# Person Tracking and Following with 2D Laser Scanners

Angus Leigh[1], Joelle Pineau[1], Nicolas Olmedo[2], and Hong Zhang[2]

*Abstract*— **Having accurate knowledge of the positions of people around a robot provides rich, objective and quantitative data that can be highly useful for a wide range of tasks, including autonomous person following. The primary objective of this research is to promote the development of robust, repeatable and transferable software for robots that can automatically detect, track and follow people in their environment. The work is strongly motivated by the need for such functionality onboard an intelligent power wheelchair robot designed to assist people with mobility impairments. In this paper we propose a new algorithm for robust detection, tracking and following from laser data. We show that the approach is effective in various environments, both indoor and outdoor, and on different robot platforms (the intelligent power wheelchair and a Clearpath Husky). The method has been implemented in the Robot Operating System (ROS) framework and will be publicly released as a ROS package. We also describe and will release several datasets designed to promote the standardized evaluation of similar algorithms.**

## I. INTRODUCTION

Having accurate knowledge of the positions of people over time provides rich, objective and quantitative data that can be highly useful for a wide range of applications. More specifically, the capability to autonomously detect, track and follow a person has been identified as an important functionality for many assistive and service robot systems [1], [2]. Over the last decade, significant progress has been made in developing person detection and tracking algorithms, often with the aim of improving human-robot interaction or robot navigation in populated environments [3], [4].

Yet most of the work to date is not easily transferable to new applications: algorithms are tested on a single robot in a single environment (if at all, sometimes only in simulation under artificial conditions), in many cases the code has not been made publicly available, datasets collected during validation sessions are not shared, and quantitative comparisons to existing algorithms are not performed. Despite our best efforts, we have not been able to replicate many of the published results onboard the SmartWheeler platform [5]. In fact, we have yet to find a robust laser-based person-following system for a mobile robot that has demonstrated capability to work indoors, outdoors and in cluttered and crowded areas.

In this paper we present a novel method for person tracking and following with 2D laser scanners. It builds on recent results in the literature [3], [4], extending them in several directions to improve accuracy and reduce the number of errors. One contribution of this work is a novel tracking method which uses tracking of both legs, rather than an individual leg such as in [3], [4], to improve reliability, especially in cases of self-occlusion. We also integrate local occupancy grid maps to improve data association by disallowing the initiation or continuation of people tracks in occupied space. However, unlike the approach in [6] which constructs occupancy grid maps of all scan points, ours only uses non-human scan points and therefore does not require people to move continuously to be tracked. Our approach also integrates the method of cluster tracking present in Robot Operating System (ROS) leg_detector package in which all detected clusters are tracked in every frame, included non-human clusters, improving data associated in cluttered environments. Finally, we incorporate a closed-loop control algorithm to allow the robot to autonomously follow tracked individuals.

A major asset of our approach is the ability to work on multiple different robots, under various operating conditions. We present extensive empirical results, validating the performance of our approach on two different robot platforms over 40 minutes of data collected across different environments, both indoor and outdoor, with moving and stationary robot, and varied crowd and obstacle conditions. These results show the benefit of our tracking approach compared to the existing open-source implementation in ROS, especially in the application of person following. Another contribution of this work is a public release of all datasets and code used during our investigations as an open-source ROS package, to facilitate ongoing research in this area in future years. These datasets include, to the best of our knowledge, the first laser-based person tracking benchmarks collected on a moving robot.

While the work we report here focuses specifically on the problem of accurate tracking and following of people by social and assistive robots, we expect our work to have significant applications outside of this field, including for security (e.g. tracking intruders), entertainment (e.g. developing interacting exhibits), marketing (e.g. location-aware personalized advertisement), rehabilitation (e.g. assessment of patients' locomotion patterns following an injury [7]), and beyond.

## II. PROBLEM DESCRIPTION

The problem of interest can be decomposed into three sub-problems: (1) detection, (2) tracking, and (3) following. For reasons of modularity and robustness, these problems are tackled with three separate (though connected) modules.

[1] Angus Leigh and Joelle Pineau are with the School of Computer Science at McGill University, QC, Canada {angus.leigh, jpineau}@cs.mcgill.ca

[2]Nicolas Olmedo and Hong Zhang are with Mechanical Engineering and Computer Science departments at the University of Alberta {olmedo, hzhang}@ualberta.ca

Such modularity is consistent with most of the literature on the topic, and allows development of a solution that is transferable to a large range of applications. For example, a security robot may require only detection and tracking; a social robot on the other hand may ask the user to *manually* specify what person to follow (e.g. from an image, or with a gesture command), and use autonomous behavior only for tracking and following.

For the purposes of this work, we limit our attention to systems with planar laser sensors. Such sensors are widely available on autonomous robots, especially those deployed in public and urban environments, due to their reliability and accuracy for mapping and navigation tasks. Laser sensing is also computationally cheap to process, lighting invariant, and functional under diverse operating conditions. Further, the wide field of view of modern laser scanners allows the robot to follow in close proximity with less risk of losing people out of frame.

We aim to build and test a system that can achieve person tracking and following under diverse conditions: different robot platforms, indoors and outdoors, single or multiple people, cluttered with stationary or moving obstacles, without an *a priori* map of the environment. In support of this, we deploy and validate our system on two different robot platforms, each in a different environment.

The SmartWheeler robot, shown in Figure 1, is an intelligent powered wheelchair, designed in collaboration with engineers and rehabilitation clinicians, aimed at assisting individuals with mobility impairments. The robot is built upon a commercial power wheelchair base, to which we have added onboard computing, three Hokoyu UHG-08LX laser scanners, several sonars, an RGB-D camera, on-wheel odometry, and a touchscreen for two-way interaction. One of the important tasks for the robot is to automate navigation in challenging environments (crowded rooms, narrow spaces) to reduce physical and cognitive load on the wheelchair user [2]. The task of moving around with another individual, whether a friend or caregiver, is also one that requires substantial concentration, and wheelchair users often comment that it is difficult for them to simultaneously control their wheelchair (presumably via joystick) and hold a conversation, thus having the ability to use autonomous navigation capabilities for walking side-by-side with (or behind) another person is highly desirable [2].

A Husky A200 from Clearpath Robotics (shown in Fig. 2) is used for the outdoor experiments in this work. This robot has been used for mapping, localization, route following and autonomous navigation on rugged terrain, such as sand, gravel and grass. An on-board computer is used to interface low-level controllers and sensors, as well as to process visual and range measurements. Typically, an additional computer is mounted on top of the platform for applications with high computational requirements. A Hokuyo URG-04LX-UG01 laser sensor, mounted on the front of the robot, is used to collect the 2D scans used in this study.

Fig. 1. The SmartWheeler robot.

## III. Related Work

Most previous work focuses on only one or two of the three identified sub-problems: detection, tracking and following. Few papers present integrated systems tackling all three components. We focus in particular on work that uses depth sensors, such as laser or RGB-D, as those are the sensors available on our SmartWheeler robot, as well as on numerous other assistive robots.

Montemerlo et al. [8] were possibly the first to present a method for automatically locating and tracking people from laser data. One drawback of this work is the necessity for an *a priori* occupancy grid map of the operational environment, which is used for background subtraction to detect the person. The tracking is achieved via a conditional particle filter.

Schulz et al. [6] proposed to estimate the number of people in the current scan based on the number of moving local minima in the scan. Unfortunately, this requires people move continuously to be tracked, and is susceptible to poor results in cluttered environments (where the number of local minima is misleading). They also introduced a Sample Based Joint Probabilistic Data Association Filter (SJPDAF) over the observed local minima to improve tracking reliability.

Topp et al. [9] extended [6], by picking out shapes of legs and person-wide blobs in laser scans using hand-coded heuristics, to allow detection and tracking of both stationary and moving people. The approach was also combined with a person following navigation algorithm, combining both the tracked person's position as well as the location of nearby obstacles to determine suitable control. Gockley et al. [10] used a similar approach, with a few modifications, including using a Brownian motion model for the tracking component. This approach was further extended by Hemachandra [11], which improved the person-following component by proposing a navigation approach that accounts for personal space, while avoiding obstacles. Unfortunately these approaches cite tracking difficulties in cluttered conditions, since they relied primarily on detecting clusters of a heuristically-determined size in the laser scan.

More recently, Arras et al. [3] reduced this limitation by proposing a method that detects legs by first clustering scan points and then using supervised learning to learn shapes of leg clusters. Detected legs are tracked over time using

constant-velocity Kalman filters and a multiple-hypothesis tracking (MHT) data association technique. This approach benefits from its ability to maintain (but not initiate) tracks of stationary people, and does not require an *a priori* occupancy grid map of the environment. Initial results for this method appear promising, but demonstrations on walking-speed robots in cluttered and crowded areas have yet to be performed. Thus, many questions remain about the robustness and generalizability of the approach.

Finally, Lu et al. [4] extended an existing ROS package originally developed at Willow Garage, that had not been formally published.[1] This method first matches scan clusters using nearest-neighbour (NN) data association, then determines which clusters are human legs using a supervised learning approach similar to [3]. In the data association literature, it is generally agreed that the NN filter is outperformed by more sophisticated methods, such as the global nearest-neighbour (GNN) filter, SJPDAF or MHT [12]. Furthermore, the data association method only considers absolute Euclidian distances while ignoring valuable uncertainty covariances that can improve tracking.

Navarro-Serment et al. [13] present the only published work, which we are aware of, that aims to track people from a mobile robot in hilly, outdoor environments using only 2D laser scanners. The method's applicability to cluttered or crowded environments is questionable since it uses a NN data association method and clusters scan points with a relatively large threshold of $80cm$.

*Person following with other sensors:* Munaro et al. [14] proposed to track people in real-time with an RGB-D sensor on a mobile robot, demonstrating promising results with high update rates running on a CPU only. Similarly, Gritti et al. [15] demonstrated a method for detecting and tracking people's legs on a ground plane from a low-lying viewpoint with the an RGB-D sensor. However the RGB-D sensor's narrow field of view, minimum distance requirement, and inability to cope with sunlight, limit this technique's applications for person-following.

Cosgun et al. [1] presented a novel person following navigation technique which is most similar to a Dynamic Window Approach [16]. The initial detection was acquired via a user indicating the desired person they wish to follow in the an RGB-D sensor's image. Tracking was then achieved via lasers using estimated leg positions. This approach can only track one person, and uses a NN matching over a small number of laser segment features, thus is not robust in situations with multiple individuals.

Kobilarov et al. [17] developed a person-following Segway robot using an omni-directional camera and a laser scanner. Nonetheless, variable environment lighting, backgrounds and appearances of people are factors which can be difficult to control for and can mislead vision-reliant tracking systems.

[1] http://www.ros.org/news/2009/12/person-following-and-detection-in-an-indoor-environment.html

## IV. Joint leg tracker

Our proposed system, which we call the Joint Leg Tracker, includes several components. The autonomous person detection is achieved using clustering over laser detections; confidence levels are also assessed to help prune false positives. The autonomous tracking is achieved using a combination of Kalman filter (for predictive over consecutive scans) and a GNN method (to resolve the scan-to-scan data association problem). Finally, a control algorithm is used to follow the tracked target.

### A. Autonomous person detection

The laser scanner returns a vector of distance measurements taken on a plane, roughly $30cm$ from the ground, at a resolution of 1/3°. Scan points returned are first clustered according to a fixed distance threshold, such that any points within the threshold are grouped together as a cluster. The threshold is chosen to be small enough to often separate a person's two legs into two distinct clusters, but to rarely generate more than two clusters per person. To mitigate noise, clusters containing less than three scan points are discarded in low-noise environments and clusters containing less than five scan points are discarded in high-noise environments. To compensate for egomotion, the position of all detected clusters are transformed to the robot's odometry frame.

Detected clusters are used as observations for the tracker. The observations at timestep $k$ are denoted as $\mathbf{z}_k = \left\{z_k^1, z_k^2, ..., z_k^{M_k}\right\}$ where $M_k$ is the total number of detected clusters at time $k$.

Clusters are further classified as human or non-human, based on a set of geometric features of the clusters. The features are listed in Table I and extend the set proposed in [3] and [4]. The classification is done using a random forest classifier trained on a set of 1700 positive and 4500 negative examples [18]. Positive examples were obtained by setting up the laser scanner in an open area with significant pedestrian traffic; all clusters which lay in the open areas and met the threshold in Sec. IV-A were assumed to be the result of people and were used as positive training samples. Negative examples were obtained by moving the sensor around in an environment devoid of people; all clusters which met the threshold in Sec. IV-A were used as negative training samples.

One benefit of using an ensemble classification method is that a measure of confidence in the classification can be extracted by considering the number of individual predictors predicting each class.[2] Thus, rather than using the binary output of the classifier (human/non-human), we consider the confidence level (from 0-100%) from the classifier and pass this information (along with the cluster location) to the tracking module.

### B. Autonomous tracking of multiple people

*1) Kalman filter tracking:* The position of all detected clusters (regardless of confidence level) are individually

[2] A function for extracting this confidence is included in the random forest classifier implementation for OpenCV [19].

TABLE I

FEATURES USED FOR CONFIDENCE CALCULATION OF SCAN CLUSTERS.

| | | |
|---|---|---|
| Number of points | Width | Length |
| Standard deviation | Avg dist. from median | Occluded (boolean) |
| Linearity | Circularity | Radius of best-fit circle |
| Boundary length | Boundary regularity | Mean curvature |
| Mean angular diff. | Inscribed angular var. | Dist. from laser scanner |

tracked over time using a Kalman filter [20]. We refer to each Kalman filter which tracks a scan cluster over time as a *track* denoted as $\mathbf{x}_k^j$ and the set of all active tracks as $\mathbf{X}_k = \left\{\mathbf{x}_k^1, \mathbf{x}_k^2, ..., \mathbf{x}_k^{N_k}\right\}$ where $N_k$ is the total number of tracks at time $k$. Further, the set of tracks $\mathbf{X}_k$ also includes a single track for each tracked person (their initiation is introduced in Sec. IV-B.4).

The Kalman filter for each track has a state estimate $\mathbf{x}_k^j = \begin{bmatrix} x & y & \dot{x} & \dot{y} \end{bmatrix}^T$ with the position and velocity of the cluster in 2D coordinates. New tracks are initialized with a velocity of zero and existing tracks are updated with a constant velocity motion model. Process noise $\mathbf{w}$ is assumed to be Gaussian white noise with diagonal covariance $\mathbf{Q} = q\mathbf{I}$.

The observation matrix $\mathbf{H}$ includes only the position of the cluster and observation noise has a covariance of $\mathbf{R} = r\mathbf{I}$. In tuning the Kalman filter, it was found that a small observation noise covariance ($r = 0.1^2$) is sufficient because position measurements from the laser scanner are highly accurate.

*2) Global nearest neighbour data association:* Since the system is designed to track all detected scan clusters, uncertainty arises pertaining to how detections $\mathbf{z}_k$ should be matched to tracks from the previous time $\mathbf{X}_{k-1}$ to produce updated tracks $\mathbf{X}_k$ for the current time. To address this, we use a GNN data association method which is solvable in polynomial time ($O(max(N_k, M_k)^3)$), via the Munkres assignment algorithm [21].

First, all tracks $\mathbf{X}_{k-1}$ are propagated to produce the state estimates $\mathbf{X}_{k|k-1}$ for time $k$. Next, the Munkres cost matrix is populated with the cost of assignment between every propagated track and every detection. The cost metric used in our tracking system is the Mahalanobis distance[3] between the detection $z_k^i$ and the propagated track $\mathbf{x}_{k|k-1}^j$. The covariance used to calculate this distance is the innovation covariance, which, in the case of the linear Kalman filter, is $\mathbf{S}_k^j = \mathbf{H}\mathbf{P}_{k|k-1}^j\mathbf{H}^T + \mathbf{R}$ where $\mathbf{P}_{k|k-1}^j$ is the prediction covariance of the propagated track $\mathbf{x}_{k|k-1}^j$. Here we use a higher noise covariance ($r = 0.5^2$) to allow more flexibility in associations.

A threshold is imposed such that detections outside the $p$th percentile confidence bounds of the expected observation are given an arbitrarily high assignment cost, $cost_{max}$. Since the Munkres algorithm requires a square matrix and $N_k$ is often different from $M_k$, the cost matrix is padded with $cost_{max}$ to produce the necessary shape. When an assignment is made between a detection and a track by the Munkres algorithm and the cost is $cost_{max}$, the assignment is ignored. Otherwise, the track is updated with the position and the confidence-level of the detection. The position is used to perform an observation update in the track's Kalman filter to produce the updated track positions $\mathbf{x}_k^j$ for the current time $k$. Tracks without matched detections are propagated forward without observations. Detections without matched tracks spawn new tracks at their current position. The confidence of each track, denoted $c_k^j$, is computed by the exponentially-weighted moving average $c_k^j = 0.95c_{k-1}^j + 0.05d_c(z_k^i)$, where $d_c(z_k^i)$ is the confidence of the $i$th detection which was assigned to the $j$th track.

Further, since people are represented as a single track but people's legs can produce between 0-2 observations (depending on the number of legs visible to the laser scanner and whether or not their legs have been clustered together), an identical temporary track is created for each person before the data association. If both of the person's tracks are matched to detected scan clusters, then the tracked person's position is updated with the mean of the detections' positions. If only one track is matched to a scan cluster, then the tracked person's position is updated with the mean of the detection's position and the propagated position of the person's track. After the person's original track has been updated, the temporary track is deleted.

*3) Local occupancy grid mapping:* To improve robustness, an odometry-corrected local occupancy grid map is constructed and updated every iteration with any scan clusters not associated with tracked people, on the assumption that these are non-human detections. The map covers a $20m \times 20m$ square area, at a resolution of $5cm \times 5cm$/cell, and is centred at the current position of the laser scanner. By default, all cells are assumed to be *freespace* until updated with a non-human observation.

The local map is used to assist with the data association by assigning the maximum cost, $cost_{max}$, to matchings between clusters in occupied space and human tracks. This is helpful to prevent situations where a tracked obstacle cluster is split due to an occlusion, and simultaneously, a nearby person's leg is occluded, potentially resulting in a match between the person track and the obstacle cluster. The map is also used to disallow person track initiations in occupied space, which can occasionally arise from non-human objects.

*4) Person track initiation and deletion:* To keep the number of person tracks manageable, initiation conditions are applied, using the following criteria: (1) a pair of tracks are detected, which move a given distance ($0.5m$) without drifting apart, (2) both tracks maintain a confidence level above a threshold $c_{min}$, and (3) both are in *freespace* (as computed from the local occupancy grid map). Upon initiating a person track, the person's position is estimated to be the mean of the positions of the pair of associated leg tracks. Then, the leg tracks are deleted and only one track is kept representing the position of the person.

Finally, human tracks are deleted when their innovation covariance $\mathbf{S}$ is greater than a threshold, or confidence over the track drops below the threshold $c_{min}$.

[3]Note that other GNN data association approaches have recommended minimizing the square of this distance [22]. We found best results minimizing the non-squared distance in our applications.

### C. Autonomous Following: Navigation and Control

To perform person following, a single human track is selected as input to an Object Following Controller (OFC) [23]. The track can be manually selected by the robot operator, or automatically selected using a predefined decision criteria, such as picking the human closest to the robot. The OFC uses the track's latest state estimate to compute the difference (position error) from the desired position of the human with respect to the robot (position goal). The position error is then used to compute the robot's velocity setpoints for low-level actuator controllers. The motion generated decreases the position error over time and results in a trajectory that causes the robot to follow the human.

To minimize the position error of the tracked human, the OFC modulates the robot's angular and linear velocity setpoints independently. Two vectors are used for these calculations: a vector from the robot's center to the goal position, and a vector from the robot's center to the human position. The first vector is expected to be constant (as long as the goal position does not change), while the second vector changes as the human and robot move with respect to each other. The control actions aim to equalize the length of these vectors, and drive the angle between them to zero. A Proportional-Integral-Derivative (PID) controller was implemented to calculate the angular velocity setpoint using the angle between the vectors, while a second PID controller calculates the linear velocity setpoint using the difference in lengths of the vectors. Both controllers were tuned using classical Ziegler-Nichols method. A dead-band zone was defined to address vibrations in the control actions when the position error is too small.

## V. Experimental validation

### A. Benchmark descriptions

Some prior benchmarks for the evaluation of the person detection and tracking module are available [24], however to the best of our knowledge, they are collected from a stationary laser scanner and are at a height of $0.8m$, which is above the leg-region of many pedestrians. The method of ground-truth annotation is also ambiguous, as it is unclear under what conditions a person is labelled as visible or occluded (an issue described in [25]). Further, the datasets are not ROS-enabled and the person-detection module used to detect people with the given laser scanner has not been released, making comparisons difficult on other platforms.

To palliate some of these gaps, and allow thorough evaluation both of our own and future methods, we collected and will be publicly sharing two new benchmarks for person detection and tracking, detailed in Table II. These provide 40 minutes of data recorded onboard moving robots, are fully ROS-enabled and include annotated people tracks labelled in the laser scans using an objective, unambiguous procedure, with video data to corroborate ground-truth when necessary.

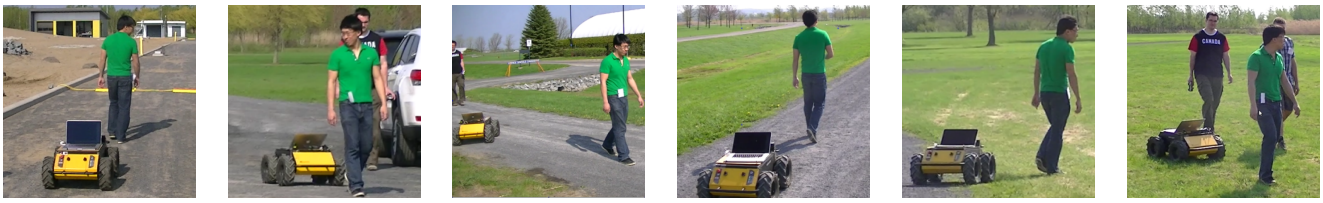
*1) General multi-person tracking:* The first benchmark, called *General multi-person tracking* is designed for evaluating performance of general multi-person tracking of pedestrians in natural environments. The benchmark is in fact composed of two datasets, one collected from a stationary robot and one collected from a moving robot. The stationary dataset includes 7 minutes of tracking data, while the moving dataset includes 5 minutes of data, and both are fully annotated, including 82 identified people tracks. The data was recorded onboard the SmartWheeler in the hallways of a university campus building during normal opening hours. Each recording includes odometry data published at 100Hz and laser data published at 7.5Hz. Video data was also captured, but was only used to as a reference for annotations of people in the laser scans, and was not included in the final curated benchmarks (though is available on demand).

All ground-truth people positions were hand-labelled in each laser scan. To address the issue of whether or not a person should be labelled as visible if they are partially occluded (an issue raised in [25]), a consistent and objective rule was used: if a minimum of at least three laser points can be clustered with a Euclidean distance of $0.13m$, and that cluster of points corresponds to a person, then they are marked as visible in the annotations.

*2) Tracking for following:* The second benchmark, called *Tracking for following*, is designed to measure the performance of the detection and tracking modules when applied specifically to the task of tracking an individual person in the presence of other pedestrians. This benchmark is also composed of two separate datasets.

In the Following Indoor dataset, the person to be followed was asked to walk naturally and stop periodically to interact with objects in the environment while the SmartWheeler followed either from behind or side-by-side, simulating a person-following situation (for data collection in this benchmark, the robot was manually controlled). The environment is the same university building as for the *General multi-person tracking* benchmark, and includes scenes with natural crowds and clutter, as well as five instances of pedestrians walking between the SmartWheeler and the person it was following. Odometry was collected at a frequency of 100Hz and laser scans at 15Hz. The position of the person being followed was hand-labelled in every frame. Altogether, 21 minutes of data were gathered and annotated, including 4.5 minutes of side-by-side following and 16.5 minutes of following from behind.

The Following Outdoor dataset was collected with a Clearpath Husky robot at the Canadaian Space Agency in Saint-Hubert. The laser scanner was mounted level with the ground, approximately $40cm$ high. The dataset contains 12 minutes of data, also fully annotated. Laser scans were collected at a frequency of 10Hz. Odometry data was collected as well but was ultimately not used as it was found to be detrimentally inaccurate. The challenge in this dataset is dealing a high degree of sensor noise, as the laser scanner used is indoor-rated and highly susceptible to interference from the sun. In this case, the robot was controlled autonomously to follow the person using the navigation system presented in Sec. IV-C and an earlier version of the Joint Leg Tracker presented in Sec. IV. During the experiment, the tracker failed on two separate occasions and steered the



(a) (b) (c) (d) (e) (f)

Fig. 2. The person tracking and following system implemented on a Clearpath Husky during the collection of the Following Outdoor dataset. The robot autonomously followed the participant in an approximately $500m$ loop on gravel and grass at the Canadian Space Agency in Saint-Hubert. Two tracking failures occurred, one of which is shown in (c), which were caused by extensive sensor noise, as the laser scanner used is indoor-rated and highly susceptible to interference from the sun.

robot such that the person to be tracked was lost out of frame. Fig. 2 shows sample images from this dataset.

### B. Comparison Person Tracking Approaches

Quantitatively comparing to existing methods from the literature is challenging due to a scarcity of publicly available code. In our search, we were only able to find one other open-source ROS-enabled method which could be compared to ours: the ROS leg_detector package used in [4]. To adapt the ROS leg_detector to our benchmarks, all parameters were kept at their default values except the confidence threshold was lowered, as the tracker would commonly fail to initiate people tracks with its default value. Also, the clustering distance and minimum required points per cluster were set to be the same as for our method.

A variant of our algorithm, which we call the Individual Leg Tracker, is also included in the comparison. It is identical to the Joint Leg Tracker described in Sec. IV, except that all tracking is performed on an individual-leg level (not pairs of legs) and it does not use a local occupancy grid map for data association. In this case, person tracks are deleted when the two leg tracks separate beyond a distance threshold, one of the leg tracks has too low of a confidence, or one of the leg tracks is deleted.

Minor modifications of all methods were made to allow for repeatable results on the benchmarks. This was necessary because the tracking methods require coordinate transformations be made in real-time, which can cause benchmarks results to vary depending on the exact time these transformations are made. Each method was therefore set to use the scan header time to perform the transforms and, when they were not available, to wait for one second and, if they were still not available, the current scan was skipped (although, we found this happened only very rarely). An option was included in each method to never wait for transforms to become available but instead to always use the most recent ones. This is how the system is intended to be used in practice and was the variation used for runtime profiling.

### C. Evaluation metrics

*1) General multi-person tracking:* We use the CLEAR MOT metrics for quantitative evaluation of the tracking system [26]. They are commonly used metrics for multi-object tracking, and provide scorings of valid assignments, ID switches, misses, false positives (FPs) and precision of matchings cumulated from every frame. These scores can be aggregated into a combined overall multi-object tracking accuracy (MOTA) score

$$MOTA = 1 - \frac{\sum_k (ID_k + Miss_k + FP_k)}{\sum_k g_k}$$

where $ID_k$, $Miss_k$, $FP_k$ and $g_k$ are number of ID switches, misses, FPs and ground truth annotations respectively, at time $k$. However, the aggregation of the scorings in the combined MOTA score assumes that three types of errors (ID switches, misses and FPs) are equally unfavorable, which is arguably not true in most applications, including ours. We therefore report the raw count of each error type alongside the MOTA score.

Another relevant CLEAR MOT metric is the mult-object tracking precision (MOTP), which is defined as

$$MOTP = \sum_{i,k} d_k^i / \sum_k c_k$$

where $c_k$ is the number of matchings made between estimated people positions and ground truth positions at time $k$ and $d_k^i$ is the distance between the $i$th match. Intuitively, it provides a measure of how precise a target is tracked when it is being tracked properly. In the context of person-following, a lower MOTP score would be beneficial because a more precise location of the person being followed would presumably allow the following controller to track the person more precisely.

A threshold distance of $0.75m$ was used for the CLEAR MOT matching to determine whether a tracked person should be matched to an annotated ground truth person.

*2) Tracking for following:* For evaluation in the *Tracking for following* scenarios, the same CLEAR MOT metrics are used. However, since we are only concerned with the tracker's ability to track a particular person (among several), only the ground-truth positions of the target person were labelled. Other pedestrians were ignored, and only person-tracking events corresponding to the target person are reported (i.e., FPs were ignored because they may have been due to unlabelled pedestrians).

Under such an evaluation framework, the relevant metrics are ID switches and misses. ID switches represent cases where the track of the target person was switched with someone or something else, and it would no longer be possible to

TABLE II

OVERVIEW OF BENCHMARKS.

| Benchmark | Dataset | Duration | Avg. estimated robot speed when moving (m/s) | Annotated people tracks | Avg. dist. to person followed (m) |
|---|---|---|---|---|---|
| *General multi-person tracking* | Stationary Robot | 7m11s | n/a | 45 | n/a |
| | Moving Robot | 5m01s | 0.9 | 37 | n/a |
| *Tracking for following* | Following Indoor | 21m24s | 0.9 | 1 | $1.33 \pm 0.57$ |
| | Following Outdoor | 12m00s | 0.6 | 1 | $1.68 \pm 0.30$ |

TABLE III

BENCHMARK RESULTS.

| Dataset | Tracker | Valid | ID Switch | Miss | FP | MOTA | MOTP (m) | Runtime Worst/Avg (Hz) |
|---|---|---|---|---|---|---|---|---|
| Stationary Robot | leg_detector [4] | 427 | 51 | 1605 | 28 | 19.2% | 0.28 | 19/25 |
| | Individual Leg | 569 | 10 | 1504 | 61 | 24.4% | 0.23 | 14/25 |
| | Joint Leg | **703** | **8** | **1372** | **11** | **33.2%** | **0.16** | 15/25 |
| Moving Robot | leg_detector [4] | 149 | 15 | 481 | 294 | -22.5% | 0.27 | 19/25 |
| | Individual Leg | 160 | **2** | 483 | **88** | **11.2%** | 0.17 | 12/24 |
| | Joint Leg | **163** | **2** | **480** | 97 | 10.2% | **0.15** | 11/24 |
| Following Indoor | leg_detector [4] | 15425 | 196 | 3425 | n/a | n/a | 0.15 | 18/25 |
| | Individual Leg | 17295 | 73 | 1678 | n/a | n/a | 0.13 | 4/19 |
| | Joint Leg | **18554** | **7** | **485** | n/a | n/a | **0.09** | 4/19 |
| Following Outdoor | leg_detector [4] | 4835 | 172[1] | 1172 | n/a | n/a | **0.09** | 22/25 |
| | Individual Leg | 5348 | 12[1] | 819 | n/a | n/a | 0.11 | 15/25 |
| | Joint Leg | **6073** | **2**[1] | **104** | n/a | n/a | **0.09** | 18/25 |

autonomously follow them. Misses represent cases where the target person was visible in the laser scanner's field-of-view but was not tracked. In such cases, the robot user would not be able to lock into the target person to initiate following.

### D. Results

Results from all datasets are shown in Table III. The Joint Leg Tracker achieves the best results in the *Tracking for following* tasks by a large margin. It suffers from only 7 ID switches in the 21 minutes of the Following Indoor dataset and makes virtually no ID switch errors in the Following Outdoor dataset (assuming the 2 ID switches which were caused by the person being tracked moving out of frame for a significant amount of time to not be preventable). It also achieves the lowest MOTP on all datasets, meaning that when a person is being tracked properly, it provides the most precise estimate of their location. This would presumably improve the performance of the following controller, which uses the location estimate to perform closed-loop control.

Its performance when applied to the *General multi-person tracking* benchmark is also favourable but by a lesser margin. It causes significantly less of every type of error on the Stationary Robot dataset compared to the others and achieves similar results to the Individual Leg Tracker on the Moving Robot dataset.

The leg_detector is outperformed in almost all respects by the other tracking methods and is generally unable to track anyone persistently in the challenging benchmark environments, as is shown by its high number of ID switches in all cases.

*Runtime:* The average-case runtime of the Joint Leg Tracker is faster than the maximum scanner frequency in all cases when run on an Intel Core i7 CPU. In the Following Indoor dataset, which has frequent open areas containing many distinct objects to be tracked, the slowest single update is slower than the scan frequency by a factor of approximately four, though the average runtime over the entire dataset is still better than real-time. The bottleneck in these cases tends to be the tracking and data association step, which is implemented in Python. In the future we plan to test a C++ implementation and, if this does not achieve worst-case, real-time performance, limit the number or distance of clusters which are tracked.

## VI. DISCUSSION

This work presents a novel method for detecting, tracking and following people using laser scanners at leg-height. The system integrates a joint leg tracker with local occupancy grid maps and a method of tracking all scan clusters, including non-human clusters, to improve tracking in cluttered areas. Empirically, it has shown to be effective in the target application of autonomous person following and presents advantages in general multi-person tracking as well. The tracking method was tested on the SmartWheeler and also integrated with an autonomous following navigation system deployed on a Clearpath Husky. All datasets and code are publicly available on the first author's website.

*Limitations:* Since the Joint Leg Tracker was developed in the ROS framework, it should be relatively straightforward to transfer to other robots with laser scanners at similar heights. One caveat, however, is that the human-confidence learning algorithm may require re-training for optimal performance with laser scanners of different resolutions.

Another potential limitation of the presented system is the GNN data association method. While it is computa-

[1]Due to the two cases where the person followed was lost out of frame for a significant amount of time, it is reasonable to expect a minimum of two ID switches in this dataset.

tionally faster than the MHT, quantitative experiments exist demonstrating better performance of more sophisticated data association methods, such as the JPDAF and the MHT, in other application areas. For example, meta-results reported by Blackman and Popoli [12] suggest that the JPDAF and the MHT generally track targets better in high-clutter environments with many false positive detections when used in the task of radar tracking. However, quantitative comparisons specifically in the area of people tracking in 2D laser scans would be beneficial because it is a fundamentally different domain. For example, false positive detections are produced systematically by objects in the scan, rather than randomly over the scanning area, as is often assumed in radar tracking, and can therefore be accounted for explicitly (e.g., as in our tracker or in Luber et al. [27]).

*Evaluation metrics:* The overall CLEAR MOT scores presented in the benchmarks are significantly lower than those presented in [24]. However, the results are not directly comparable because of the different sensor configurations and environments. For example, the laser sensor used in the *General multi-person tracking* benchmark has a lower scanning frequency (7.5Hz vs 37.5Hz) and a shorter range ($8m$ vs $80m$) than the laser sensor from the benchmarks in [24]. The shorter range naturally increased the number of misses since pedestrians would enter and leave the field of view quicker and with fewer detections, and would therefore spend proportionally more time being tracked as non-persons before person tracks were initiated.

*Impact and future work:* The system development and experiments completed thus far indicate that we have achieved a portable and reliable system. Moving forward, we intend to verify the usefulness of the system on the SmartWheeler with the target user population. One goal will be to determine if the system is capable of allowing a smart wheelchair user to comfortably carry on a conversation with a companion while autonomously following them side-by-side in busy environments. We are also investigating the potential for using this system to help automatically assess patients' locomotion patterns following an injury [7] in the context of a rehabilitation therapy.

## ACKNOWLEDGEMENTS

The authors would like to thank Bénédicte Leonard-Cannon, Gheorghe Comanici, Michael Mills, Qiwen Zhang, Andrew Sutcliffe, Martin Gerdzhev and Chenghui Zhou for their gracious help.

This work was supported by the Natural Sciences and Engineering Research Council (NSERC) through the NSERC Canadian Field Robotics Network (NCFRN).

## REFERENCES

[1] A. Cosgun, D. A. Florencio, and H. I. Christensen, "Autonomous person following for telepresence robots," in *Int. Conference on Robotics and Automation (ICRA)*, 2013.

[2] D. Kairy, P. Rushton, P. Archambault, E. Pituch, C. Torkia, A. ElFathi, P. Stone, F. Routhier, R. Forget, L. Demers, J. Pineau, and R. Gourdeau, "Exploring powered wheelchair users and their caregivers' perspectives on potential intelligent power wheelchair use: A qualitative study," *Int. Journal of Environmental Research and Public Health*, pp. 1–7, 2014.

[3] K. O. Arras, B. Lau, S. Grzonka, M. Luber, O. M. Mozos, D. Meyer-Delius, and W. Burgard, "Range-based people detection and tracking for socially enabled service robots," in *Towards Service Robots for Everyday Environments*, pp. 235–280, 2012.

[4] D. V. Lu and W. D. Smart, "Towards more efficient navigation for robots and humans," in *Int. Conference on Intelligent Robots and Systems (IROS)*, 2013.

[5] P. Boucher, A. Atrash, S. Kelouwani, W. Honore, H. Nguyen, J. Villemure, F. Routhier, P. Cohen, L. Demers, R. Forget, and J. Pineau, "Design and validation of an intelligent wheelchair towards a clinically-functional outcome," *Journal of NeuroEngineering and Rehabilitation*, vol. 10, pp. 1–16, 2013.

[6] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers, "People tracking with mobile robots using sample-based joint probabilistic data association filters," *Int. Journal of Robotics Research*, 2003.

[7] A. Leigh and J. Pineau, "Laser-based person tracking for clinical locomotion analysis," *IROS Workshop on Rehabilitation and Assistive Robotics*, 2014.

[8] D. Montemerlo, S. Thrun, and W. Whittaker, "Conditional particle filters for simultaneous mobile robot localization and people-tracking," in *Int. Conference on Robotics and Automation (ICRA)*, 2002.

[9] E. A. Topp and H. I. Christensen, "Tracking for following and passing persons.," in *Int. Conference on Intelligent Robots and Systems (IROS)*, 2005.

[10] R. Gockley, J. Forlizzi, and R. Simmons, "Natural person-following behavior for social robots," in *Int. conference on Human-robot Interaction*, 2007.

[11] S. Hemachandra, T. Kollar, N. Roy, and S. Teller, "Following and interpreting narrated guided tours," in *Int. Conference on Robotics and Automation (ICRA)*, 2011.

[12] S. S. Blackman and R. Popoli, *Design and analysis of modern tracking systems*. Artech House, 1999.

[13] L. E. Navarro-Serment, C. Mertz, N. Vandapel, and M. Hebert, "Ladar-based pedestrian detection and tracking," in *IEEE Workshop on Human Detection from Mobile Platforms*, 2008.

[14] M. Munaro and E. Menegatti, "Fast rgb-d people tracking for service robots," *Autonomous Robots*, 2014.

[15] A. P. Gritti, O. Tarabini, J. Guzzi, G. A. D. Caro, V. Caglioti, L. M. Gambardella, and A. Giusti, "Kinect-based people detection and tracking from small-footprint ground robots," in *International Conference on Intelligent Robots and Systems (IROS)*, 2014.

[16] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, 1997.

[17] M. Kobilarov, G. Sukhatme, J. Hyams, and P. Batavia, "People tracking and following with mobile robot using an omnidirectional camera and a laser," in *Int. Conference on Robotics and Automation (ICRA)*, 2006.

[18] L. Breiman, "Random forests," *Machine learning*, 2001.

[19] G. Bradski, "The opencv library," *Doctor Dobbs Journal*, vol. 25, no. 11, pp. 120–126, 2000.

[20] R. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, 1960.

[21] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[22] P. Konstantinova, A. Udvarev, and T. Semerdjiev, "A study of a target tracking algorithm using global nearest neighbor approach," in *Int. Conference on Computer Systems and Technologies*, 2003.

[23] N. A. Olmedo, H. Zhang, and M. Lipsett, "Mobile robot system architecture for people tracking and following applications," in *Int. Conference on Robotics and Biomimetics (ROBIO)*, 2014.

[24] M. Luber and K. O. Arras, "Multi-hypothesis social grouping and tracking for mobile robots.," in *Robotics: Science and Systems*, 2013.

[25] A. Milan, K. Schindler, and S. Roth, "Challenges of ground truth evaluation of multi-target tracking," in *Int. Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2013.

[26] B. Keni and S. Rainer, "Evaluating multiple object tracking performance: the clear mot metrics," *EURASIP Journal on Image and Video Processing*, 2008.

[27] M. Luber, G. D. Tipaldi, and K. O. Arras, "Place-dependent people tracking," *Int. Journal of Robotics Research (IJRR)*, 2011.