Autonomous Person Following for Telepresence Robots

Akansel Cosgun, Dinei A. Florencio and Henrik I. Christensen

Abstract— **We present a method for autonomous person following in office-like environments for a telepresence robot. Our approach takes into account the dynamicity of the person and plans the motions of the robot by searching future trajectories. Instead of determining goal points near the person, we introduce a task dependent goal function which provides a map of desirable areas for the robot to be at, with respect to the person. The planning framework is flexible and allows to encode different social conventions for tasks that require spatial interactions. We conduct a small controlled user study to compare the autonomous method to teleoperation for following a person while having a conversation. By designing a behavior specific to a flat screen telepresence robot, we show that the person following behavior is perceived as safe and socially acceptable by remote users. All 10 participants preferred our autonomous following method over manual teleoperation.**

I. Introduction

Mobile telepresence robots constitute a promising area for the robotics industry, and early systems are already commercially available from companies such as Vgo Communications, Anybots, and Double Robotics. These are small scale deployments, and a number of issues need to be addressed before wide deployment, many posing interesting technical challenges. These challenges include designing more intuitive user interfaces [1, 2], better video conferencing [3, 4], better audio capture [5, 6], how the remote user's presence is displayed [7], overcoming wireless connection restrictions [8, 9], assisting teleoperation [10, 11] and adjusting the level of autonomy [11, 12]. Telepresence robots are a level above video conferencing since the robot is used as the communication medium and the remote user can now control the movement. Therefore, the spatial interaction between people and a telepresence robot in social situations is worth investigating. One of those situations is moving with a group of people. In an effort to analyze the spatial and verbal interaction, we focus on engagement with one person where the remote user interacts with the person while following him/her in a corridor. This is situation is very likely to happen in office environments, for example when the remote user is having a discussion with a co-worker while walking to his office after a meeting. As telepresence robots become more common, there will be need to have the functionality of autonomous following of a person so that the remote user doesn't have to worry about controlling the robot.

In this paper, we propose a planning framework for autonomous person following. As opposed to traditional motion planning, our approach does not determine explicit goal states with respect to the person but achieves the desired behavior by searching for the best utility over time. By doing so, we can account for moving targets and design the behavior of the system by adjusting the cost function coefficients and the goal function. The framework considers the future trajectory of the followed person and allows the robot to exhibit different following behaviors with the help of the goal function defined for the specific task. The robot behavior is dependent on the location of the person in relation to the robot, distance to obstacles as well as the velocity and acceleration of the robot.

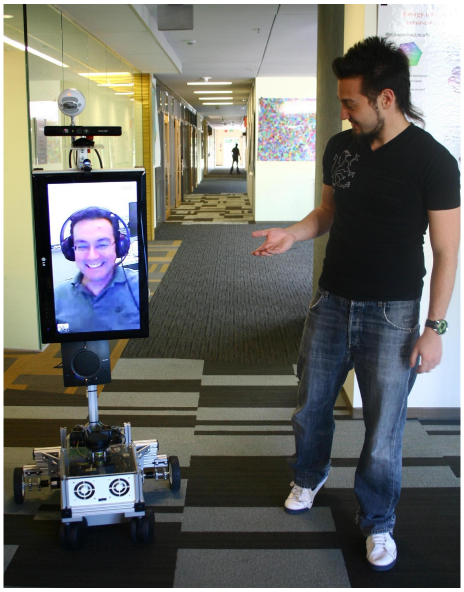

Fig. 1: Remote user and the followed person walking together while having a conversation.

We evaluate our system by first showing quantitative results for the performance and then conducting a user study. The user study aims to compare manual and autonomous person following when the remote user has a task at hand that involves interaction. The task consists of listening to a passage the followed person reads and answering related questions afterwards. We also observe subjects' experiences using the system, get useful feedback and pinpoint future challenges that can be helpful designing new applications for telepresence robots.

First, we examine the relevant literature in Section II including person following, social navigation and telepresence systems. The robot platform used in this work is described in Section III. Section IV describes our algorithm in detail. Section V presents our evaluation and the user study and we discuss the results in Section VI.

A. Cosgun is a PhD student at the Center for Robotics and Intelligent Machines, Georgia Tech, Atlanta, GA.

H. Christensen is the KUKA Chair of Robotics and the director of Center for Robotics and Intelligent Machines at Georgia Tech, Atlanta, GA.

D. Florencio is a researcher with Microsoft Research, Redmond WA.

II. Related Work

The problem of following a person with a mobile robot has found a lot of interest in the literature. However, most existing work do not consider the interaction between the robot and the followed person.

One of the methods to track people is to detect legs in laser scans [13–18]. Leg tracking has been a popular method since for most robots, the laser scanners are placed at ankle level for navigation. One other common method to detect people is to use face or color blob detection [19] and fusing it with laser-based methods [20, 21]. More recently, depth cameras have been used to detect and track people [22]. We use a RGB-D camera to select the person to be followed and then track the legs of that person in laser scans.

Some of the related work demonstrated the behavior of accompanying a person. In [23], a robot that escorts a person by his/her side while avoiding obstacles is mentioned. Miura [24] employs randomized tree expansion and biases the paths towards a sub-goal which is the current position of the person. Prassler [25] uses the predicted position of the person in the next frame as a virtual moving target. Although the robot was a wheelchair, the social issues that might arise by having a person sitting in the wheelchair were not considered. Hoeller [26] adopts the virtual targets idea and selects a goal position in a circular region around the person. They perform randomized search for sequences of velocity commands with the help of an informed heuristic in order to reach intermediate goals as the person is moving. Path planning is a traditional and still quite active research area, with an extremely rich literature. However, most of the literature assumes a fixed, specified goal position. In contrast, we do not determine goal positions and our search is uninformed.

Gockley [27] observed how older adults walk together. It is reported that partners who were conversing tended to look forward with occasional glances to each other. People navigate obstacles and bottlenecks in a socially acceptable manner by either taking initiative or waiting for the other partner to lead. In [13], they showed that for person following, people have found direction-following behavior more natural than the path-following behavior. Some researchers worked on social navigation to a goal location in presence of people, using *proxemics* [15, 18, 28, 29]. Althaus [30] developed a reactive behavior to join a group of people and slightly move within the group while interacting. Svenstrup [17] describes a user study where a robot follows random people in an urban transit area. Most people were cooperative and positive towards to the robot. Loper [22] presents a system that is capable of following a person and responding to verbal commands and non-verbal gestures. This work exhibits interaction and demonstrates the robot and the human being in the same team, however it is fundamentally different from ours because there is human intelligence behind the telepresence robot.

There is recent interesting work related to telepresence. Venolia [3] shows that deploying telepresence systems has facilitated social integration and improved in-meeting interaction in workplace settings. Tsui [31] makes an attempt to introduce quantitative and qualitative measures to the quality of interaction for telepresence robots. In [32], customers' long term experiences on a telepresence robot are surveyed. It is observed that most users' attention was divided between driving the system and carrying on a conversation. Similar observations are also reported in clinical interactions [33]. It is mentioned in [34] that reducing the operator's responsibility will improve the usefulness of teleoperated robots in navigation tasks. An assistive control system to reduce collisions for direct teleoperation is presented in [11]. The choice of user interface for direct teleoperation also affects the situation awareness and cognitive workload of the remote users [34, 35].

Desai [12] conducts user studies on two of the commercially available telepresence units. It is said that 20 out of 24 participants thought a 'follow the person' autonomy mode will be useful for a telepresence robot. It is hypothesized that this behavior will allow the remote user to dedicate his/her attention to the conversation. We use our autonomous person following method to investigate this claim.

III. Platform

The system described in this paper is implemented on an experimental telepresence robot shown in Figure 1. The robot has a differential drive base and can be used for about 8 hours with full charge. For the experiments in this paper, the speed of the robot was limited to 0.55 m/s. A laser scanner with 360° field of view, which was taken from Neato XV-11 vacuum cleaning robot, was mounted horizontally at 0.3m height. The system runs on Windows 7 and Robotics Developer Studio (MRDS) as its distributed computing environment. On the remote end, standard webcam and headphones are used. The remote user connects to the robot via wireless internet and communicates using Skype. Omni-directional Blue Snowball Microphone and ClearOne Speakerphone are placed on the robot to provide a good voice quality.

There are two operation modes for the robot: Teleoperation and Autonomous Person Following. A Xbox 360 Wireless Controller is used to remotely teleoperate the robot. A wide-angle MS LifeCam is placed on top of the monitor and tilted slightly downward to help the remote user to see the floor, robot base and people's faces at the same time, as it is suggested in [35]. A Kinect Sensor is also placed above the monitor. Person following is initiated through the user interface (Figure 2).

IV. Autonomous Person Following

A. Person tracking

The remote operator initiates the person following behavior by clicking on a person in the depth image from the Kinect sensor. On the UI, a depth pixel is displayed green if the Kinect retrieves a valid distance value for that pixel; else it is displayed red (Figure 2b). Whenever the user clicks on a green pixel, we try to locate the head position

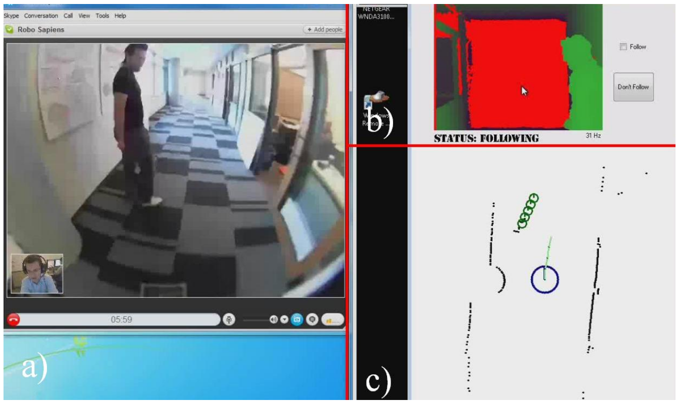

Fig. 2: UI as seen by the remote user. a) Wide lens camera image in Skype. b) Depth image from the Kinect. User can click on a person to start person following. c) Laser scan and the planned path.

of the person. First, all the pixels belonging to that person is found by region growing. Second, the headtop position is determined by finding the pixel with the greatest z value in world coordinates. We have observed that human hair returns noisy depth data from the Kinect. Therefore, we find the center of the head using the headtop pixel. The height of the head center in world coordinates is checked in order to rule out the cases where the user clicked on an object or a wall. This point is then projected to the horizontal plane at the height of the laser scanner. If a leg is detected in the laser scan in the vicinity of that point, then leg tracking is activated. After the initial detection of the leg, it is tracked until the the remote user decides to stop following or the robot loses track of the person.

The first step in the leg detection process is to segment the laser data. Two adjacent distance measurements are considered to be in the same segment if the Euclidean distance between them is below a threshold. In a laser scan, legs can appear in different patterns [14]: *Single leg (SL)*, *Two legs appropriately separated (TL)* and *Person-wide blob (PW)*. Three geometric features of a segment are calculated for leg detection: Segment Width, Circularity and Inscribed Angle Variance (IAV) [36]. We captured several laser frames when the robot is mobile and manually labeled the segments with associated leg patterns. About 1.7×10^4 *SL*, 600 *PW* and 1.2×10^5 *Not Leg* segments were captured. For the training set, two people's legs are recorded with different clothing (i.e. shorts, baggy pants, trousers). The average and variances of leg features are given in Table I.

Given a segment $S = (f_1, f_2, f_3)$ where f_n is the value of leg feature n, a variant of Mahalanobis Distance from the average leg is calculated by: $D_M = \sum_{i=1}^{3} k_i \frac{(f_i - \mu_i)^2}{\sigma_i^2}$, where k_i are weights for the parameters. D_M is compared with a fixed threshold, do determine if the segment is a leg or not. While a leg is being tracked; a fourth parameter, the proximity of the segment center to the estimated position of the leg, is also considered. The leg is tracked in the odometry frame using a single hypothesis and constant velocity model. The segment with the least overall Mahalanobis distance is considered a match. When no segment is below the leg distance threshold, the tracker expects the leg to reappear in the same location for some time, before declaring the person is lost. This allows handling no-detection frames and temporary occlusions (i.e. another person passes between the tracked person and robot). We look for a *Single Leg (SL)* pattern for the initial detection and track only one leg afterwards. The mean and variances of leg features are adaptively updated using the observations for the last 4 seconds, so that the tracker adjusts to the leg shape/clothing of the person after the initial detection. The leg position is considered as the position of the person and is fed to the motion planner.

	Width(m)		Circularity		IAV(rad)	
Pattern	μ	σ	μ	σ	μ	σ
Single Leg	0.13	0.03	0.25	0.15	2.23	0.40
Personwide blob	0.33	0.07	0.14	0.09	2.61	0.16
Not Leg	0.22	0.12	0.1	0.11	2.71	0.38

TABLE I: Mean and variances of geometric leg features from the training set

B. Motion Planning

Traditional motion planners require a goal state, however for person following, since the goal is moving at every frame, the completely specified plan at any time instance would never be valid after the execution of the first action. To account the dynamicity of the target, we define cost and goal functions and search for the best utility in a limited time. Our approach is similar to one of the earliest local navigation schemes Dynamic Window Approach (DWA) [37]. DWA forward-simulates the allowable velocities and chooses an action that optimizes a function that will create a goal-directed behavior while avoiding obstacles. We extend DWA, to follow a dynamic goal and plan for the future steps instead of planning for just the current time slice.

Our planner takes the laser scan, predicted trajectory of the person and the number of time steps as input and outputs a sequence of actions. A robot and person configuration at time t is expressed as $q^t = (x^t, y^t, \theta^t, v^t, \omega^t)^{\mathrm{T}}$, where x^t and y^t denote positions, θ^t is the orientation, v^t and w^t are the linear and angular velocities at time t. The person configuration p^t is defined the same way. An action of the robot is defined as a velocity command for some duration: $a(t, \Delta t) = (v_a^t, \omega_a^t, \Delta t)$. Robot motion model we use is adopted from [26]:

$$\begin{aligned} q^{t+\Delta t} &= f(q^t, a(t, \Delta t)) \\ &= \begin{pmatrix} x^t - \frac{v_a^t}{\omega_a^t} sin(\theta^t) + \frac{v_a^t}{\omega_a^t} sin(\theta^t + \omega_a^t \Delta t) \\ y^t + \frac{v_a^t}{w_a^t} cos(\theta^t) - \frac{v_a^t}{\omega_a^t} cos(\theta^t + \omega_a^t \Delta t) \\ \theta^t + \omega_a^t \Delta t \\ v_a^t \\ \omega_a^t \end{pmatrix} \end{aligned}$$

Using this model, we generate a tree up to a fixed depth, starting from the current configuration of the robot. A tree node consists of a robot configuration as well as the information about the previous action and parent node. Every depth of the tree corresponds to a discretized time slice. Therefore, every action taken in the planning phase advances the time by a fixed amount. This enables the planner to consider future steps of the person and simulate what is likely to happen in the future. The planner uses depth-limited Breadth First Search (BFS) to search all the trajectories in the generated tree and determines the trajectory that will give the robot the maximum utility over a fixed time in the future. See Algorithm 1 for the pseudocode. When a node is being expanded, first the feasible actions from configuration q^t are found (Line 7). The feasible actions are calculated using the acceleration and velocity limits. Note that stopping action $a(t, \Delta t) = (0, 0, \Delta t)$ is always allowed so the robot may choose to stop or wait instead of moving. Typically, the number of possible actions is too high, and if all of them were expanded, it would quickly make the search intractable. Therefore, the expanded nodes are sorted according to their utility values (Line 13) and only the best b_{max} of the expanded nodes are added to the tree (Line 20). This allows expansion of a maximum of b_{max} nodes from a parent node, limiting the effective branching factor to b_{max}. Note that b_{max} and d_{max} are fixed parameters. The leaf nodes at depth d_{max} are candidates to be the solution node. The node at depth d_{max} which yields the maximum utility (Line 17) is back-traced up to the start node $q^{t=0}$ and returned as the solution (Line 4). Since the search is depth-limited BFS, all nodes with depth $d \leq d_{max}$ will be explored. The solution consists of a sequence of actions that the robot should apply for the next d_{max} time steps with Δt intervals. Since the task of the robot is to accompany a moving person, the utility functions change at every time frame so the plan is re-calculated every time a new laser scan observation is received. The complexity of the algorithm is $O((b_{max})^{d_{max}})$.

Algorithm 1 *plan*$(q^{t=0}, b_{max}, d_{max}, \Delta t)$

```
 1: $Q$.enqueue$(q^{t=0})$
 2: loop
 3:   if $Q$.empty() then
 4:     return $Backtrace(q_{best})$
 5:   end if
 6:   $q^t \leftarrow Q$.dequeue()
 7:   $A \leftarrow$ getAvailableActions$(q^t)$
 8:   List $L$
 9:   for $a_i \in A$ do
10:     $q^{t+\Delta t} \leftarrow$ expandNode$(q^t, a_i)$
11:     $L$.append$(q^{t+\Delta t})$
12:   end for
13:   $L$.sort($L$[:].getUtility())
14:   for $j = 0 : b_{max}$ do
15:     if $L[j].depth == d_{max}$ then
16:       if $L[j]$.getUtility()>$q_{best}$.getUtility() then
17:         $q_{best} = L[j]$
18:       end if
19:     else
20:       $Q$.enqueue$(L[j])$
21:     end if
22:   end for
23: end loop
```

The *getUtility* function in Line 16 of Algorithm 1 returns the total utility U of a node that is the discounted summation of instantaneous utilities of all intermediate nodes in a branch from the start node to the queried node. Total Utility U^{t_n} of a node at time slice $t = t_n$ is given as:

$$U^{t_n} = \sum_{t=0}^{t_n} \beta^t u_t(q^{t-\Delta t}, q^t, p^t, C^t) \text{ where}$$

$$u_t(q^{t-\Delta t}, q^t, p^t, C^t) = w_g g(q^t, p^t) + w_o(1 - c_o(q^t, C^t)) + w_a(1 - c_a(q^{t-\Delta t}, q^t)) + w_v(1 - c_v(q^t))$$

p and q represents the person and robot configurations respectively. u_t is the instantaneous utility of a node, C^t is the configuration space at time t. Node configurations colliding C-space obstacles are ruled out. Laser hits corresponding to the person are removed from the laser scan and are not considered for C-space calculation. This allows us to use the same C-space obstacles when we are expanding the nodes and planning for the future. $0 < \beta < 1$ is a scalar so it can be interpreted as a discounting factor. β determines how much importance are given to the future steps. A low β will lead to a more reactive behavior. $0 \leq c_o(C^t) \leq 1$ is the obstacle cost and is acquired from the 2D obstacle costmap that is created using the laser scan. The obstacle cost gets closer to 1 as the robot position gets closer to the configuration space obstacle. $0 \leq c_a(q^{t-\Delta t}, q^t) \leq 1$ is the acceleration cost. It punishes rapid velocity changes and helps to choose smoother motions. $c_v(q^t)$ is the velocity cost. It punishes the configurations with a non-zero angular velocity so it encourages the robot to choose straight trajectories instead of arcs. w's are associated weights for costs and they sum up to 1.

Given the robot and person configuration at some particular time, goal function $0 \leq g(q^t, p^t) \leq 1$ determines how desirable the spatial joint configuration is. Goal function can be defined in any way and provides flexibility to the designer of the behaviors that the robot exhibits. Figure 3 illustrates the goal function for a fixed robot configuration $q^t_{fixed} = (0, 0, \pi/2, v^t_r, \omega^t_r)^{\mathrm{T}}$ and variable person configurations $p^t = (x_p, y_p, \theta_p, v^t_p, \omega^t_p)^{\mathrm{T}}$, represented in robot's local coordinate frame. It shows the overhead view of the floor plane when the robot is thought at the origin heading up. From the robot's perspective, the brightness signifies how much utility would be earned if the person was at x_p and y_p. In other words, when the robot is considered at the origin, it gets higher utility over time when the person is in the whiter regions. For example, if the person is at $(x_p, y_p) = (-0.8, 0.8)$ (green mark); $g(q^t_{fixed}, p^t) = g(q^t_{fixed}, (-0.8, 0.8, \theta_p, v^t_p, \omega^t_p)) = 1$ and it is a desirable position. Similarly when the person is 1.2m on the left of the robot, $g(q^t_{fixed}, (-1.2, 0.0, \theta_p, v^t_p, \omega^t_p)) = 0.3$ (yellow mark) and the robot earns some utility even though it is not the most desired location to be at.

We have chosen to define a robot-centric goal function instead of person-centric because our estimation of the

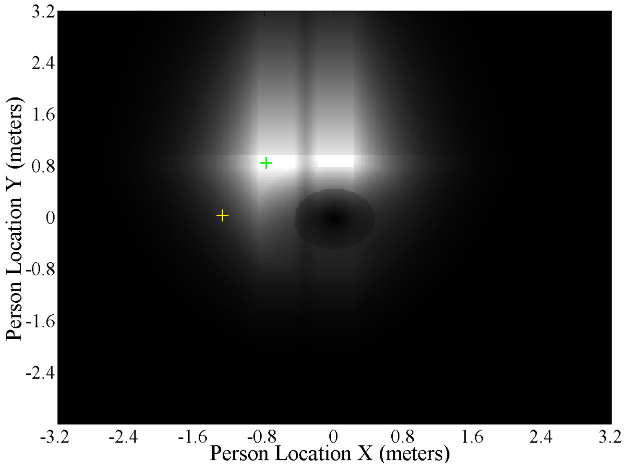

Fig. 3: Goal function $g(q^t, p^t)$. The figure shows the overhead view, when the robot is at (0,0) and heading up. The robot earns higher utility over time when the person is in whiter regions.

person orientation and velocity is not as accurate as our estimates of the robot's. Moreover, people tend to change their orientations more frequently than the robot. The shape of this goal function tells that the robot wants to keep the person two lanes: in front or front-left, which means the robot will try to stay behind the person and possibly to his right. We assume that it is desirable for the robot to trail from behind since it is the only way to have a face-to-face interaction for a flat screen monitor. Moreover, following from right would likely to give the feeling of 'walking together'. We discourage the robot to change sides frequently by having only two local minima, with the assumption that people would feel uncomfortable if the robot motions are unpredictable. Getting too close to the person is also discouraged by having a low utility region around the origin. Other researchers used different cost functions to account for the personal spaces [18, 28].

V. Evaluation

The parameters for our autonomous following algorithm in the experiments are: Leg tracking: $k_{width} = 0.13$, $k_{circ} = 0.16$, $k_{IAV} = 0.06$, $k_{proximity} = 0.65$. Planner: $w_g = 0.42$, $w_o = 0.22$, $w_a = 0.14$, $w_v = 0.22$, $b_{max} = 10$, $d_{max} = 4$, $\beta = 0.9$, goal function in Figure 3.

We first evaluate our method quantitatively by running the autonomous person following on different people and validate our approach. The robot is used without interaction and there is no remote user behind the robot. In the second experiment, we conduct a user study to compare teleoperation vs. autonomous following when the subjects are remote users.

For manual teleoperation, the remote user controls the robot using the thumbstick of an Xbox 360 controller. We have linearly mapped the horizontal axis of the thumbstick to the angular velocity and vertical axis to the linear velocity. For example, a half throttle on the vertical axis commands the robot to move with half of the maximum linear velocity. A simple collision prevention was employed to make the teleoperation safer. Whenever the robot receives a control command, it looks to the closest obstacle in the last observed laser scan. Depending on how close the objects are, the input velocity magnitudes are reduced. This functionality helps to reduce the number and impact of collisions. The robot stops if no user input is received for 5 seconds. This safety measure prevents the robot from continuing with the same velocity in the event of a lost connection.

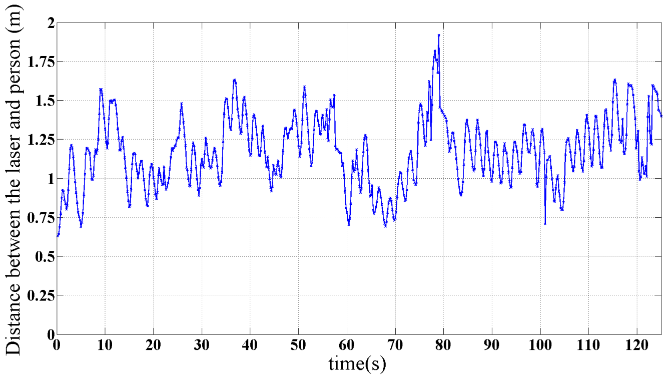

Fig. 4: Distance between the followed person and the laser scanner as a function of time for a sample run.

A. Results

We have asked 7 people who didn't have prior experience with the system to walk in a corridor while the robot is following them. A lap consisted of leaving the starting point, going to an intermediate point at the end of the corridor and coming back to the starting point from the same path. The robot followed every subject for 3 laps. There was one significant turn in the course and turning back from the same path counter-balanced the number of left and right turns. When the robot lost track of the person due to the fast walking pace, the subject was verbally notified and the experimenter restarted the person tracker. We logged the total distance the robot traversed and distance between the robot and the followed subject (Table II). The average distance between the person and the robot across all 7 runs was 1.16m.

Subject	Total Dist (m)	Avg Following Dist (m)
1	171.4	1.1
2	161.8	1.13
3	160.4	1.14
4	169.9	1.25
5	174.2	1.04
6	166.2	1.3
7	171.5	1.2

TABLE II: Total distance covered by each subject on the run and average following distance

Figure 4 plots the time following distance as a function of time for a sample run that consists of 1 lap. At t=0, the following is initiated and robot leaves the starting point. Around t=8s, the robot and the person start making a right turn. The sudden drop at t=57.2s signifies that the robot lost track of the person and the person tracker is reinitialized. At t=65.6s, the intermediate point at the end of the corridor is reached so the person makes a 180° turn. The robot is close to the person (about 0.75m) around this time because it is just rotating around place while the person is turning back. Between t=70-80s, the person is faster than the robot and the distance to the person reaches to a maximum of 1.91m. At t=79.8s, the person is lost again. Around t=115s, the robot and the person make a left turn. The lap ends at t=124s. Note that the person distance as a function of time isn't smooth. This is a result the walking pattern because only one leg is tracked and the leg velocity shows oscillatory behavior even if the person walks at constant speed.

B. User Study

In this study, remote user is the subject and the followed person is the experimenter. To investigate the effectiveness of using autonomous person following for an interaction task, we ran a controlled experiment and varied manual vs. autonomous following within subjects.

a) Design: The study was conducted in the same corridor used in Section V-A. The experiments were conducted in working hours and bypassers were allowed to walk across the experiment area or talk. The subjects were given the task of following the experimenter through the course for a lap and listen to the passage he is reading. In the first run, the subject used the autonomous following or teleoperation method to follow a person and complete the lap. In the second run, the subject used the other method. At the end of each run, the subject was asked to complete a 4-question quiz about the passage. The passages and quiz questions were taken from Test of English as a Foreign Language (TOEFL) listening section examples. One passage was about "behaviorism" and the other one was about "manila hemps", and passages were chosen so that they are at a similar difficulty level. The time it takes to read a passage corresponded approximately to the same time a lap is completed. We also asked numbered 7 point Likert scale questions, administered after each run, about how *Understandable* the experimenter was, *Easiness of UI*, if the robot exhibited *Natural Motions*, how *Safe* the remote user felt, if the subject was able to *Pay Attention* to the passage, how *Fast* the robot was and how much *Fun* the subject had. At the end of both runs, the user was asked which method he/she will prefer over the other for this type of a scenario.

b) Participants: 10 volunteers participated in the study (6 male and 4 female between the ages of 25-48). Participants consisted of 4 researchers and 6 interns at Microsoft Research. 5 of the participants had little knowledge, 4 had average knowledge and 1 had above average knowledge on robotics. The participants weren't gamers: 4 participants never played console games, 4 played rarely, 1 sometimes played and 1 often played. 6 of the participants often used video conferencing software, while 2 sometimes and 2 rarely used. 9 of the participants were not native English speakers and all of them had taken the TOEFL before. Participants were recruited through personal relations and were given a small gift (valued at approximately US$10) for their help.

c) Procedure: The participants were first greeted by the experimenter and instructed to complete a pre-task questionnaire regarding their background. The robot was shown to the participant and basic information about its capabilities was told. The experimenter explained the task while walking with the participant in the corridor and showing the course to be followed. Participants were told that they should stay close to the experimenter while he is walking and there will be a quiz regarding the passage afterwards. The participant was informed that there are 2 operation modes: manual and autonomous following. Before the experiment started, the participant went through training for about 15 minutes. First, the participant learned the basic controls for the Xbox controller when he/she was nearby the robot. Then the participant was taken to the remote station, which was in a room about 20 meters away from the corridor area. The participant was informed about the UI and was shown how the autonomous following can be activated. Then a test run was executed, where the remote user followed the experimenter via teleoperation and had a conversation.

After the training, the actual run was executed using either the manual or autonomous method. When the lap was completed, first the passage quiz, then the survey questions were answered by the subject. Then the second experiment using the other method was executed, and the second passage quiz and survey questions were given to the subject. As the last question, the subject was asked to state his/her method of preference. Lastly, the participants were debriefed about the study and engaged in a discussion. We switched the starting method for every other experiment in order not to bias the subjects' opinions about one particular method.

In all experiments, the followed person was the experimenter. Having confirmed the validity of our approach in Section V-A and by fixing the walking behavior of the followed person, we could measure remote users' experience in a controlled fashion.

d) Measures: We had three measurement criteria to compare manual vs autonomous following: 1) Number of correct answers to passage quizzes: Assuming the standardized TOEFL exercises were of same difficulty, we ran a paired t-test on two groups of autonomous and manual. 2) Survey questions: We ran a paired t-test using 7-point Likert Scale on each of the seven questions. 3) Preferred Method: We looked at which method subjects chose over the other one.

e) Results: Out of 4 quiz questions, the correct answers for autonomous group (M=2.9, SD=.9) were more than the manual group (M=2.2, SD=1.2) but the statistical difference was not very significant (t(9)=1.48, p=.17 on t-test).

Table III summarizes the survey results. For *Understandable* and *Fun*, the scores slightly favored autonomous method but the difference wasn't statistically significant. Manual method User Interface (gaming controller) was found to be easy to use (5.0, SD=2.2), but the UI for autonomous method (clicking) was found to be marginally easier (6.5, SD=.9), (t(9)=2.13, p=.06). The motions of the robot was found to be significantly more *Natural* to have a conversation for autonomous (5.4, SD=1.0) than manual (3.5, SD=1.9), (t(9)=2.52, p=.03). Participants thought they were able to *Pay more Attention* to the passage the experimenter is reading

	Autonomous		**Manual**		*t*-test	
Question	μ	σ	μ	σ	p	t
1. Understandable	4.0	1.5	3.6	1.7	0.47	0.73
2. Easy UI	6.5	0.9	5.0	2.2	0.06	2.13
3. Natural motion	5.4	1.0	3.5	1.9	0.03	2.52
4. Safety	5.1	1.7	2.3	1.4	0.01	3.09
5. Pay attention	5.3	1.8	3.4	1.5	0.02	2.63
6. Fast	3.9	0.3	4.3	0.8	0.10	-1.80
7. Fun	5.3	1.5	5.1	1.7	0.66	0.45

TABLE III: Survey Results comparing Autonomous vs Manual Methods

when the robot was following the him autonomously (5.3, SD=1.8) compared to manual control (3.4, SD=1.5) and the statistical difference was significant (t(9)=2.63, p=.02). Participants have found the autonomous method (5.1, SD=1.7) much safer than manual method (2.3, SD=1.4) and there was a significant difference between two groups (t(9)=3.09, p=.01). The speed of the robot was found to be neither fast nor slow for both methods (3.9, SD=.3) and (4.3, SD=.8).

All 10 subjects chose autonomous person following over teleoperation for this task.

VI. Design Implications and Discussion

Our user study showed that a person following behavior is desirable for telepresence robots when there is interaction. The follow-up discussions also agreed with the survey results, as one subject (R10) stated: *"It just gives me more focus and concentration."* Below, we list our observations and implications for future research and design for telepresence robots:

Motor Noise: Even though the motors on the robot were relatively quiet, 8 out of 10 participants expressed that the motor noise made communication harder. This justifies the close scores we collected in the survey question asking if the subject was able to understand what the experimenter was saying. (R8) was disturbed by the noise: *"When I was driving, it was always this constant sound. It was worse for the autonomous one. It was constantly adjusting and compensating for the movement."* On the other hand, (R5) found the motor noise useful: *"I actually like it because it gives me the feedback whether I'm driving faster or slower. It also gives me a little bit feeling of life."* Thus, although loud motor noise should be avoided, some noise might be useful.

Wireless Connection: Second most cited problem for video conferencing was the video quality and time lags. (R8) clearly expressed why it was hard to walk with the experimenter using the manual method: *"The frame rate drops all of a sudden and you have no choice but to stop."* Another subject (R9) made use of the displayed sensor data when the video conferencing quality went bad: *"Because of the lag, I just switched to the Kinect (depth image) and the overhead view (laser)."* This was possible because the wide angle camera image was coming from Skype whereas sensor displays were received from the Windows Remote Assistance. Clearly, a big challenge for telepresence systems is to deal with wireless connection problems.

Natural Interaction: Even though the participants thought the motions of the robot were natural to have a conversation (5.4, SD=1.0), some didn't feel it was a natural way to communicate. As seen in Figure 1, the screen displaying the remote user's face is flat and it introduced problems when the robot was traveling on the side of the person. (R5), when asked about walking side by side: *"..we don't have face-to-face. It is not really a conversation."* This raises design considerations on how the remote user's face is brought out. One of the subjects (R5) discovered that the microphone characteristics are different than human hearing: *"I don't have a distance sense if the experimenter is further away or close. If you have the fading audio, then I'll immediately notice."* Whether a telepresence robot should exhibit the same characteristics of human perception or not is an open question and needs further investigation.

Assisted Teleoperation: Telepresence robots should possess a layer to assists the remote user to avoid obstacles and collisions. *Safety* ratings for the manual method were very low (2.3, SD=1.4) and (R8) expressed the concern: *"I was especially worried about running into the experimenter."* This suggests that scenarios involving interaction would demand more attention of the remote users. The teleoperation should also be intuitive and be similar to driving modalities that people are already used to. (R4) stated: *"I was thinking about Manual mode compared to driving a car."* before suggesting *".. maybe something like a cruise control might be good."*

Gaming Experience: Since the robot was controlled by a gaming console controller, some participants likened the manual mode to gaming. (R9) said: *"Manual is like playing video games."* and (R5) said: *"I don't play video games so controlling those consoles is not natural to me."* Thus, it is possible that gamers are less likely to have trouble driving the robot. This observation is also made in [10].

Long Term Interaction: None of the subjects participated in our study had used a telepresence robot before. (R6) justified the inability to use the manual method: *"Maybe if I have some more practice for about several hours of driving the robot, I can use manual as well as autonomous."* (R8) on having fun using teleoperation: *"It was fun because it was the first time I did it but I can imagine that over time, I'll get bored of it."* The *Fun* question in the survey received similar scores for autonomous and manual, possibly because using a telepresence robot was a new experience for the subjects. Studies regarding long term interaction for telepresence robots can yield interesting results, as in [32].

Error recovery: When the person was lost during following, the UI displayed a text that the person was lost so that the remote user can re-initiate the following by clicking on the person. None of the subjects complained about the robot losing the person. When asked explicitly about the robot losing the experimenter, (R10) answered: *"That's not a big deal in comparison to me driving the robot."* Therefore, applications developed for telepresence robots can take advantage of the human being in the loop and does not have to be error-free for deployment.

VII. Conclusion

We presented a method for person following and its evaluation on a telepresence robot. The main contributions of our paper are a novel way of specifying a goal for path planning purposes, and the insights about telepresence robots coming from the usability experiments. More specifically, our approach does not calculate explicit goal positions with respect to the followed person, but makes use of task dependent goal and cost functions to maximize the utility of the robot. Such an approach accounts for the mobility of the target and provides flexibility for designing behaviors. By designing a goal function specific to a flat screen telepresence robot, we have shown that the person following behavior is perceived as socially acceptable by remote users.

User studies showed that autonomous person following is a desired capability for a telepresence robot and it was favored over direct teleoperation for an accompanying task. Autonomous following was found to be safer, easier to use and helped the remote users to pay more attention to the conversation instead of controlling the robot. From the experience we earned from user studies, there are still interesting challenges to explore in terms of human-robot interaction for telepresence robots.

References

[1] B. Keyes, M. Micire, J. Drury, and H. Yanco, "Improving human-robot interaction through interface evolution," *Human-Robot Interaction*, pp. 183–202, 2010.

[2] B. Ricks, C. Nielsen, and M. Goodrich, "Ecological displays for robot interaction: A new perspective," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2004.

[3] G. Venolia, J. Tang, R. Cervantes, S. Bly, G. Robertson, B. Lee, and K. Inkpen, "Embodied social proxy: mediating interpersonal connection in hub-and-satellite teams," in *Proc. of the 28th Int. Conf. on Human factors in computing systems*, 2010.

[4] C. Zhang, Z. Yin, and D. Florencio, "Improving depth perception with motion parallax and its application in teleconferencing," in *IEEE Int. Workshop on Multimedia Signal Proc. (MMSP)*, 2009.

[5] Y. Rui, D. Florencio, W. Lam, and J. Su, "Sound source localization for circular arrays of directional microphones," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2005.

[6] F. Ribeiro, C. Zhang, D. Florencio, and D. Ba, "Using reverberation to improve range and elevation discrimination for small array sound source localization," *IEEE Trans. on Audio, Speech, and Language Processing,*, vol. 18, no. 7, pp. 1781 –1792, sept. 2010.

[7] N. Jouppi and S. Thomas, "Telepresence systems with automatic preservation of user head height, local rotation, and remote translation," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2005.

[8] A. Conceicao, J. Li, D. Florencio, and F. Kon, "Is IEEE 802.11 ready for VoIP?" in *Int. Workshop on Multimedia Signal Proc.*, 2006.

[9] D. Florencio and L.-W. He, "Enhanced adaptive playout scheduling and loss concealment techniques for voice over ip networks," in *IEEE Int. Symposium on Circuits and Systems (ISCAS)*, 2011.

[10] L. Takayama, E. Marder-Eppstein, H. Harris, and J. Beer, "Assisted driving of a mobile remote presence system: System design and controlled user evaluation," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011.

[11] D. Macharet and D. Florencio, "A collaborative control system for telepresence robots," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2012.

[12] M. Desai, K. Tsui, H. Yanco, and C. Uhlik, "Essential features of telepresence robots," in *IEEE Conf. on Technologies for Practical Robot Applications (TePRA)*, 2011.

[13] R. Gockley, "Natural person-following behavior for social robots," in *ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI)*, 2007.

[14] E. Topp and H. Christensen, "Tracking for following and passing persons," in *Int. Conf. on Intelligent Robotics and Systems*, 2005.

[15] E. Pacchierotti, H. Christensen, and P. Jensfelt, "Embodied social interaction for service robots in hallway environments," in *Field and Service Robotics*. Springer, 2006, pp. 293–304.

[16] K. O. Arras, O. Mozos, and W. Burgard, "Using boosted features for the detection of people in 2d range data," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2007.

[17] M. Svenstrup, T. Bak, O. Maler, H. Andersen, and O. Jensen, "Pilot study of person robot interaction in a public transit space," *Research and Education in Robotics, EUROBOT*, 2008.

[18] E. Sisbot, L. Marin-Urias, R. Alami, and T. Simeon, "A human aware mobile robot motion planner," *IEEE Trans. on Robotics*, vol. 23, no. 5, pp. 874–883, 2007.

[19] H. Kwon, Y. Yoon, J. B. Park, and A. C. Kak, "Person tracking with a mobile robot using two uncalibrated independently moving cameras," in *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2005.

[20] M. Kobilarov and G. Sukhatme, "People tracking and following with mobile robot using an omnidirectional camera and a laser," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2006.

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

[22] M. Loper, N. Koenig, S. Chernova, C. Jones, and O. Jenkins, "Mobile human-robot teaming with environmental tolerance," in *Proc. of the 4th ACM/IEEE Int. Conf. on Human robot interaction*, 2009.

[23] A. Ohya and T. Munekata, "Intelligent escort robot moving together with human-interaction in accompanying behavior," in *Proc. 2002 FIRA Robot World Congress*, 2002.

[24] J. Miura, J. Satake, M. Chiba, Y. Ishikawa, K. Kitajima, and H. Masuzawa, "Development of a person following robot and its experimental evaluation," in *Proc. 11th Int. Conf. on Intelligent Autonomous Systems*, 2010.

[25] E. Prassler, D. Bank, and B. Kluge, "Motion coordination between a human and a mobile robot," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2002.

[26] F. Hoeller, D. Schulz, M. Moors, and F. Schneider, "Accompanying persons with a mobile robot using motion prediction and probabilistic roadmaps," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2007.

[27] R. Gockley, "Developing spatial skills for social robots," *AAAI Spring Symposium on Multidisciplinary Collaboration for Socially Assistive Robotics*, 2007.

[28] M. Svenstrup, T. Bak, and H. Andersen, "Trajectory planning for robots in dynamic human environments," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2010.

[29] L. Scandolo and T. Fraichard, "An anthropomorphic navigation scheme for dynamic scenarios," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011.

[30] P. Althaus, H. Ishiguro, T. Kanda, T. Miyashita, and H. Christensen, "Navigation for human-robot interaction tasks," in *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2004.

[31] K. Tsui, M. Desai, and H. Yanco, "Towards measuring the quality of interaction: Communication through telepresence robots," in *Performance Metrics for Intelligent Systems Workshop (PerMIS)*, 2012.

[32] M. Lee and L. Takayama, "Now, i have a body: uses and social norms for mobile remote presence in the workplace," in *Proc. of the 2011 annual Conf. on Human factors in computing systems*, 2011.

[33] D. Nestel, P. Sains, C. Wetzel, C. Nolan, A. Tay, R. Kneebone, and A. Darzi, "Communication skills for mobile remote presence technology in clinical interactions," *Journal of Telemedicine and Telecare*, vol. 13, no. 2, p. 100, 2007.

[34] H. Yanco, M. Baker, R. Casey, B. Keyes, P. Thoren, J. Drury, D. Few, C. Nielsen, and D. Bruemmer, "Analysis of human-robot interaction for urban search and rescue," in *Proc. of the IEEE Int. Workshop on Safety, Security and Rescue Robotics*, 2006.

[35] B. Keyes, R. Casey, H. Yanco, B. Maxwell, and Y. Georgiev, "Camera placement and multi-camera fusion for remote robot operation," in *IEEE Int. Workshop on Safety, Security and Rescue Robotics*, 2006.

[36] J. Xavier, M. Pacheco, D. Castro, A. Ruano, and U. Nunes, "Fast line, arc/circle and leg detection from laser scan data in a player driver," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2005.

[37] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.