PEOPLE AWARE MOBILE ROBOT NAVIGATION

A Thesis
Presented to
The Academic Faculty

by

Akansel Cosgun

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
College of Computing

Georgia Institute of Technology
August 2010

PEOPLE AWARE MOBILE ROBOT NAVIGATION

Approved by:

Professor Ignatius Arrogant,
Committee Chair
College of Computing
Georgia Institute of Technology

Professor Henrik Christensen, Advisor
College of Computing
Georgia Institute of Technology

Professor General Reference
School of Mathematics
Georgia Institute of Technology

Professor Ivory Insular
Department of Computer Science and
Operations Research
North Dakota State University

Professor Earl Grey
College of Computing
Georgia Institute of Technology

Professor John Smith
College of Computing
Georgia Institute of Technology

Professor Jane Doe
Another Department With a Long
Name
Another Institution

Date Approved: 1 July 2010

To myself,

Perry H. Disdainful,

the only person worthy of my company.

PREFACE

Theses have elements. Isn't that nice?

ACKNOWLEDGEMENTS

I want to thank people

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Why should I provide a summary? Just read the thesis.

CHAPTER I

INTRODUCTION

Introduction

CHAPTER II

MAP ANNOTATION

Map Annotation

2.1 Related Work

Related Work

2.2 Semantic Maps

Semantic Maps

2.2.1 Waypoints

2.2.2 Planar Landmarks

2.2.3 Objects

2.3 User Interface

User Interface

2.4 Pointing Gestures for Human-Robot Interaction

Pointing Gestures

CHAPTER III

NAVIGATION AMONG PEOPLE

Autonomous Robot Navigation

3.1 Related Work

Related Work

3.2 State of Autonomous Robot Navigation

State of Autonomous Robot Navigation

3.3 Finding Goal Points for Navigation

Finding Goal Points for Navigation

3.4 People Aware Navigation

People Aware Navigation

3.5 Speed Maps for Safe Navigation

Speed Maps for Safer Navigation

CHAPTER IV

MULTIMODAL PERSON DETECTION AND TRACKING

The ability to robustly track a person is an important prerequisite for human-robot interaction. To realize any task that involves humans, the challenge is the detection and tracking of humans in the vicinity of the robot considering the robot's movements, sensing capabilities and occlusions. The scope of how much information is needed from the human perception module depends on the objective of the application. First, the robot should determine if there are people nearby. If the robot senses people around, the robot should find out *where* they are. Representing people as points (x,y) in maps is common practice for navigation planning. If the task requires the robot to face a person, then the orientation θ needs be detected. The robot further can determine *who* the detected person is. Identification of humans is necessary for enabling non-generic service. Finally, the robot should interpret *what* the person is doing by analyzing the motion features and through gesture analysis. Tracking body parts of humans over time give significant information about human activity.

We focus on tracking people who are either walking or standing, as these are the two most common human poses around a mobile robot. Many full-body or body part detectors have been developed in the literature, reviewed in Section 4.1. Full-body detectors are not suitable for mobile robot navigation applications because of their inability of capturing the entire body with on-board sensors when people are close to the robot. We aim to robustly track a person 360° around the robot. However, most sensors have a limited field of view and using only a single detector can lead to a system with a single point of failure. Therefore, we think a multimodal detection system is better suited for on-board people tracking for our use cases.

Laser scanners are the natural sensor of choice as state-of-the-art mobile robots are already equipped with an ankle-height laser scanner that is mainly used for navigation. The laser scanners we used on our robot are Hokuyo UTM 30-LX, which has 270° Field of View (FOV), 0.25° angular resolution, $40Hz$ refresh rate and $30m$ maximum range. We are only interested in detections in close range (less than $5m$). In that range interval, and the accuracy of each laser reading is $\pm 3cm$, which is sufficient for our use cases. The relatively higher accuracy and resolution are the two advantages of laser scanners over cameras and RGB-D cameras. Cameras, on the other hand, have the advantage of providing richer information, which can be used to extract body parts. We use a combination of detectors using either a laser scanner and RGB-D camera for robustness and better coverage, described in Section 4.2. Representing people as a points in the map is sufficient for mobile robot navigation and each detector produces a point as a person hypothesis. We use a real-time probabilistic tracking framework that relies on the fusion of the multiple person detections, described in Section 4.3. For certain applications, identifying specific users allows the robot to go beyond generic capabilities. We present our face recognition method in Section 4.4.

4.1 Related Work

Person detection was first addressed by the computer vision community as an object detection problem. Early research on person detection using vision is surveyed by Moeslund [21]. Face detection is a common method for detecting people, with the work of Viola and Jones [32] being the most popular one. See Zhang [35] for a survey on contemporary approaches on vision based face detection. Another popular topic has been pedestrian detection in crowded scenes Leibe [19] and Tuzel [31].

In 2000's, laser scanners became the de-facto sensor for localization and mapping. Laser scanners are usually placed slightly above floor for obstacle avoidance, therefore leg detection is common practice. Early works by Montemerlo [22] and Schulz [25]

focused on tracking multiple legs using particle filters. Legs are typically distinguished in laser scans using geometric features such as arcs [33] and boosting can be used to train a classifier on a multitude of features [1]. Topp [29] demonstrates that leg tracking in cluttered environments is prone to false positives. For more robust tracking, some efforts fused information from multiple lasers such as Carballo's work [7], which uses a second laser scanner at torso level. Glas [12] uses a network of laser sensors at torso height in hall-type environments to track the position and body orientation of multiple people. Several works used different modalities of sensors to further improve the robustness. Kleinehagenbrock [18] and Bellotto [4] combine leg detection and face tracking in a multi-modal tracking framework. Other examples include combining sound localization and vision [5] and combining RFID tracking and vision [11].

Laser-based person methods pertains tracking of humans in 2D, projected to floor plane. Tracking of the body parts has long been a topic of interest in vision [3, 27]. With the recent introduction of 3D sensors such as the Velodyne, Swissranger and Kinect, more robust tracking became possible. Spinello [28] trains geometrical features at different height levels in the 3D point cloud for pedestrian detection. Ganapathi [9] estimates body part locations with a probabilistic model. One of the well-known skeleton tracking algorithms is the Microsoft Kinect SDK by Shotton [26], which trains decision forests using simple depth features in a vast database. This software is not suitable to work on a mobile robot as it is designed to work on a stationary sensor. In the robotics community, there are efforts to develop skeleton trackers that work on mobile robots and in unstructured scenes [6].

Face recognition is a widely used application as surveyed by Phillips [24]. One of the pioneers in face recognition uses a set of patch masks for features that doesn't necessarily correspond to eyes, ears or noses [30]. [36] combines PCA (Principal

Component Analysis) and LDA (Linear Discriminant Analysis) to improve the generalization capability when only a few samples are available.

There has been some work to identify humans using 3D data, such as the head-to-shoulder signature [17] and body motion characteristics [23]. Biometric person identification techniques, such speaker recognition [16], 3D ear shape [34] and multi-modal cues [10] have potential to be more accurate than face recognition. However, these approaches are better suited to work in controlled environments.

4.2 Person Detection

In this section, we present our person detectors, namely leg detection (Section 4.2.1) and torso detection (Section 4.2.2). We also use an implementation of an upper body detector by Mitzel [20], which uses a template and the depth information of a RGB-D camera to identify upper bodies (shoulders and head), designed to work for close range human detection using head mounted cameras.

4.2.1 Leg Detection

A front-facing laser scanner at ankle height is used for leg detection. The output of a laser scanner at each iteration is an array of range measurements, represented in the polar coordinate system. We first convert the range data to Cartesian coordinate system:

$$x_i = \sum_{\phi=\phi_{start}}^{\phi_{end}} r_i \cos(\phi)$$

$$y_i = \sum_{\phi=\phi_{start}}^{\phi_{end}} r_i \sin(\phi)$$

Then we apply segmentation, Segmentation produces clusters of consecutive scan points, which due to their proximity, have a high likelihood of belonging the same object. Two adjacent distance measurements are considered to be in the same segment if the Euclidean distance between them is below a threshold value. Starting from the

start of the range array, a new segment is started if $|r_i - r_{i+1}| < d_{cluster}$. Although some approaches use a variable segmentation threshold that is a function of the range, we use a fixed clustering threshold $d_{cluster} = 0.1m$. The segmentation process results in a set of segments $\mathbf{S}$. A set of geometric features are extracted from the laser segment.

In a laser scan, legs can appear in different patterns [29]. We look only single leg and person-wide blob patterns as these two cover all the ways legs can be seen in a laser scan. Depending on the application, we accept either only the single leg pattern or both of the patterns (explained in Section **??**.

There are a number of geometric features that can be extracted from a laser segment, as delineated by Arras [1]. We use three geometric features that is used to detect a leg: segment width, circularity, and Inscribed Angle Variance (IAV):

1. Segment Width: Measures the Euclidean distance between the first and last point of a segment S_i

2. Segment Circularity: This measure is a simple measure to assess if the segment shape resembles a circle. The circularity criterion we used is the ratio of the perpendicular distance from the middle point to the line segment that connects start and end points, to the segment width. For example, in a perfect half circle in Figure 1, the circularity criterion is $|\overline{P_0 P_n}/d_{mid} = 0.5$. In case of a laser scan, as can be seen in Figure 2, we again consider the ratio of d_{mid} to segment width. For this calculation we only consider the middle point as it provides a simple heuristic on circularity.

3. Inscribed Angle Variance (IAV): This feature is originally proposed by Xavier [33], in order to detect circles. We adopt IAV in order to detect legs, which are not necessarily circle-shaped, especially for the person-wide blob pattern. As an example, inscribed angles on a circle is shown in Figure 3. As a geometric

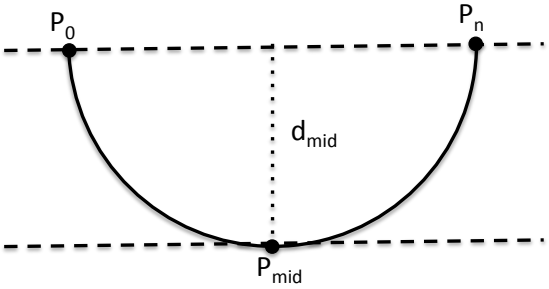

Figure 1: Circularity criterion in a perfect circle is: $|P_0P_n|d_{mid} = 0.5$

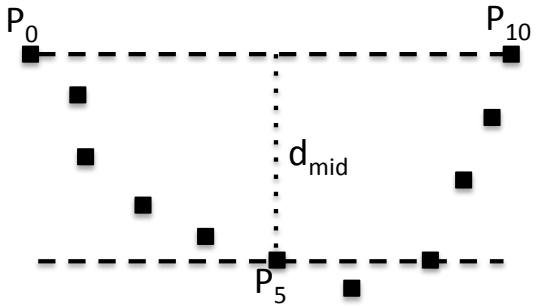

Figure 2: Circularity criterion in a this laser segment is: $|P_0P_{10}|/d_{mid}$

property of the circle, $\angle P_0P_1P_4$ and $\angle P_0P_2P_4$ are equal angles. IAV for a given set of points is the average of all inscribed angles:

$$IAV_S = \sum_{P=P_1}^{P_{n-1}} \angle P_0PP_n$$

For a perfect circle, $IAV_S = 90°$. For shapes that are not perfect circles but are similar to circles, IAV feature should be consistent. Laser segments from a leg usually resemble a circle, therefore we use IAV as one of the features for leg detection.

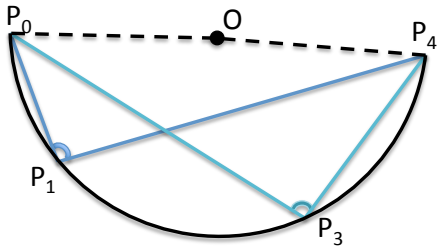

Figure 3: Inscribed angles of an arc are shown in the figure. Inscribed Angle Variance (IAV) is calculated by taking the average of all inscribed angles on a laser segment.

In order to be able to use these values, we first found the nominal feature values for an average human leg. We captured the laser scan data while the robot followed a person through an office environment. The following method used for this experiment will be discussed in detail in Section 5.2. For the training set, two people's legs were recorded with different clothing (shorts, baggy pants and trousers) to account for

Segment type	Width(m)		Circularity		IAV($radians$)	
	μ	σ	μ	σ	μ	σ
Single Leg	0.13	0.03	0.25	0.15	2.23	0.4
Personwide blob	0.33	0.07	0.14	0.09	2.61	0.16
Other	0.22	0.12	0.1	0.11	2.71	0.38

Table 1: Table shows average and standard deviations of geometric leg features calculated in our dataset.

variance in the leg parameters. About 17×10^3 Single Leg patterns and 0.6×10^3 person-wide blobs were manually labeled in the data. In addition, 120×10^3 segments were labeled as 'other' or 'not a leg'. The average and variance of the aforementioned geometric features for single leg, personwide blob, as well as other segments are given in Table 1.

For every segment S_i in a test laser scan, we first extract the geometric features f_1^i, f_2^i, f_3^i. We then calculate the weighted Mahalanobis distance to the average leg parameters for the each leg pattern:

$$D_{mah}^i = \sum_{j=1}^{n_{features}} w_j \frac{(f_j^i - \mu_j)^2}{\sigma_j^2} \tag{1}$$

where w_j are the weights for each feature, mu_j and $sigma_j$ are pulled from Table 1. The resulting Mahalanobis distance is then compared with a detection threshold. If $Dmah_{leg}^i < Threshold_{leg}$, the segment S_i is considered a detection. $Threshold_{leg}$ defines how many standard deviations away from the average features are allowed. In our implementation, we empirically set the feature weights as: $\mathbf{W}_{leg} = (0.35, 0.26, 0.39)$, in the feature order given in Table 1. For normal operation, we set $Threshold_{leg} = 1.5$, which accounts for about %95 of the detections. If only one person is being tracked, we use a higher threshold. The reason behind will be explained in Section 4.3.

4.2.1.1 Associating Leg Segments

After single leg patterns are detected, we try match the leg segments. We extend our leg detection approach to determine which leg segments are connected. Note that

this method applies if there is a RGB-D camera pointing to the lower body of the human. For each leg segment pair, if both of them are within the FOV of the RGB-D sensor, we use our algorithm to determine whether there is a connectivity between two candidate leg segments. If a connectivity is found, then the leg segments pair is qualified to be a leg segment pair representing a person. See Figure 4 as an example result. Figure 5 shows the flow chart of the association algorithm.

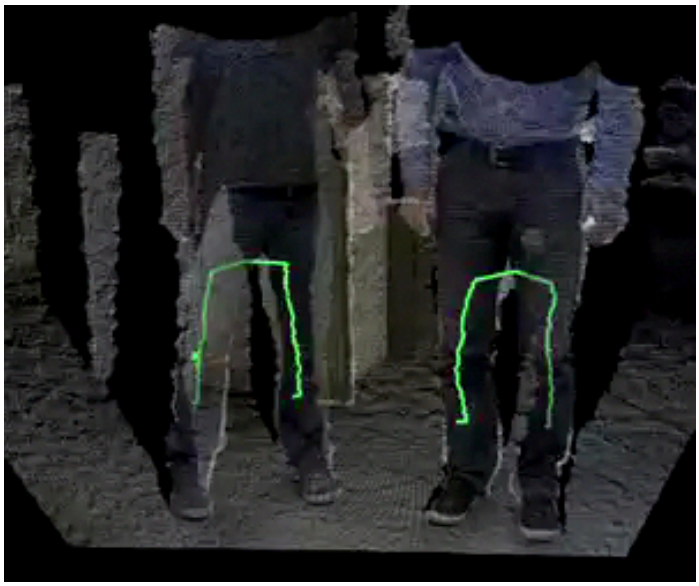

Figure 4: Two person detections are seen in this figure. Our leg segment association algorithm propagates pixels vertically from candidate leg segments and connects leg pairs.

First, the centroids each of the two candidate leg segments are found. These points are projected onto the depth image acquired from the RGB-D camera. At each iteration, each leg segment, our algorithm first propagates horizontally to both directions in the depth image, then the center pixel is located and it propagates 1 pixel vertically ($+z$ direction). If there are no connectivity after a number of iterations, then we conclude that the candidate leg pair does not represent a person. If there is a connectivity at some point, we then assign a likelihood score to the pair as a function of the vertical propagation height. If this score is higher than a threshold, then the algorithm concludes that the leg candidate segments represent a person.

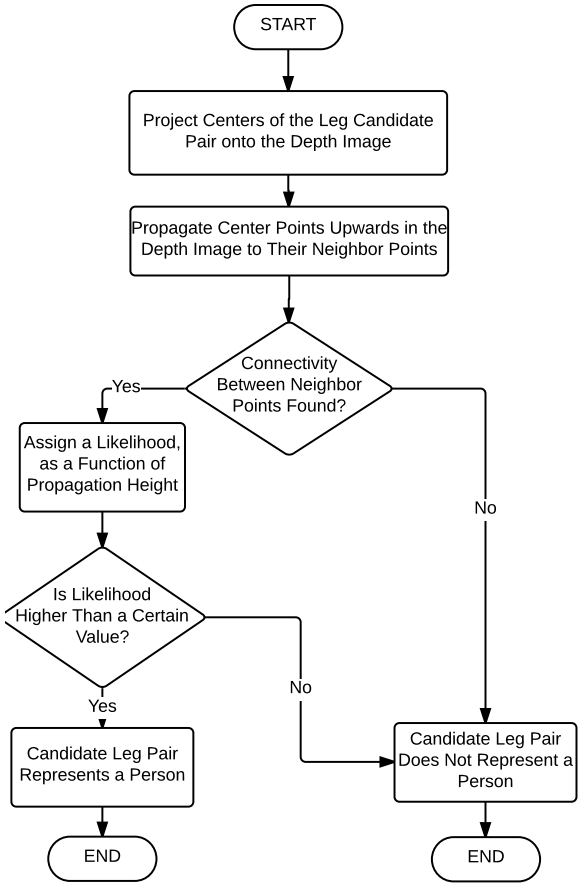

Figure 5: Flow chart for determining if two leg segment candidates belong to a person.

The propagation scoring eliminates most of the false positives due to sensor noise and non-human shapes.

4.2.2 Torso Detection

In this section, we describe our torso detection approach. For this detector, we used another Hokuyo UTM 30-LX laser scanner, placed at torso height ($1.27m$). Our approach relies on fitting an ellipse to laser segments and determining the detection result by interpreting the axis lengths (Figure 6). Our torso detector allows us to detect the orientation of the person unlike the laser-based leg detectors, therefore this detector is also suitable for applications that relies on extracting the orientation of the person from a single laser scan.

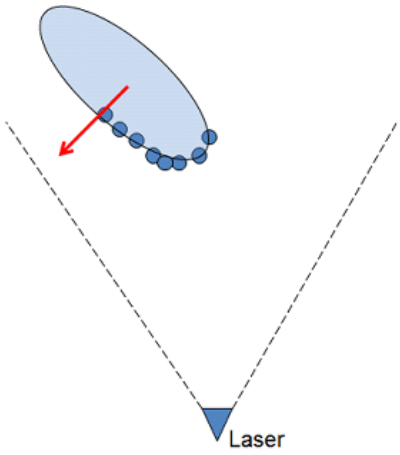

Figure 6: Our torso detector fits and ellipse to the human torso and estimate its position and orientation.

The first step to detect torsos in a laser scan is to segment the laser scan. We use the same segmentation technique used for leg detection, explained in Section 4.2.1. We then fit an ellipse to each laser segment. We use a numerical ellipse fitting method that solves the problem with a generalized eigensystem, introduced Fitzgibbon [8]. This ellipse fitting method is robust, efficient and ellipse-specific, so that even very noisy sensor data will always return an ellipse. Compared to iterative methods, it is computationally very efficient, therefore the speed of the calculations is limited to the laser scan refresh rate.

The ellipse fitting algorithm provides us with the centroid and orientation of the ellipse as well as the minor and major axis lengths. We use the minor and major axis lengths, as well as the three geometric features introduced in Section 4.2.1 in order to determine if the laser segment is a person. We gathered a 450 laser scans while a person stood in front of the sensor and made a one full turn around himself. We calculated the mean and standard deviation of the all five features, which is given in Table 3. For a given laser segment, we find the weighted Mahalanobis distance in Equation 1 to the averaged parameters. If $Dmah^{i}_{torso} < Threshold_{torso}$, the segment is considered a detection. The feature weight constants we used was $\mathbf{W}_{torso} = (0.19, 0.09, 0.35, 0.24, 0.13)$, in respective order given in Table 2. These

Torso Features	μ	σ
Width(m)	0.44	0.12
Circularity	0.32	0.18
IAV($radians$)	2.57	0.38
Major axis length(m)	0.39	0.08
Minor axis length(m)	0.17	0.06

Table 2: Table shows average and standard deviations of geometric features for a human torso in laser scans.

values were empirically determined, although one can do more sophisticated analysis for optimal weights.

Figure 7 shows how the torso detection rate changes for a given Mahalanobis Distance Threshold in our dataset. What is not displayed in the plot is that higher torso detection rate also means higher rates of false positives. For normal operation, we set $Threshold_{torso} = 1.25$, which accounts for about %90 detection rate. If the tracker is dedicated to track only a single person, then we use a higher threshold: $Threshold_{torso} = 2.5$. The reasoning behind this threshold selection will be discussed in Section 4.3.

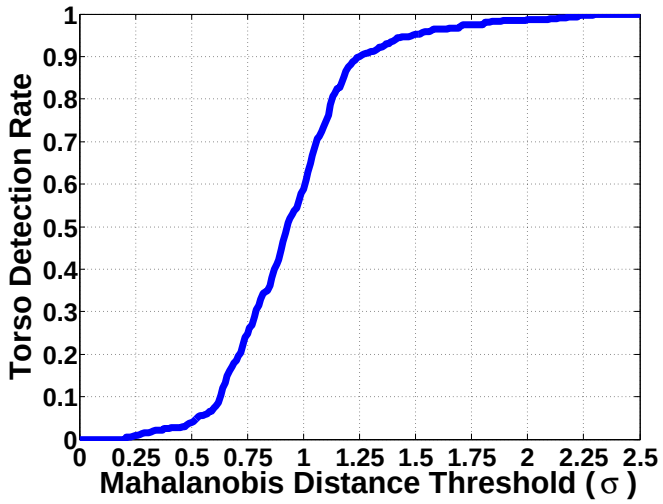

Figure 7: Torso detection rate vs weighed Mahalanobis Distance Threshold in our dataset

4.2.2.1 Evaluation of Torso Detection

In order to evaluate the accuracy of the position and orientation estimations of our torso detection method, we collected torso data from 23 people. Subjects were instructed to stand on 4 targets at different distances with 8 different orientations on each target. Experimental setup from the sensor's view is shown in Figure 8. For each pose at every target, we logged the position and orientation estimation of the torso detector and compared it with ground truth, which is fixed.

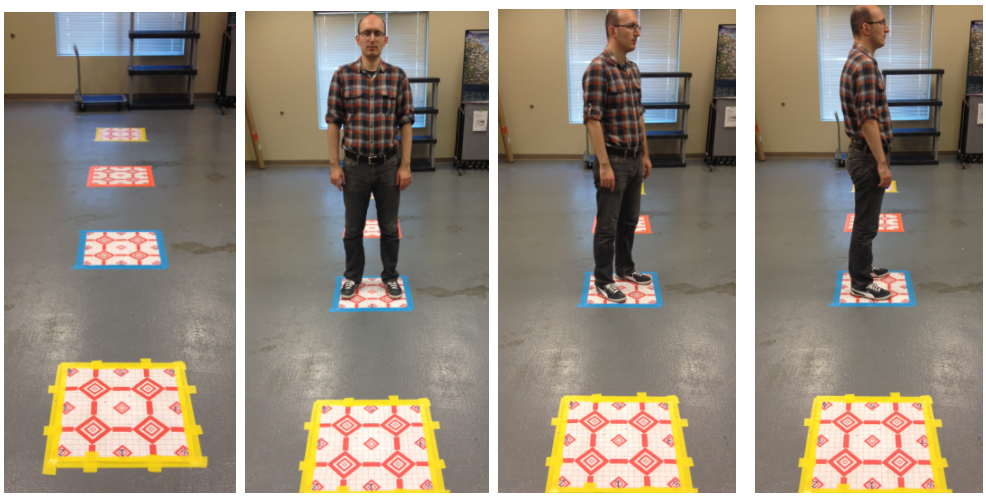

Figure 8: Experimental setup for the evaluation study of the Torso Detector.

Table 3 shows the angular error at every target distance and human orientation with respect to the laser scanner.

Distance To Laser	N	NE	E	SE	S	SW	W	NW	ALL
$1.0m$	4°	12°	22°	13°	5°	7°	26°	17°	13°
$2.5m$	5°	16°	19°	10°	3°	6°	14°	17°	11°
$4.0m$	4°	10°	30°	16°	7°	11°	21°	17°	15°
$5.5m$	5°	11°	41°	18°	10°	6°	38°	23°	19°
ALL	4°	12°	27°	14°	6°	7°	24°	18°	14.5°

Table 3: Average orientation error of the torso detector with respect to distance from sensor and body pose in a study with 23 people

The average positional error was about $5cm$ regardless of the distance and the

orientation of the human. The average orientation error throughout all the experiments was 14.5°. Error in orientation, however, varied greatly by pose of the person with respect to the laser scanner. Average error in orientation differed slightly with respect to the distance from the sensor and was the least with 11° when the humans were $2.5m$ away from the sensor. We attribute to the fact that when humans closer than $2.5m$ to the laser scanner, it captures more of the arms, which makes the fitted ellipse slightly worse. The orientation of the human with respect to the sensor had a significant effect on orientation error. Least error was achieved when people faced the sensor (4°) or the opposite way (6°). On the other hand, average orientation error was $24° - 27°$ when humans are perpendicular to the sensor, because a large portion of the torso is not visible to the laser scanner in that configuration.

4.3 Person State Estimation

The position and velocity of the person can not be determined by direct observation due to measurement noise and false detections. Therefore there is a need for a filtering algorithm in order to estimate the state of a person. Using a state predictor for human movement has two advantages. First, the predicted trajectories are smoother than raw detections. Smooth tracking helps the robot maintain consistent trajectories for high-level applications such as Person Following (Section 5). Second, it provides a posterior estimation that can be used for data association when there is a lack of matching detections. This allows the tracker to handle temporary occlusions. We use a discrete Kalman Filter [14] to predict the position of a person. There are other types of filtering techniques available in the literature, such as Particle Filters [15]. Since the results of the person state estimator is used by time-critical higher level applications, the tracker should come up with an estimate in real time. Therefore the choice of using Kalman Filters was motivated by its computational efficiency. Efficient person state estimation also increases the safety of the robot, as the robot

can react faster if there are people in close proximity.

Accoding to Hicheur [13], humans tend to maintain a constant speed when they are walking straight and reduce speed while turning. We used constant velocity model which assumes people will maintain their speed. Even though this assumption is not always true, it provides a simple model without sacrificing too much from tracking performance.

The Kalman filter estimates a process as a predictor-corrector cycle using feedback control. The process has two cycling states: time update and measurement update as shown in Figure. Time update projects the state forward by using the current state and error covariance. Measurement update is responsible for the feedback and corrects the previous estimate.

The Kalman Filter is governed by two linear stochastic difference equations:

$$s_k = As_{k-1} + Bu_{k-1} + w \tag{2}$$

$$z_k = Hs_k + v \tag{3}$$

Where s_k represents the process state at time step k, A is the state propagation matrix, B relates the optional control input u, z_k is a measurement, H is the measurement observation matrix. w and v represent the process and measurement noises, respectively, drawn from normal probability distributions with zero mean $N(0, Q)$ and $N(0, R)$.

We define the state of a person s_k at time step k as:

$$s_k = \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} \tag{4}$$

where (x_k, y_k) is the position and $(\dot{x}_k, \dot{y}_k)$ is the velocity of the person in Cartesian

Coordinates. With the constant velocity model, the time update equations are:

$$x_k = x_{k-1} + \dot{x}_{k-1}\Delta t_k + w \tag{5}$$

$$y_k = y_{k-1} + \dot{y}_{k-1}\Delta t_k + w \tag{6}$$

$$\dot{x}_k = \dot{x}_{k-1} \tag{7}$$

$$\dot{y}_k = \dot{y}_{k-1} \tag{8}$$

resulting in the following Kalman Filter matrices:

$$A = \begin{bmatrix} 1 & 0 & \Delta t_k & 0 \\ 0 & 1 & 0 & \Delta t_k \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{9}$$

where Δt_k is the time difference from the previous detection. A track is lost if there are no detections for a fixed amount of time. At every time update of a filter, if Δt_k is larger than a fixed threshold, the track is killed.

The reason B vector is zero is that we track people in the world frame and robot motion is already accounted for with robot localization. For this reason, we assume there are no control inputs to our system. The noise matrices we used are:

$$Q = qI_4 \qquad\qquad R = rI_2 \tag{10}$$

where we used $q = 0.02$ and $r = 1.0$ in practice.

Our approach is multimodal in the sense that asynchronous measurements are accepted from different sources as long as they provide a positional estimate in the respective sensor frames. Using the latest localization information, this position is converted to the world frame and then fed as a measurement to the active filters. We apply an additional layer of filtering to every detection before it is considered a measurement. We check if a new detection is in collision with the static map, and it if is in collision, we reject that particular detection. The check against the static

map is fast and helps reduce false positives in practice. We use Nearest Neighbor (NN) data association [2], which is a reasonable compromise between performance and computational cost.

Depending on the task, a single person or multiple people must be tracked. We examine each case below:

- **Single target tracking:** For some tasks, such as person following, dedicated tracking of a single specific user is required and tracking bystanders is not required for task success. In this case, our goal is to keep tracking the specific user, so we significantly relax the detection thresholds of the detectors. Even though doing so results in more spurious detections, we do not start more than a single track. This approach improves the tracking performance of a single person.

- **Multi-target tracking:** When the robot is navigating to a goal point with human bystanders, tracking multiple people at the same time is necessary. Moreover, losing track of a bystander would not be very detrimental to task success. We keep a separate Kalman filter for each tracked person. If a detection is matched to multiple filters, only the closest filter is associated with the detection and the other filters are considered to have no detections for that time step.

4.4 Face Recognition

For certain interactive navigation tasks such as finding a specific person, a robot needs to have person recognition capability. Our person recognition approach uses face recognition and optionally shirt color features. We detect faces in RGB images using the popular face detector by Viola and Jones [32]. We use the Eigenface method by Turk and Penland [30] for face recognition. Our approach allows new faces to be trained on-the-fly.

With the *Eigenface* approach, face are represented in a lower-dimensional space. Sirovich and Kirby [**?**] showed that dimension reduction method Principal Component Analysis (PCA) can be used on face images to form a set of basis features. The main idea of PCA for faces is to find vectors that best account for variation of face images in all training images. These vectors are called *eigenvectors*. Then a face space is constructed called *eigenfaces* and the images are projected onto this space. Our approach of face recognition works as follows:

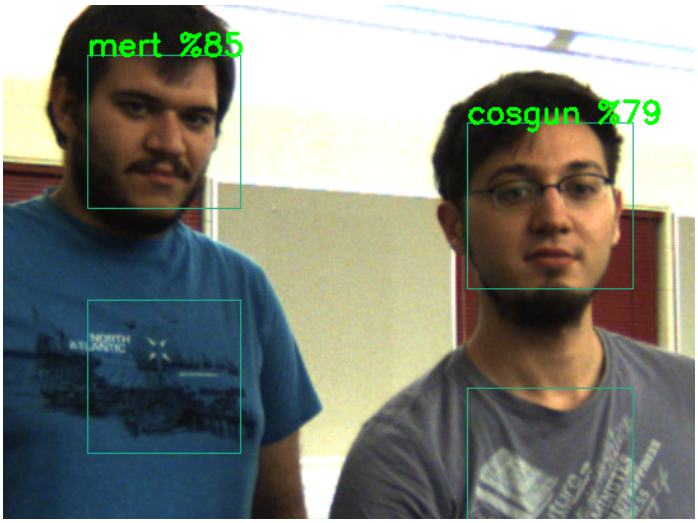

Figure 9: Example results of our person recognition method is shown in the image. We use *Eigenfaces* face recognition method and optionally shirt color recognition.

1. A person unknown to the system comes up to the robot and initiates training.

2. Robot asks the person to turn his face one side to another, and takes M face and shirt images of this person.

3. Eigenfaces from the entire training set is calculated, and every known face is projected to the corresponding M-dimensional weight *facespace*.

4. After training is completed, face recognition is reactivated.

5. A distance value from face recognition and optionally from shirt color recognition is received and it is thresholded for a decision. An example recognition

result is in Figure 9.

Using the UI of the robot, a user can start training and adjust the information in the person database. The person data is managed by a SQLite database hosted locally on the robot.

Shirt color recognizer can be used when there is little time between the training and recognition. Activating the shirt recognition should improve recognition and reduce false positive detections. We assume a rectangular region below the face captures the shirt (1.5 times below the the face rectangle size). The distribute the histogram into bins using normalized RGB color space because of its relative robustness to lighting. For detection, we calculate the distance between the training histogram to the test histogram using Earth Mover Distance [**?**]. The color histogram is adaptively updated at every high confidence detection in order to account for illumination changes. The overall person score is calculated by a weighted average of face and shirt distance.

CHAPTER V

PERSON FOLLOWING

Person Following

5.1 Related Work

Related Work

5.2 Basic Person Following

Basic Person Following

5.3 Situation Aware Person Following

Situation Aware Person Following

5.3.1 Door Passing

5.3.2 User Activity Awareness

5.3.3 Corners

5.4 Application To Telepresence Robots

Application To Telepresence Robots

CHAPTER VI

PERSON GUIDANCE

Person Guidance

6.1 Related Work

Related Work

6.2 Guide Robot

Guide Robot

6.3 Application To Blind Users

Application To Blind Users

CHAPTER VII

CONCLUSION

Conclusion

Table 4: A table, centered.

Title	Author
War And Peace	Leo Tolstoy
The Great Gatsby	F. Scott Fitzgerald

APPENDIX A

QR CODE BASED LOCATION INITIALIZATION

QR Code Based Location Initialization

APPENDIX B

ASSISTED REMOTE CONTROL

Assisted Remote Control

APPENDIX C

VIBRATION PATTERN ANALYSIS FOR HAPTIC BELTS

Vibration Pattern Analysis for Haptic Belts

REFERENCES

[1] Arras, K. O., Mozos, O. M., and Burgard, W., "Using boosted features for the detection of people in 2d range data," in *Robotics and Automation, 2007 IEEE International Conference on*, pp. 3402–3407, IEEE, 2007.

[2] Bar-Shalom, Y. and Li, X.-R., *Multitarget-multisensor tracking: principles and techniques*, vol. 19. YBS Storrs, Conn., 1995.

[3] Baumberg, A. and Hogg, D., "Learning deformable models for tracking the human body," in *Motion-Based Recognition*, pp. 39–60, Springer, 1997.

[4] Bellotto, N. and Hu, H., "Multisensor-based human detection and tracking for mobile service robots," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 39, no. 1, pp. 167–181, 2009.

[5] Bernardin, K. and Stiefelhagen, R., "Audio-visual multi-person tracking and identification for smart environments," in *Proceedings of the 15th international conference on Multimedia*, pp. 661–670, ACM, 2007.

[6] Buys, K., Cagniart, C., Baksheev, A., De Laet, T., De Schutter, J., and Pantofaru, C., "An adaptable system for rgb-d based human body detection and pose estimation," *Journal of Visual Communication and Image Representation*, 2013.

[7] Carballo, A., Ohya, A., and Yuta, S., "Fusion of double layered multiple laser range finders for people detection from a mobile robot," in *Multisensor Fusion and Integration for Intelligent Systems, 2008. MFI 2008. IEEE International Conference on*, pp. 677–682, IEEE, 2008.

[8] Fitzgibbon, A., Pilu, M., and Fisher, R. B., "Direct least square fitting of ellipses," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 21, no. 5, pp. 476–480, 1999.

[9] Ganapathi, V., Plagemann, C., Koller, D., and Thrun, S., "Real time motion capture using a single time-of-flight camera," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 755–762, IEEE, 2010.

[10] Garcia-Salicetti, S., Beumier, C., Chollet, G., Dorizzi, B., les Jardins, J. L., Lunter, J., Ni, Y., and Petrovska-Delacrétaz, D., "Biomet: a multimodal person authentication database including face, voice, fingerprint, hand and signature modalities," in *Audio-and Video-Based Biometric Person Authentication*, pp. 845–853, Springer, 2003.

[11] Germa, T., Lerasle, F., Ouadah, N., and Cadenat, V., "Vision and rfid data fusion for tracking people in crowds by a mobile robot," *Computer Vision and Image Understanding*, vol. 114, no. 6, pp. 641–651, 2010.

[12] Glas, D. F., Miyashita, T., Ishiguro, H., and Hagita, N., "Laser-based tracking of human position and orientation using parametric shape modeling," *Advanced robotics*, vol. 23, no. 4, pp. 405–428, 2009.

[13] Hicheur, H., Vieilledent, S., Richardson, M., Flash, T., and Berthoz, A., "Velocity and curvature in human locomotion along complex curved paths: a comparison with hand movements," *Experimental brain research*, vol. 162, no. 2, pp. 145–154, 2005.

[14] Kalman, R. E., "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[15] Khan, Z., Balch, T., and Dellaert, F., "An mcmc-based particle filter for tracking multiple interacting targets," in *Computer Vision-ECCV 2004*, pp. 279–290, Springer, 2004.

[16] Kinnunen, T. and Li, H., "An overview of text-independent speaker recognition: From features to supervectors," *Speech communication*, vol. 52, no. 1, pp. 12–40, 2010.

[17] Kirchner, N., Alempijevic, A., and Virgona, A., "Head-to-shoulder signature for person recognition," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 1226–1231, IEEE, 2012.

[18] Kleinehagenbrock, M., Lang, S., Fritsch, J., Lomker, F., Fink, G. A., and Sagerer, G., "Person tracking with a mobile robot based on multi-modal anchoring," in *Robot and Human Interactive Communication, 2002. Proceedings. 11th IEEE International Workshop on*, pp. 423–429, IEEE, 2002.

[19] Leibe, B., Seemann, E., and Schiele, B., "Pedestrian detection in crowded scenes," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 878–885, IEEE, 2005.

[20] Mitzel, D. and Leibe, B., "Close-range human detection for head-mounted cameras," in *British Machine Vision Conference (BMVC)*, 2012.

[21] Moeslund, T. B. and Granum, E., "A survey of computer vision-based human motion capture," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 231–268, 2001.

[22] Montemerlo, M., Thrun, S., and Whittaker, W., "Conditional particle filters for simultaneous mobile robot localization and people-tracking," in *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, vol. 1, pp. 695–701, IEEE, 2002.

[23] Munsell, B. C., Temlyakov, A., Qu, C., and Wang, S., "Person identification using full-body motion and anthropometric biometrics from kinect videos," in *Computer Vision–ECCV 2012. Workshops and Demonstrations*, pp. 91–100, Springer, 2012.

[24] Phillips, P. J., Flynn, P. J., Scruggs, T., Bowyer, K. W., Chang, J., Hoffman, K., Marques, J., Min, J., and Worek, W., "Overview of the face recognition grand challenge," in *Computer vision and pattern recognition, 2005. CVPR 2005. IEEE computer society conference on*, vol. 1, pp. 947–954, IEEE, 2005.

[25] Schulz, D., Burgard, W., Fox, D., and Cremers, A. B., "Tracking multiple moving targets with a mobile robot using particle filters and statistical data association," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 2, pp. 1665–1670, IEEE, 2001.

[26] Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., Cook, M., and Moore, R., "Real-time human pose recognition in parts from single depth images," *Communications of the ACM*, vol. 56, no. 1, pp. 116–124, 2013.

[27] Sidenbladh, H., Black, M. J., and Fleet, D. J., "Stochastic tracking of 3d human figures using 2d image motion," in *Computer Vision?ECCV 2000*, pp. 702–718, Springer, 2000.

[28] Spinello, L., Arras, K. O., Triebel, R., and Siegwart, R., "A layered approach to people detection in 3d range data," in *AAAI Conf. on Artif. Intell.(AAAI)*, 2010.

[29] Topp, E. A. and Christensen, H. I., "Tracking for following and passing persons," in *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pp. 2321–2327, IEEE, 2005.

[30] Turk, M. A. and Pentland, A. P., "Face recognition using eigenfaces," in *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pp. 586–591, IEEE, 1991.

[31] Tuzel, O., Porikli, F., and Meer, P., "Human detection via classification on riemannian manifolds," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pp. 1–8, IEEE, 2007.

[32] Viola, P. and Jones, M. J., "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[33] Xavier, J., Pacheco, M., Castro, D., Ruano, A., and Nunes, U., "Fast line, arc/circle and leg detection from laser scan data in a player driver," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 3930–3935, IEEE, 2005.

[34] Yan, P. and Bowyer, K. W., "Biometric recognition using 3d ear shape," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 8, pp. 1297–1308, 2007.

[35] Zhang, C. and Zhang, Z., "A survey of recent advances in face detection," tech. rep., Tech. rep., Microsoft Research, 2010.

[36] Zhao, W., Krishnaswamy, A., Chellappa, R., Swets, D. L., and Weng, J., "Discriminant analysis of principal components for face recognition," in *Face Recognition*, pp. 73–85, Springer, 1998.

INDEX

VITA

Perry H. Disdainful was born in an insignificant town whose only claim to fame is that it produced such a fine specimen of a researcher.

People Aware Mobile Robot Navigation

Akansel Cosgun

33 Pages

Directed by Professor Henrik Christensen

This is the abstract that must be turned in as hard copy to the thesis office to meet the UMI requirements. It should *not* be included when submitting your ETD. Comment out the abstract environment before submitting. It is recommended that you simply copy and paste the text you put in the summary environment into this environment. The title, your name, the page count, and your advisor's name will all be generated automatically.