Learning Spatial Constraints using Gaussian Process for Shared Control of Semi-autonomous Mobile Robots

Kun Qian[1,2], Dan Niu[1,2], Fang Fang[1,2] and Xudong Ma[1,2]
[1]*Key Laboratory of Measurement and Control of CSE, Ministry of Education, No.2, Sipailou, Nanjing 210096, China*
[2]*School of Automation, Southeast University, No.2, Sipailou, Nanjing 210096, China*

Keywords: Mobile Robot, Gaussian Process, Shared control, Semi-autonomous, Wi-Fi Signal Strength.

Abstract: In this paper, a novel human-robot shared control approach is proposed to solve the problem of semi-autonomous mobile robot navigation with the spatial constraints of maintaining reliable Wi-Fi connection. In particular, the presented approach benefits from using Gaussian Process Regression method to learn the distribution of indoor Wi-Fi signal strength (WSS) and to fuse it with the environmental occupancy probability. The resulting WSS-Occupancy hybrid map is further utilized for generating paths that prevent the robot from violating the spatial restriction. A shared control strategy is designed to implement the WSS-aware navigation behaviour. The approach is evaluated by both simulation and real-world experiments, in which the results validate the practicability and effectiveness of the approach.

1 INTRODUCTION

With the increasing prevalence of wireless LAN in indoor environments, tele-operated mobile robots have been applied to environment inspection and monitoring applications(Sgouros and Gerogiannakis, 2003)(Pitzer et al., 2012). In situations that spatial constraints other than obstacles are imposed to such a networked mobile robot, full tele-operation may not be reliable enough and semi-autonomous robots(Tang et al., 2009) with human-robot collaboration have become important in network robotic system.

In the context of this study, the focus is particularly on an unknown spatial constraint of maintaining reliable Wi-Fi connections during robot navigation. Our motivation originates from the fact that in indoor environments, when a robot navigates through an area with poor Wi-Fi signal strength (WSS), the tele-operation system may temporarily lose control over the robot, neither can continuous and high quality tele-presence (visual and audio) feedback be ensured.

By treating the Wi-Fi distribution as a static spatial constraint here, the solution is to design a WSS-aware navigation behavior for ensuring continuous and reliable Wi-Fi connection during the robots exploration. We define the problem mentioned above as *spatially restricted navigation* of a tele-operated mobile robot.

Intuitively, learning the distribution of spatial constraints in the environment can be benefit to prevent a robot from violating the restriction. Learning the spatial distribution of an indoor environment from a mobile platform can be formulated as a well-known regression problem, i.e., to predict sensor values at locations where the robot doesn't traverse. Gaussian Process(Rasmussen and Williams, 2006)(Qian et al., 2016) is a powerful formalism for predict the probability distributions over sensor values at uncovered locations. In Jadaliha's work(Jadaliha et al., 2012), the authors employed Gaussian Processes (GPs) to build non-parametric probabilistic models using data from a pilot sensor work deployment, for monitoring spatial phenomena of interest. In order to handle the diffusion and patches effects of complex interaction of gas, Stachniss(Stachniss et al., 2009) proposes to learn two-dimensional spatial models of gas distributions using a sparse Gaussian process mixture model, which accurately represents the smooth background signal and the areas with patches of high concentrations. These recent studies(Krause et al., 2008)(Ferris and D. Hahnel, 2006)(Xu et al., 2011)(Xu and Choi, 2011)(Engel et al., 2003)(Ko et al., 2007) have shown that Gaussian processes are an attractive modeling technique in this context since they do not only provide an estimate of sensory data for each point in the space but also the predictive uncertainty. To our best knowledge, the GPs approach has not yet been applied to model the indoor Wi-Fi signal strength dis-

Qian, K., Niu, D., Fang, F. and Ma, X.
Learning Spatial Constraints using Gaussian Process for Shared Control of Semi-autonomous Mobile Robots.
DOI: 10.5220/0006382902010208
In *Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2017) - Volume 2*, pages 201-208
ISBN: Not Available

tribution.

Another challenge is how to utilize the recovered spatial constraints to implement *spatially restricted navigation* behaviors of a robot. Semi-autonomous navigation will outperform full tele-operation when spatial constraints are considered. Two major problems are concerned in this paper:

- Firstly, if the spatial distribution of WSS is recovered by the GPs, the distribution should be integrating with the occupancy grid map in order to achieve a WSS-aware navigation behaviour.
- Secondly, a semi-autonomous behaviour of robot is desired to correct the path designated by the remote user and prevent itself from entering the areas with low WSS.

To deal with semi-autonomous control of mobile robots, human-robot shared control methods(Li et al., 2015) have been applied when multiple inputs and multiple constraints are considered. The concept of shared control is to select between the two sources of commands and combine them to ensure task accomplishment while satisfying dynamic constraints. However, to our best knowledge, there is little research towards the solution of the WSS-aware navigation problem.

In this paper, we propose a practical problem called WSS-aware navigation of a tele-operated mobile robot, which is an example of a general concept of *spatially restricted navigation*. To tackle this problem, we present an approach that enables semi-autonomous navigation of a tele-operated mobile robot by learning WSS-Occupancy hybrid map, in which Gaussian Process is used as an effective model to learn the WSS distribution with limited number of training samples and predict the WSS distribution in a coordinate frame that is consistence with the robots occupancy grid map. The concept of shared control is introduced to implement a semi-autonomous navigation behaviour of mobile robot that can avoid the risk of violating the constraints, i.e., continuous and reliable Wi-Fi connection during navigation is ensured.

2 WEB-BASED ROBOTIC SYSTEM FOR MONITORING

A web-based robotic system is developed for remote environment monitoring. The mobile robot is equipped with a HOKUYO laser range finder for self-localization and obstacle avoidance and a video camera for capturing local video streams. The onboard computer integrates a network card to measures the Wi-Fi signal strength (WSS). The web-based robotic system is built with three-layer architecture, as shown in Figure 1:

- Web Client Layer (WCL). The WCL provides a user with a client GUI to monitor the remote environments as well as to get access to the direction control of a robot.
- Network Service Layer (NSL). The NSL is comprised of a Web server and an Audio/Video Server. Another key role of NSL is to conduct the shared control strategy, and thus the shared control effect is transparent to the user.
- Local Robot Control Layer (LRCL), The LRCL runs various ROS (Robot Operating System)(Quigley et al., 2009) based nodes to access robot hardware and to perform perception, localization, navigation, mapping and other modules. The proposed WSS distribution mapping function is also deployed in LRCL for creating and storing the robots knowledge about the spatially constrained environment.

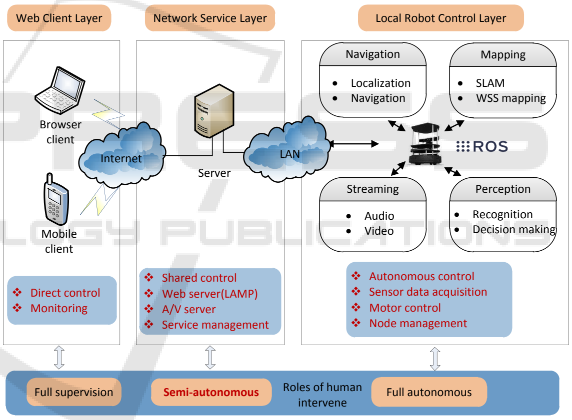


Figure 1: Web-based robotic system architecture.

A browser-based GUI is also provided to receive human input commands from keyboard, joystick and other types of interface, as shown in Figure 2. Given the direction human input commands and the robots autonomous control commands, the concept of shared control is to select between the two sources of commands and combine them to ensure task accomplishment while satisfying dynamic constraints.

In our remote monitoring application, a typical constrain concerned in this paper is to maintain solid Wi-Fi connection for ensuring smooth video streaming. With regard to this, the key to shared control is to model the distribution of indoor Wi-Fi signal strength and fuse such spatial distribution with the environment occupancy probability distribution, which we define as WSS-Occupancy hybrid map building.

Based on this a decision making module is proposed to select or combine the commands from human inputs and the robots autonomous control inputs generated from a navigation controller based on the WSS-Occupancy hybrid map. The shared control framework is shown in Figure 3.

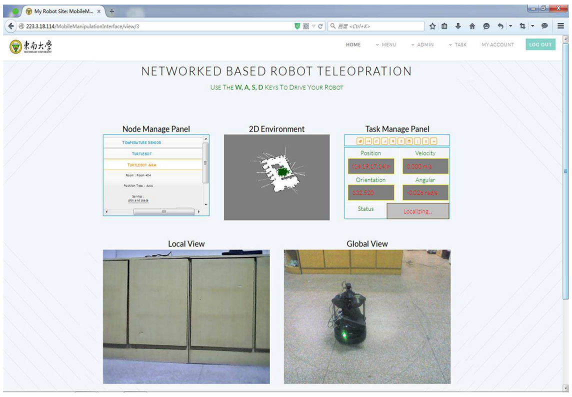


Figure 2: Browser-based GUI.

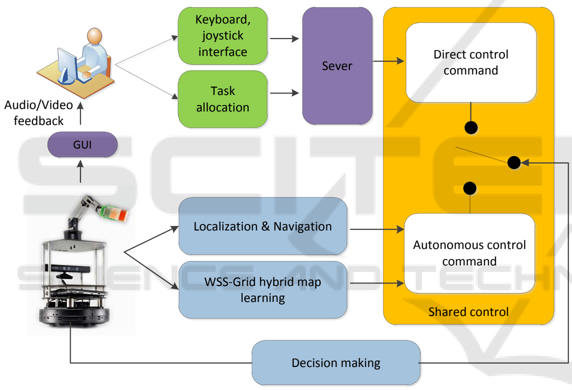


Figure 3: Shared control framework.

3 LEARNING WSS SPATIAL DISTRIBUTION

3.1 WSS Samples Accquisition

A robot explores the environment and constructs the grid map of the environment using its onboard laser sensor readings. In the situation of both robot position and environmental map are unknown, the popular GMapping algorithm(Grisetti et al., 2007) is employed to implement highly efficient SLAM (simultaneously localization and mapping).

As the exploration and mapping proceeds, the robot simultaneously captures the Wi-Fi signal strength (WSS) from the network card of the onboard computer along the path it travels. At every fixed time interval t the robot encapsulates the WSS data as each data packet:

$$D = (x_i, y_i) | i = 1, 2, \ldots, n \tag{1}$$

in which x_i is the positions of sample points, and y_i is the WSS measured at the corresponding position timestamp t_i.

3.2 Learning WSS Distribution with Gaussian Processes

A Gaussian process defines a distribution over a space of functions. Let $D = \{\{x_1, \overline{y_1}\}, \{x_2, \overline{y_2}\}, \ldots, \{x_n, \overline{y_n}\}\}$ be a set of training data generated from a noisy process $\overline{y_i} = f(x_i) + \varepsilon_i$, where $x_i \in \mathcal{R}^d$ is an input sample data and $y_i \in \mathcal{R}$ is a target or an observation. ε_i is additive Gaussian noise with zero mean and unknown variance σ_ε^2 , i.e.,$\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$. A Gaussian Process considers $f(\cdot)$ as a random function and estimates the posterior distribution $p(f|D)$ based on the prior $p(f)$ and the training data D. A GP prior $p(f)$ is completely specified by its mean function and covariance function,i.e., $f \sim GP(m_f, k_f)$,which can be formulated as:

$$p(f(x)) \sim \mathcal{N}(m_f(x), k_f(x, x)) \tag{2}$$

Assuming that we have two input points x_p and x_q, and the corresponding function values at these points are $f(x_p)$ and $f(x_q)$, respectively. We consider a prior mean function $m_f \equiv 0$ and a typical squared exponential (SE) covariance function, or called kernel function:

$$cov(y_p, y_q) = k_{SE}(x_p, x_q) + \delta_{pq}\sigma_\varepsilon^2 \tag{3}$$

in which:

$$k_{SE}(x_p, x_q) := \alpha^2 \exp\left(-\frac{1}{2}(x_p - x_q)^T \Lambda^{-1} (x_p - x_q)\right) \tag{4}$$

with $x_p, x_p \in \mathcal{R}^d$. In the above equation, $\Lambda = diag([l_1^2, \ldots, l_d^2])$ is a diagonal matrix of squared characteristic length-scales l_i^2, $i = 1, \ldots, d$. α^2 is the signal variance and δ_{pq} is the Kronecker symbol that is unity when $p = q$ and zero otherwise.

If we denote X as the entire set of input values, Equation (3) can be rewritten as:

$$cov(y) = K + \sigma_\varepsilon^2 I \tag{5}$$

where K is the $n \times n$ covariance matrix with $K[p, q] = k_{SE}(x_p, x_q)$. The prior over function f depicted by Equation (5) indicates that one can generate the matrix K from a set of input values X and then sample a set of corresponding target values $y \sim \mathcal{N}(0, K + \sigma_\varepsilon^2 I)$.

Given a training dataset D of n input-output pairs that contains samples positions

$X = \{x_i | i = 1, 2, ..., n\}$ and their corresponding WSS values $y = \{y_i | i = 1, 2, ..., n\}$, a GP model is trained. Based on the trained GP model the posterior distribution can be computed. For a new input x_* that represents a position where WSS at this position is never measured, the joint conditional distribution of the estimated target value y_* can be estimated by the prediction distribution $p(y_*|x_*, D)$. The posterior over function values, conditioned on the training data X and y is Gaussian with mean μ_{x_*} and variance $\sigma^2_{x_*}$ (Deisenroth et al., 2014):

$$p(f(x_*)|x_*, X, y) = \mathcal{N}(\mu_{x_*}, \sigma^2_{x_*}) \tag{6}$$

where

$$\mu_{x_*} = {k_*}^T (K + \sigma^2_\varepsilon I)^{-1} y \tag{7}$$

$$\sigma^2_{x_*} = k(x_*, x_*) - k_*^T \left(K + \sigma^2_\varepsilon I\right)^{-1} k_* \tag{8}$$

The chosen kernel function determines the performance of the GP regression. The hyper-parameters of a GP with Gaussian kernel for a particular dataset can be learned from the training data, in which the hyper-parameters are derived by maximizing the log marginal likelihood using empirical Baysian inference(Rasmussen and Williams, 2006).

3.3 Fusion of Grid Map with Wi-Fi Signal Strength Distribution

The learned WSS distribution map models the spatial distribution of WSS within a given indoor context. We use Equation (9) to transfer raw WSS data S into G_W that takes similar value as a grey image pixel:

$$G_W = 255(S + S_{\min})/(S_{\max} - S_{\min}) \tag{9}$$

, in which $S_{\min}$ and $S_{\max}$ are the lower bound and the upper bound of transformation, which are set as:

$$S_{\min} = -113dBm \tag{10}$$

$$S_{\max} = 0dBm \tag{11}$$

, according to our inspection of the range of sample WSS values in typical indoor environments. We define the minimal value of WSS that ensures reliable network connection in the environment is S_{accept} and thus the corresponding value after the transform is G_{accept}. Furthermore, since the Wi-Fi distribution map is learned on actual sensor readings recorded at locations in the world coordinate that is consistent with the occupancy grid map of the environment, the WSS distribution map can be fused with the occupancy grid map, resulting in a WSS-Occupancy hybrid grid map. Denote G_I as the grey value of a pixel in the occupancy grid map and G_W are defined in Equation (9). The resulting grey value G_B of the corresponding pixel in the WSS-Occupancy hybrid grid map is given in Equation (12):

$$G_B = \{ \begin{array}{cc} G_W & G_W < \min(G_{accept}, G_W) \\ G_I & else \end{array} . \tag{12}$$

4 SHARED CONTROL STRATEGY

Using the WSS-Occupancy hybrid grid map described in Section 3, a desired trajectory can be computed by a regular navigation controller that is comprised of a global path planner and a reactive local planner. The path will ensure goal-directed navigation while avoid the robot to pass through the areas with low WSS, so that continuous network connection will be strictly ensured.

Using a local path planner such as VFH or ND algorithm(Minguez and Montano, 2005), the autonomous control set u_r is obtained. Given the human direct control $u_h = [v_h, w_h]^T$ and the output of the reactive local planner $u_r = [v_r, w_r]^T$, in which v and w represent the translational speed and the rotational speed of the robot, respectively, the shared controller adopted in this work can be formulated as a combination of both two sources of command as shown in Equation (12). ρ is the allocation weight with domain $0 \leq \rho \leq 1$.

$$u_s = (1 - \rho)u_h + \rho u_r \tag{13}$$

Since a robot control command will not exactly cause ideal movement but introducing certain uncertainties, we roughly predict the distribution of the robot's possible future positions in a small time interval Δt ahead by adding Gaussian noises to the translational and rotational velocity. Firstly we assume that the robot's heading is constant but the translational velocity is v, thus the robot's possible movements are placed along a vector $\overrightarrow{OA}$ after Δt. Then we assume that the robot's rotational velocity changes randomly within $[-\Delta w, \Delta w]$ in Δt. The result is a fan-shaped area as shown in Figure 4 that contains all possible u_r.

Let the feasible autonomous control input set as U_r and feasible human input set as U_h. Four types of relationships between U_r and U_h as shown in Figure 5 are considered. If U_h is a subset of U_r, i.e.,$U_h \subseteq U_r$, the tele-operation command is regarded safe for ensuring both goal-reaching and obstacle avoidance. In this case, any ρ that satisfies $0 \leq \rho \leq 0.5$ can be taken to generate a synthetic control command. In three other situations such as $U_r \subseteq U_h$, $U_h \cap U_r \neq \{\emptyset, U_h, U_r\}$ and

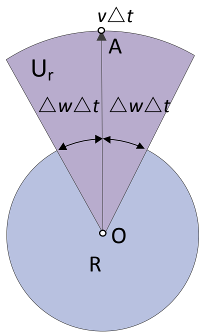


Figure 4: Definition of U_r.

$U_h \cap U_r = \{\emptyset\}$, the tele-operation commands exceed the safe range of the autonomous decision and it is therefore difficult for the user to manually direct the robot to achieve the goal while meeting the constrains. In this case, we simply assign the allocation weight $\rho = 1$ to ensure successful task accomplishment by robot's autonomy only.

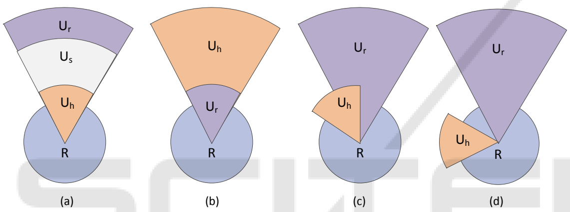


Figure 5: Four types of relations between U_h and U_r.

5 EXPERIMENTS

5.1 WSS-Occupancy Hybrid Mapping

We firstly verify the performance of WSS distribution mapping in a simulation experiment. A Wi-Fi signal generator is utilized to simulate the Wi-Fi signal that travels in a simulated indoor environment as shown in Figure 6(a). In the simulation, we use Indoor Propagation model, a.k.a, the MK model(Motley and Keenan, 1988), to generate the ground truth Wi-Fi distribution. This model takes into consideration the signal decline when it travels through various kinds of obstacles. Table 1 gives the signal attenuation rate when the signal propagates through different types of obstacles, such as walls, wood or glasses, et al.

The MK model based simulator uses Equation (14) to compute the predicted path loss for a testing receiver place:

$$S_{pico} = S_0 + 10 \times n_0 \times log(x) + (N_1 S_1 + L + N_n S_n) \tag{14}$$

in which S_0 equals to the pass loss at 1m from the transmitter, n_0 is the signal attenuation rate long the propagation path due to electromagnetic interference, x is the distance between the transmitter place and the testing receiver place, N_i is the number of interior walls between the transmitter and testing receiver place and S_i is the loss factor of different interior wall materials. The result of the simulated WSS distribution is shown in Figure 6(b), which is taken as the ground truth.

Table 1: WI-FI Signal attenuation rate.

Types of obstacles	Attenuation rate
electromagnetic interference	$30 \sim 50$dBm
Concrete walls	$12 \sim 32$dBm
Block walls	$5 \sim 15$dBm
Wooden walls	$3 \sim 8$dBm
Glasses	$1 \sim 5$dBm

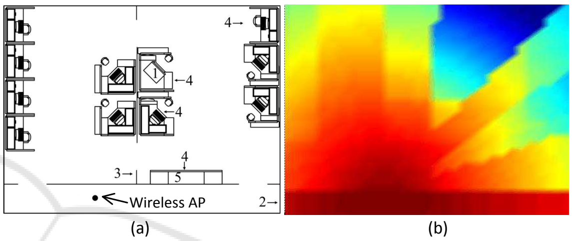


Figure 6: Simulated environment and the ground truth.

Using a Stage(Vaughan, 2008) based robot simulator, a robot travels through the environment and obtains the training dataset. Figure 7 shows the results of occupancy grid map and the robot trajectory for collecting the WSS sample data. Subsequently, a WSS distribution map is trained with Gaussian Process Regression method and the predicted WSS spatial distribution in shown in Figure 8. The result is evident that by utilizing a limited number of training samples, a smooth 2D spatial distribution of WSS can be inferred and the WSS at positions that is not measured can be predicted.

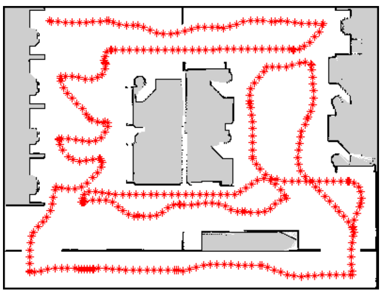


Figure 7: Robot trajectories for collecting the training samples.

In addition, a WSS-Occupancy hybrid map is obtain by fusing the WSS distribution map with the occupancy grid map, resulting in WSS-aware path planning results. A comparison of WSS-aware and non

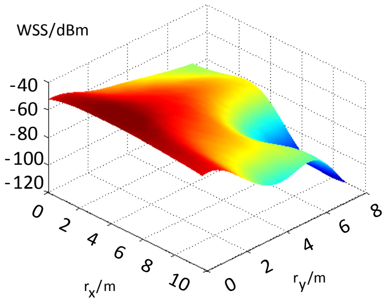
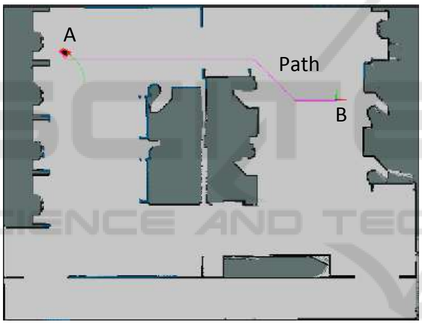
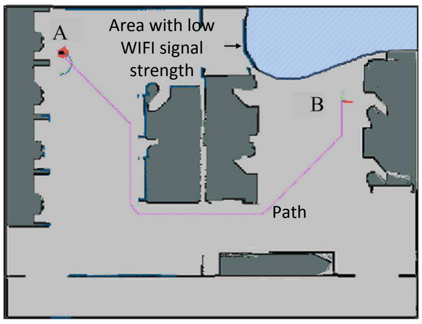


Figure 8: Learned WSS distribution.

WSS-aware path planning results are shown in Figure 9. The result implies that using a WSS-Occupancy hybrid map instead of a traditional occupancy grid map, the robot detours to avoid entering areas with poor WSS. In the experiment the threshold is taken as $S_{accept} = -75dBm$, i.e., $G_{accept} = 68$ to obtain the shaded area in Figure 9(b). A comparison of the WSS captured along two different paths is shown in Figure 10, which indicates that the WSS-aware navigation behavior guarantees continuous and reliable Wi-Fi connection.



(a) Without WSS-constrain



(b) With WSS-constrain

Figure 9: Path planning results.

5.2 Shared Control using WSS-Occupancy Hybrid Map

The shared control performance was further validated in two other real world environment scenarios. The

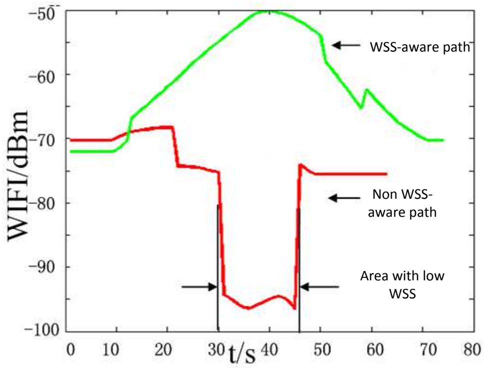


Figure 10: The WSS along two paths.

experiment made use of Turtlebot mobile robot platform in an office environment of size about $100m^2$, where Wi-Fi networks were deployed over multiple Access Points (APs). Scenario 1 was designed as a WSS-aware navigation trial experiment. The actual position of the Access Point (AP) in the testing environment was outside the room as marked in Figure 11. By building the WSS-Occupancy hybrid map of the environment the robot was aware of the area with low WSS, depicted by the shaded area in Figure 11. When the robot was steered by a remote user to travel from point A toward the top-right corner of the map as shown in Figure 11, the shared controller drove the robot to avoid the area with low WSS, as shown by the Part B of the trajectory in Figure 11. The results of the translational and rotational velocity of the robot during the corresponding trial test are given in Fig.12, which reveal the decision making results of the shared controller. The result implies that a robot is capable of adapting its behavior to reduce the risk that confronts with the spatial constrains we have exerted.

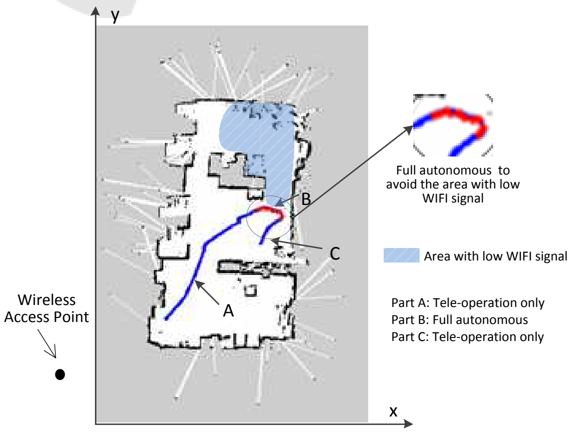
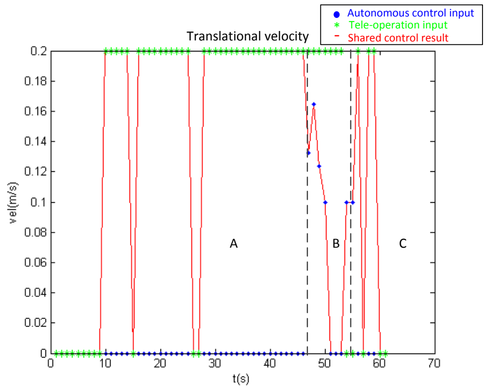
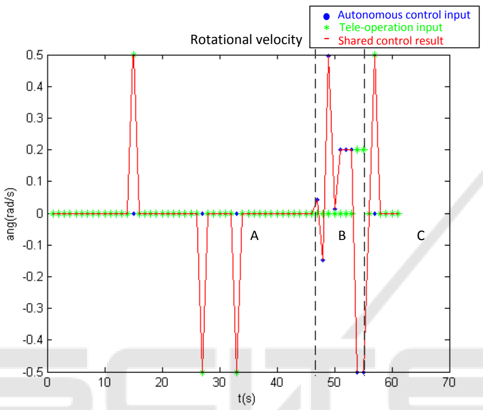


Figure 11: Shared control result in scenario1.

Scenario 2 validates the effectiveness of human supervision for assisting robot's autonomy. The task



(a) Translational Velocity



(b) Rotational Velocity

Figure 12: The robot's velocity in scenario1.

assigned to the robot in this scenario was to travel from P1 to P3 autonomously, but the robot had to travel through a short part of corridor which is too narrow for a robot to pass on its own. When the robot's path planner failed to report a path across the corridor, the shared controller with $\rho = 0$ led to full human supervision, which was achieved by receiving human direction control through the GUI. The robot therefore successfully reached the goal. This experiment indicates that the proposed shared control strategy is also beneficial to the general purposes of semi-autonomous control in a web-based robotic system.

6 CONCLUSIONS

In this paper, a novel human-robot shared control approach is proposed to solve the problem of semi-autonomous navigation with a static spatial constraint of maintaining reliable Wi-Fi connection. In the training mode, a robot explores the environment while building the WSS-Occupancy hybrid map based on Gaussian Process Regress method. In the runtime

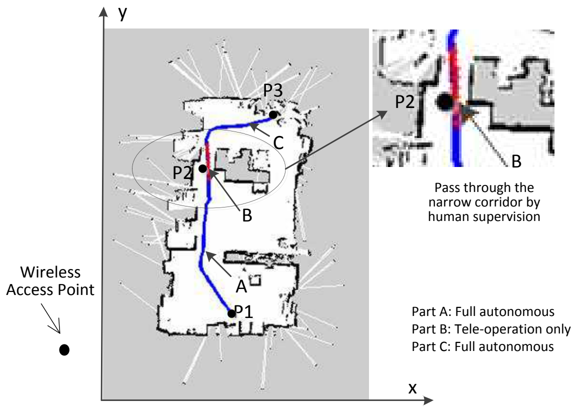


Figure 13: Shared control result in scenario2.

mode, the robot utilizes the WSS-Occupancy hybrid map in a shared control framework to generate paths that prevent itself from entering areas with low Wi-Fi signal strength. A series of experimental studies have been performed, where promising results are obtained. The presented approach can be extended to solve other similar problem of spatially restricted navigation of a tele-operated mobile robot to reduce the risk of violating the constraints while reaching the goal.

ACKNOWLEDGEMENTS

This work is supported by supported by the National Natural Science Foundation of China (Grant No. 61573101, 61573100 and 61504027) and the NSF of Jiangsu Province(NO.BK20140647).

REFERENCES

Deisenroth, P. M., Fox, D., and Rasmussen, C. E. (2014). Gaussian processes for data-efficient learning in robotics and control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37:408–423.

Engel, Y., Mannor, S., and Meir, R. (2003). Bayes meets bellman: The gaussian process approach to temporal difference learning. In *Proceedings of the 20th International Conference on Machine Learning*, pages 154–161.

Ferris, B. and D. Hahnel, D. (2006). Gaussian processes for signal strength-based location estimation. In *Robotics Science and Systems*.

Grisetti, G., Stachniss, C., and Burgard, W. (2007). Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, 23:34–46.

Jadaliha, M., Xu, Y., Choi, J., Johnson, N., and Li, W. M. (2012). Gaussian process regression for sensor networks under localization uncertainty. *IEEE Transactions on Signal Processing*, 61:223–237.

Ko, J., Klein, D. J., Fox, D., and H?hnel, D. (2007). Gaussian processes and reinforcement learning for identification and control of an autonomous blimp. In *IEEE International Conference on Robotics and Automation*, pages 742–747.

Krause, A., Singh, A., and Guestrin, C. (2008). Near-optimal sensor placements in gaussian processes. *Journal of Machine Learning Research*, 9:235–284.

Li, Z., Yang, C., Zhao, S., and et al (2015). Shared control of an intelligent wheelchair with dynamic constraints using brain-machine interface. In *8th International Conference on Intelligent Robotics and Applications, pp.260-271*. Springer.

Minguez, J. and Montano, L. (2005). Sensor-based robot motion generation in unknown, dynamic and troublesome scenarios. *Robotics and Autonomous Systems*, 52:290–311.

Motley, A. and Keenan, J. (1988). Personal communication radio coverage in buildings at 900 mhz and 1700 mhz. *Electronics Letters*, 24:763–764.

Pitzer, B., Osentoski, S., and G. Jay, e. a. (2012). Pr2 remote lab: An environment for remote development and experimentation. In *IEEE International Conference on Robotics and Automation, pp. 3200-3205*. IEEE.

Qian, K., Ma, X., Dai, X., Fang, F., and Zhou, B. (2016). Gaussian process based iaq distribution mapping using an interactive service robot. *Journal of Ambient Intelligence and Smart Environments*, 8:359–373.

Quigley, M., Conley, K., Gerkey, B. P., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A. Y. (2009). Ros: an open-source robot operating system. In *ICRA Workshop on Open Source Software*.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian processes for machine learning*. The MIT Press, Cambridge, Massachusetts, London.

Sgouros, N. M. and Gerogiannakis, S. (2003). Robot teleoperation environments featuring wap-based wireless devices. *Journal of Network and Computer Applications*, 26:259–271.

Stachniss, C., Plagemann, C., and Lilienthal, A. J. (2009). Learning gas distribution models using sparse gaussian process mixtures. *Autonomous Robots*, 26:187–202.

Tang, H., Cao, X., Song, A., and et.al. (2009). Human-robot collaborative teleoperation system for semi-autonomous reconnaissance robot. In *IEEE International Conference on Mechatronics and Automation,pp.1934-1939*.

Vaughan, R. (2008). Massively multiple robot simulations in stage. *Swarm Intelligence*, 2:189–208.

Xu, Y. and Choi, J. (2011). Adaptive sampling for learning gaussian processes using mobile sensor networks. *Sensors*, 11:3051–3066.

Xu, Y., Choi, J., and Oh, S. (2011). Mobile sensor network coordination using gaussian processes with truncated observations. *IEEE Transactions on Robotics*, 27:1118–1131.