interaction," in AI*IA 2011: Artificial Intelligence Around Man and Beyond. vol. 6934, R. Pirrone and F. Sorbello, Eds., ed: Springer Berlin Heidelberg, 2011, pp. 432-437.

[8] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," Computer vision and image understanding, vol. 104, pp. 90-126, 2006.

[9] T. Teixeira, G. Dublon, and A. Savvides, "A survey of human-sensing: Methods for detecting presence, count, location, track, and identity."

[10] J. Cui, H. Zha, H. Zhao, and R. Shibasaki, "Laser-based detection and tracking of multiple people in crowds," Comput. Vis. Image Underst., vol. 106, pp. 300-312, 2007.

[11] A. Panangadan, M. Mataric, and G. Sukhatme, "Detecting anomalous human interactions using laser range-finders," in Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on, 2004, pp. 2136-2141 vol.3.

[12] A. Fod, A. Howard, and M. A. J. Mataric, "A laser-based people tracker," in Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on, 2002, pp. 3024-3029.

[13] H. Zhao and R. Shibasaki, "A novel system for tracking pedestrians using multiple single-row laser-range scanners," Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, vol. 35, pp. 283-291, 2005.

[14] C. Wongun, C. Pantofaru, and S. Savarese, "Detecting and tracking people using an RGB-D camera via multiple detector fusion," in Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, 2011, pp. 1076-1083.

[15] A. Bevilacqua, L. Di Stefano, and P. Azzari, "People Tracking Using a Time-of-Flight Depth Sensor," in Video and Signal Based Surveillance, 2006. AVSS '06. IEEE International Conference on, 2006, pp. 89-89.

[16] D. W. Hansen, M. S. Hansen, M. Kirschmeyer, R. Larsen, and D. Silvestre, "Cluster tracking with Time-of-Flight cameras," in Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on, 2008, pp. 1-6.

[17] E. J. Almazan and G. A. Jones, "Tracking People across Multiple Non-overlapping RGB-D Sensors," in Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on, 2013, pp. 831-837.

[18] D. Brščić, T. Kanda, T. Ikeda, and T. Miyashita, "Person Tracking in Large Public Spaces Using 3-D Range Sensors," Human-Machine Systems, IEEE Transactions on, vol. 43, pp. 522-534, 2013.

[19] D. F. Glas, T. Miyashita, H. Ishiguro, and N. Hagita, "Laser-Based Tracking of Human Position and Orientation Using Parametric Shape Modeling," Advanced Robotics, vol. 23, pp. 405-428, 2009.

[20] A. W. Senior, A. Hampapur, and M. Lu, "Acquiring multi-scale images by pan-tilt-zoom control and automatic multi-camera calibration," in Application of Computer Vision, 2005. WACV/MOTIONS'05 Volume 1. Seventh IEEE Workshops on, 2005, pp. 433-438.

[21] J. Hightower and G. Borriello, "Location systems for ubiquitous computing," Computer, vol. 34, pp. 57-66, 2001.

[22] D. F. Glas, F. Ferreri, T. Miyashita, H. Ishiguro, and N. Hagita, "Automatic calibration of laser range finder positions for pedestrian tracking based on social group detections," Advanced Robotics, 2012.

[23] K. Schenk, A. Kolarow, M. Eisenbach, K. Debes, and H.-M. Gross, "Automatic calibration of a stationary network of laser range finders by matching movement trajectories," in Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, 2012, pp. 431-437.

[24] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D point cloud based object maps for household environments," Robotics and Autonomous Systems, vol. 56, pp. 927-941, 2008.

[25] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," Sensors, vol. 12, pp. 1437-1454, 2012.

[26] B. K. Horn, "Closed-form solution of absolute orientation using unit quaternions," JOSA A, vol. 4, pp. 629-642, 1987.

[27] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," Commun. ACM, vol. 24, pp. 381-395, 1981.

[28] D. F. Glas, T. Miyashita, H. Ishiguro, and N. Hagita, "Automatic position calibration and sensor displacement detection for networks of laser range finders for human tracking," in Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on, 2010, pp. 2938-2945.

[29] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment—a modern synthesis," in Vision algorithms: theory and practice, ed: Springer, 2000, pp. 298-372.