

Mobile Robot as Physical Agent of Intelligent Space

Dražen Brščić and Hideki Hashimoto

Institute of Industrial Science, The University of Tokyo, Japan

The Intelligent Space is an area (room, public space, etc.) that has networked distributed sensors, which can be used for observing and gathering information from the space. The main objective for the introduction of Intelligent Space is to provide services to humans inside the space. These services can be either informational, such as the ones provided by displays, or physical. In order to be able to provide physical services mobile robots are introduced in the Intelligent Space as actuators. The network of distributed sensors in the space can therefore be utilized to provide data from the space needed for the control of the robot. Here we present our implementation of an Intelligent Space system that uses spatially distributed laser range finders for tracking the mobile robot and humans inside the space and building the map of the space. Based on these measurements, the control of mobile robot acting as physical agent of the Intelligent Space is developed.

Keywords: intelligent spaces, mobile robot control, ubiquitous sensor system, laser range finders

1. Introduction

1.1. Intelligent Space

In recent years there have been a growing number of researches on spaces that have distributed sensors for monitoring the space. These types of spaces have been given various names, such as smart spaces or ambient intelligence, whereas in our research we have been using the name Intelligent Space (iSpace) [1]. iSpace is a space (room, public space) in which various sensors are distributed in order to gather information from the space. It processes the obtained information in order to achieve a better understanding of the current state and processes in the space. Moreover, the iSpace is provided with different actuators, which enable it to act inside

the space. These three functions – ‘sensing’, ‘understanding’, and ‘acting’ – define the basic framework of the Intelligent Space.

The main purpose for the introduction of iSpace was to develop a system that is able to support humans, i.e. users of the space, in various ways. In that sense, sensors are used to observe the space and gather information about the humans and other objects in the space, whereas actuators are used to provide both physical and informative services to the users of the space. Figure 1 shows the configuration of the iSpace.

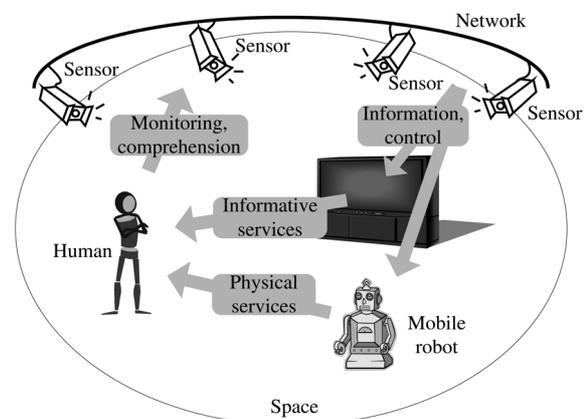


Figure 1. Intelligent Space concept.

There are a number of research approaches similar to Intelligent Space. In the research on Easy Living Technologies [2] human tracking technology is used with the purpose to guess the intent of users in the space in order to automate and facilitate everyday tasks. The Oxygen Project [3] is a similar research aiming at the development of intelligent environments based on human-centered computation. The Interactive Workspace Project [4] is yet another project

in which the possibilities of using ubiquitous embedded sensors and information displays are explored. There are also a number of recent research approaches, for example [5], [6], [7], [8], where mobile robots are considered as physical agents like in our iSpace research. Finally, there is a large number of research work on sensor networks, ubiquitous computing, and mobile robot control, which is also more or less related to the iSpace research.

Based on its structure, the Intelligent Space could be characterized as a combination of ubiquitous sensing and robotics. Each of the distributed devices (i.e. both sensors and actuators) is provided with a processing unit and a network interface, so it can process and exchange information with other components of the iSpace. However, they do not act completely independently, but are parts of the iSpace and under its direct control. For example, physical agents, such as mobile robots, do not act on their own, but are used by iSpace as means of performing actions and changing its state. Therefore, sensors and actuators are considered constituents of the iSpace itself, and the whole iSpace is regarded as one robotic system.

In this paper we focus on the physical acting in Intelligent Space, i.e. the control of a mobile robot. Apart from the robot navigation part, the sensing function of the space is also described as a prerequisite for the implementation of control.

1.2. Mobile Robots in Intelligent Space

As described above, spatially distributed sensors give iSpace the capability to obtain various information about the space. But in order to implement actual services to the users of the space, actuating devices are needed. Informative services, such as providing visual or textual data to the user can be realized using displays or projectors, for example using a projector with a pan-tilt unit [9]. However, in order to be able to provide physical services to humans, robotic technology is indispensable. Here mobile robots, due to their mobility, provide a good choice for versatile physical agents inside iSpace. The main tasks for mobile robots inside Intelligent Space are the ones typical for service robots, and include various services such as carrying, delivering, cleaning, etc.

From the point of view of mobile robot control, the presence of distributed sensors is advantageous compared with the control using only onboard sensors. One advantage that the iSpace concept provides for mobile robot control is for the tasks of localization and mapping. By using external sensors, the global position of the mobile robot can be directly measured. This alleviates the need for using a map of the space during localization and estimating the location by comparing the onboard sensors readings with the map. Also, localization and mapping can be done independently, unlike in the case of localization and mapping with mobile robots where the two tasks are dependent and have to be considered together, leading to the simultaneous localization and mapping (SLAM) problem. SLAM has achieved much attention recently and there are several approaches that give good solutions, but also have several restrictions [10]. On the other hand, in iSpace the SLAM problem is avoided, which makes the implementation much easier.

Another advantage the iSpace gives is that, compared to onboard sensors, distributed sensors enable the measurement of obstacles, even when the robot is not near them, or if they are hidden behind other obstacles. This means that the map of the space and the position of moving obstacles, such as humans, can be updated in real time. Since the map is constantly updated, advanced motion control algorithms that consider the changes in the map in real time can be used.

Apart from using them as physical agents, the iSpace can also benefit from mobile robots by utilizing them as mobile sensors. In other words, in order to obtain more information about the space, the sensors onboard the robot can be utilized in addition to the sensors distributed in the space, e.g. to update the map of the space or provide details about the location and pose of the users in the space. However, the introduction of onboard sensors for tracking and mapping is not straightforward, since it would introduce correlations in the estimates, like in the SLAM problem described above. For that reason, in this paper we concentrate only on sensors distributed in the environment.

In this paper we consider the implementation of the control of mobile robot as physical agent of Intelligent Space. To be more precise, we want the iSpace to be able to move the robot

through the space toward a goal. Although this by itself might not be a very useful action, it represents the basis for different physical services, such as object carrying, human guiding, etc. To implement the mobility of the mobile robot inside iSpace, two main objectives need to be developed: sensing and robot navigation. In the sensing part, in addition to robot tracking, tracking of humans is also necessary. Also, in order to implement robot control, a map of the space is needed. For the implementation of mobile robot navigation inside the space, path planning and local control are developed. These are described in the rest of the paper.

Figure 2 shows the iSpace experimental room in our laboratory where the developed methods were tested. It consists of a system of eight CCD cameras, an ultrasound global positioning system, and two mobile robots by ActivMedia Robotics. One laser range finder device, which is the sensor we use for tracking in this paper, is also shown.

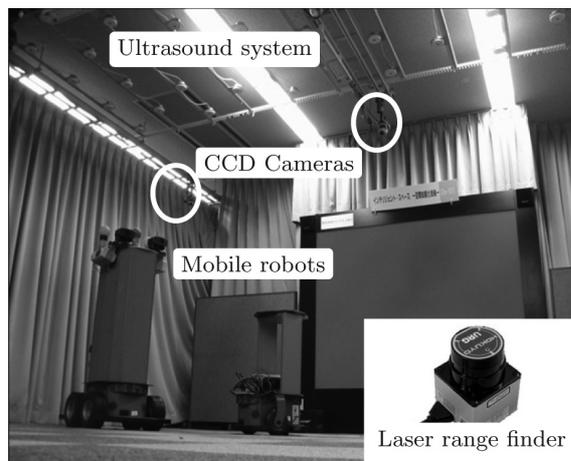


Figure 2. Intelligent Space experimental room.

The rest of the paper is organized as follows. A description of tracking and mapping process using distributed sensors and an overview of commonly used sensors is given in Section 2. Section 3 describes our implementation of the tracking and mapping system using multiple laser range finders, and experimental results of tracking both human and robot. Section 4 describes the implementation of robot navigation and the obtained experimental results. Finally, a conclusion is given in Section 5.

2. Tracking Inside Intelligent Spaces

There are various systems for indoor tracking of humans or other objects. The characteristics of such systems vary greatly with the type of sensors used. A comparison between mainly used sensors is given in Table 1. Some other approaches for object tracking are available, such as magnetic trackers or similar motion capture systems, but they are not appropriate for use in iSpace, so their description will be omitted. Further details on sensors for tracking can be found e.g. in [11].

As can be seen from Table 1, all of these sensors have their advantages and disadvantages. RFID sensors, which are used e.g. in [5], are cheap, but give only a very rough measurement of the position. Floor pressure sensors [12], [13] give rather accurate measurements, but are expensive, and their installation is complex. In the following text we give some more details on cameras, which are probably the most commonly used sensor and ultrasound systems, which we use in our lab as benchmark. Finally, sensors used in this work – laser range finders – are presented.

2.1. Cameras

Computer vision has a very long research history and the tracking of humans and other objects has been studied extensively. A good survey on visual based tracking can be found in [14]. Cameras are inexpensive and easy to install, which is the reason why they have frequently been employed in research work on intelligent environments [2], [3], [4]. Our own camera-based system for tracking both humans and robots in iSpace is described in [15]. However, both in our and other researches, it has proven hard to obtain a robust and easy-to-use tracking system for humans and robots in intelligent environments using cameras only. This is due to the complexity of the observed environments, effects of changing illumination, small resolution if cameras are too far away, the need for accurate calibration, etc. In order to overcome these difficulties, other sensors have to be considered.

Method	Advantage	Limitation
Camera	Cheap Easy to install	Not robust in complex environments
RFID	Cheap	Not accurate Need to wear tags
Ultrasound, iGPS and similar	Accurate	Rather expensive Need to wear tags
Pressure sensors	Non-obtrusive Accurate	Rather expensive
IMU / Encoders	Direct measurement of speed, orientation	Accumulating error Only if equipped onboard
Laser range sensors	Accurate Relatively low price	Occlusions

Table 1. Comparison of Different Sensors for Object Tracking.

2.2. Ultrasound, Indoor GPS, etc.

There are several localization systems based on ultrasound. Examples include Active Bat and Cricket systems [11] and the Zone Positioning System by Furukawa Electric, Ltd. installed in our laboratory (shown in Figure 2). The intelligent environment research project described in [6] also utilizes an ultrasound system in research.

Similar systems also exist, but based on different signal carriers such as radio waves or light. These systems are sometimes given the name "indoor GPS". This is due to their similarity with the GPS system, i.e. using external devices with known location (satellites in the GPS case) to determine the absolute position. One example are the so called "pseudolites" [16], which provide localization by using a signal similar to the GPS signal. This makes possible the use of the same GPS receiver for both outside and inside localization.

The main characteristic of all these systems is their relatively high accuracy. But due to a rather high price, the need for installation of multiple devices in the space and the their accurate calibration prior to deployment, these systems are less attractive for application in real environments. Furthermore, like in radio frequency systems, users have to wear tags.

In the experiments, we use the measurements obtained using the ultrasound positioning sys-

tem in our lab (Figure 2) as a benchmark for the developed laser range finder-based tracking.

2.3. Laser Range Finders

Laser range finders (or LIDARs – Laser Imaging Detection and Ranging) are devices that determine the distance to an object or surface using laser rays. Although these devices have been available on the market for some time, due to the appearance of relatively low priced eye-safe laser range finders, they are frequently employed for mapping and surveying tasks. Also they have become a standard onboard sensor for mobile robots. Since the installation of these devices is very simple and there is no need for wearing special tags, laser range finders appear to be a very good solution for object tracking in Intelligent Space.

There have been several researches on the use of laser range finders for tracking humans and robots. Nevertheless, they have mostly been concentrated on employment of laser scanners on board of mobile robots for tracking humans in the vicinity. Examples of researches on tracking using static laser range scanners are in [17], where laser range finders are used for tracking people in everyday environments and [18], where position of pedestrians in large open spaces was estimated. In our developed system we adopted a approach similar to these two researches, but extend it further to achieve tracking of both humans and mobile robots in iSpace.

In our research we use Hokuyo URG-04LX laser range finders, like the one shown on Figure 2. It has a scan angle of 240° with angular resolution of about 0.36° . The measurement error is about 10 mm for the 1 m range, and about 1% for larger distances. The main possible disadvantage in using this device is its maximum measurement distance of only 5.6 meters. This is a quite small range compared to other laser range finders available on the market, but its price is much lower, which makes it a good and affordable option as sensor for indoor tracking applications and as mobile robot's onboard sensor.

In the tracking system described below, we use several laser range finders as iSpace's distributed sensors. For simplicity, in this paper we focus only on this type of sensor (and encoders for the robot tracking case). A combination of several sensors with different characteristics would probably give better tracking results, however, it would also result in a more complicated tracking system. On the other hand, the described system is flexible enough to allow easy and straightforward incorporation of other sensors, too.

The laser range finders are distributed in the space close to the ground (about 20 cm above the floor). The advantage of the low position is that it is possible to scan also relatively short objects, assuring that all objects on the floor will enter the scan. Also, since humans are scanned at the level of their legs, this reduces the occlusion of the objects behind them. Unfortunately, this way the position of the human has to be estimated based on the position of his/her two legs, which is more difficult than the case when the scan is done e.g. at the waist level, when they appear as just one object in the scan.

3. Tracking and Mapping Using Multiple Laser Range Finders

The flow of the tracking and mapping process is shown in Figure 3. It consists of background subtraction, clustering, data association and tracking, which are performed separately on each distributed sensor (laser range finder). The results are then sent to the position server and mapping server, where the data from all sensors are combined. (Note that the position

server and mapping server can also be implemented on the same computer.) Each of these steps is described here. Some further details about the tracking system can be found in [19].

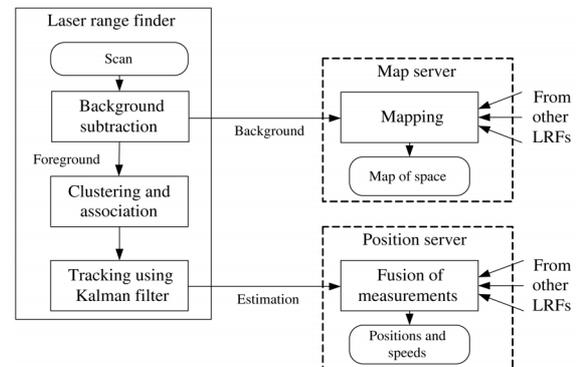


Figure 3. Flow of the tracking and mapping process.

3.1. Extraction of Object Position Form Scan

3.1.1. Background Subtraction

Background subtraction is the processes of determining which parts of the scan are due to static objects, and which come from reflections from moving objects, which, in the case of iSpace, are mainly humans and robots. The part belonging to static objects is called the background, whereas the part belonging to moving objects is called the foreground. The background can be easily determined, for example, by taking several scans while there are no moving objects in the space and taking their average.

The foreground (moving objects) can be extracted from the scan by comparing with the learned background: the parts of the scan that differ from the background more than a given threshold are marked as the foreground.

Sometimes it can happen that some static objects enter in the foreground, for example some moved furniture, etc. After the change, these types of objects are completely static and can be easily identified and inserted in the background. This way the background is always kept updated.

3.1.2. Clustering and Data Association

The scan points in the foreground are clustered based on Euclidean distance between them using a nearest neighbor classifier. This divides the foreground to a number of clusters, each belonging to one of the tracked objects. Clusters with a small number of scan points (1 or 2 points) are discarded, which eliminates possible noise in the measurements.

Before position estimation can be performed, it is necessary to do the data association, that is, to determine which obtained cluster belongs to which tracked object (either human or robot). In order to do that, the cluster centers, which are obtained by averaging the position of all scan points in the cluster, are compared with the positions of currently tracked objects, and each cluster is assigned to the nearest object. The clusters that are far from all currently tracked objects are considered as new objects, and a new tracking process is started for them. This data association based on the Euclidean distance is very simple, but it proved to be sufficient in our experiments, where there were not many tracked objects in the space. For tracking in very crowded environments a more advanced data association method such as JPDA [20] may be needed.

3.1.3. Estimation of the Object Center

From the previous step the positions of cluster centers were obtained. But since the objects are scanned from one side, the center of the obtained cluster of points in general does not coincide with the center of the tracked object. To obtain the positions of the tracked object from the cluster positions we use an approximate relation (see Figure 4):

$$r_{obj} = r_{cl} + d, \quad (1)$$

Therefore, the object position is approximated as being further from the sensor than the cluster center, where the difference d is dependent on the radius of the object, whereas the angle θ of the line between the laser range finder and the object is assumed to be the same as for the cluster center (r_{obj} and θ can be considered as the polar coordinates of the object position with respect to the sensor.) In our experiments d was

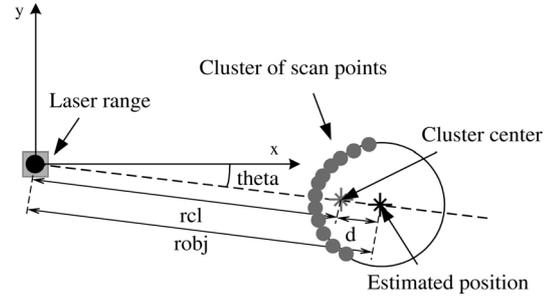


Figure 4. Tracking with laser range finders: the object center is assumed to be at a fixed distance from the center of the cluster of scan points belonging to it.

set to 6 cm for human (i.e. human's leg) and 15 cm for the mobile robot.

In the case of tracking robots, the position estimated using (1) can be considered as a measurement of the robot position. But in the case of tracking a human, the position has to be determined based on two clusters belonging to the human's legs, as noted earlier. (We assume that the humans are not wearing a long skirt or similar clothes, in which case the legs would not be detected.) In a given measurement step these clusters may or may not be available, depending on occlusions in the scan. There are two possible approaches to track humans in this circumstances. One is to track both legs independently and then combine them to obtain the human's position, which is the approach taken e.g. in [18]. The other way is to combine the corresponding clusters and track the human directly, which we adopt here. Since the position of the human can be assumed to be in the middle between both legs, in case both legs are visible, the measurement of the human's position is taken as their mean value. When only one leg is visible, the other leg's position is assumed based on the current human's tracked position, and the measured position is calculated accordingly. However, in this case, the measurement is considered less reliable.

The number of scan points in the cluster and the number of clusters belonging to an object can be used as a simple way to distinguish between humans and other objects. A filtered value of the number of detected clusters is kept for each object. We found out that, for the robot, this value tends to be about 1–1.3, whereas for humans it is usually larger than 1.7, so it proved to be an easy and robust way for determining

the object type. Once the type is known, it is possible to decide which of the described ways to use to calculate the object position.

3.2. Object Model

The mobile robot used in the experiment is a differential drive robot. The pose of the robot is represented by its (x, y) position and orientation θ in the space, which can be described by the following model:

$$\mathbf{x}_{k+1} = \begin{bmatrix} x_k + v_k \Delta T \cos(\theta_k) \\ y_k + v_k \Delta T \sin(\theta_k) \\ \theta_k + \omega_k \Delta T \end{bmatrix} + \mathbf{V} \mathbf{v}_k, \quad (2)$$

where v is the translational velocity and ω is the rotational velocity as measured by the robot's wheel encoders. ΔT is the sample time. The process noise \mathbf{v} represents the uncertainty on the robot velocities, with \mathbf{V} being the appropriate transformation matrix. The noise is assumed zero mean Gaussian with covariance matrix \mathbf{Q} , which is equal to zero when the robot is stopped and grows linearly with increasing speed.

In the human tracking case, the state is represented by the (x, y) position and velocities in the x and y direction v_x and v_y . The model is given by the following linear state space model:

$$\mathbf{x}_{k+1} = \begin{bmatrix} x_k + \Delta T v_{xk} \\ y_k + \Delta T v_{yk} \\ v_{xk} \\ v_{yk} \end{bmatrix} + \mathbf{v}_k \quad (3)$$

The noise \mathbf{v} has the same characteristics as in the robot case.

Using the procedure described in the previous section, laser range finders detect objects in the scan (humans or robot) and measure their position relative to the sensor. The output of the measurement are the range and angle to the detected objects (e.g. in the robot case the range is given by (1), while for humans the measurement is obtained by combining the measurements of two legs as described above.) Therefore the measurement model can be described as:

$$\mathbf{z}_k = \begin{bmatrix} q \\ \tan^{-1} \left(\frac{\Delta x}{\Delta y} \right) + \theta_0 \end{bmatrix} + \mathbf{w}_k, \quad (4)$$

where Δx , Δy and q are given by the following relations:

$$\Delta x = x_0 - x_1, \quad (5)$$

$$\Delta y = y_0 - y_1, \quad (6)$$

$$q^2 = \Delta x^2 + \Delta y^2. \quad (7)$$

The index 0 and 1 stand for the coordinates of the sensor (determined beforehand by calibrating the sensors, as described below in 3.6) and observed object, respectfully. \mathbf{w} represents the measurement noise, which is assumed zero mean Gaussian with covariance matrix \mathbf{R} .

Linearizing the robot process model (2) gives the following process matrix in the robot case:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & -v \sin(\theta) \\ 0 & 1 & v \cos(\theta) \\ 0 & 0 & 1 \end{bmatrix}. \quad (8)$$

When tracking humans, whose process model is given by (3), the process matrix \mathbf{F} becomes:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & \Delta T & 0 \\ 0 & 1 & 0 & \Delta T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (9)$$

Linearizing the measurement model (4) gives the following measurement matrix:

$$\mathbf{H} = \begin{bmatrix} \frac{-\Delta x}{q} & \frac{-\Delta y}{q} & 0 & 0 \\ \frac{\Delta y}{q^2} & \frac{-\Delta x}{q^2} & 0 & 0 \end{bmatrix}. \quad (10)$$

3.3. Estimation Using the Extended Kalman Filter (EKF)

Using the model and its linearization derived in the previous subsection, an Extended Kalman Filter-based estimator can be implemented. The EKF equations are [20]:

$$\hat{\mathbf{x}}_k^- = f(\hat{\mathbf{x}}_{k-1}, v_k, w_k), \quad (11)$$

$$\mathbf{P}_k^- = \mathbf{F}_k \mathbf{P}_{k-1} \mathbf{F}_k^T + \mathbf{Q}_k, \quad (12)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - h(\hat{\mathbf{x}}_k^-)), \quad (13)$$

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1}, \quad (14)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^-. \quad (15)$$

The function f in (11) is given by either (2) or (3), depending on whether it describes the state of the robot or human. Function h in (13) stands for the measurement model given in (4).

3.4. Fusion of Multiple Sensors

By using the method described above, each laser range finder can independently track objects in the area it covers. But, in order to cover a larger area and to achieve a more stable measurement, estimates from multiple sensors are fused together. Every sensor sends the estimated positions and speeds to one computer, called the position server (Figure 3), where the estimates are combined to obtain positions of all tracked objects in the space. This fusion of independent estimates is usually called track-to-track estimation [20].

A simple way to combine measurements is to take the mean of the estimated values as obtained on different sensors for each of the tracked objects. However, since the estimation is done using a Kalman filter, the covariance of the estimate is also available, and it can be used in the fusion process. This results in the following equations [20]:

$$\mathbf{P}_f = \left(\mathbf{P}_1^{-1} + \mathbf{P}_2^{-1} + \dots + \mathbf{P}_n^{-1} \right)^{-1} \quad (16)$$

$$\mathbf{x}_f = \mathbf{P}_f \left(\mathbf{P}_1^{-1} \mathbf{x}_1 + \mathbf{P}_2^{-1} \mathbf{x}_2 + \dots + \mathbf{P}_n^{-1} \mathbf{x}_n \right) \quad (17)$$

where \mathbf{x}_f and \mathbf{P}_f are the fused value and its covariance matrix, while \mathbf{x}_i and \mathbf{P}_i are the estimated value and the associated covariance matrix for the sensor i .

Note that the fusion algorithm does not depend on the type of sensors used for tracking. Therefore the same position server can be easily utilized for fusion of measurements from different sensors.

3.5. Mapping

For each laser range finder, the background of the scan represents the scan points that belong to static objects in the space, as explained earlier. Therefore the backgrounds of all sensors can be used to obtain the map of the space. This is done by sending the background to the mapping server, as shown in Figure 3.

There are several approaches to mapping using range finder devices. Because of the inherent noise in the measurements, most of them rely on

some kind of uncertainty representation. Probability, fuzzy logic and Dempster-Shafer theory-based approaches are the most commonly used, and their comparison can be found in [21]. According to the results presented in that article, fuzzy logic gives somewhat better results than other methods, which is the reason why we chose to use the fuzzy logic-based mapping algorithm. For the map representation we use the occupancy grid [10], in which the space is divided into a grid of rectangular cells of the same size.

Full details of the used mapping algorithm are given in [22]. The fuzzy logic (possibilistic) approach to mapping uses the idea that both empty and occupied areas are defined as fuzzy sets. These sets are not complementary, and for a given cell partial membership to both sets is possible. This allows the identification of areas where there are conflicting observations, in order to build conservative occupancy grids of the space. An example of an obtained map is shown in Figure 5. (Note that this is just one example – we use movable partitions and tables, so the configuration of the experimental space is not always the same).

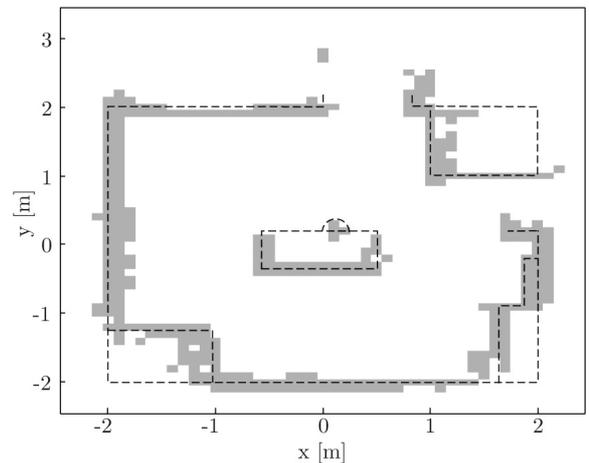


Figure 5. Result of mapping using distributed laser range finders. The grey squares represent the cells estimated as occupied. The dashed line approximately outlines the plan of the room.

Similar to the position server case, the mapping server does not depend on the type of used sensors. This makes the system flexible with respect to the type of sensor used.

3.6. Calibration

In order to convert the estimated positions of tracked objects to the global coordinate system, the sensors need to be calibrated, i.e. their position and orientation in space has to be known. This is important both for the fusion of sensor measurements and building of the map of the space.

Calibration is performed in the following way. A calibration object (an object which can be well detected by a laser range finder) is placed in turn on several points with known global coordinates. For each of these points the laser calculates the position of the object in its local coordinate system. The sensor parameters can then be determined based on the obtained global and local coordinates for these points. By solving the least square error problem, the following equations are obtained:

$$\theta_g = \text{atan} \left(\frac{\sum_{i=1}^n \frac{-(x_{i1}y_{i2} - y_{i1}x_{i2})}{n} + \mu_{x1}\mu_{y2} - \mu_{y1}\mu_{x2}}{\sum_{i=1}^n \frac{(x_{i1}x_{i2} + y_{i1}y_{i2})}{n} - \mu_{x1}\mu_{x2} - \mu_{y1}\mu_{y2}} \right) \quad (18)$$

$$T_{xg} = \mu_{x1} - \mu_{x2} \cos \theta_g + \mu_{y2} \sin \theta_g \quad (19)$$

$$T_{yg} = \mu_{y1} - \mu_{x2} \sin \theta_g - \mu_{y2} \cos \theta_g \quad (20)$$

where T_{xg} , T_{yg} are the sensor coordinates in the global coordinate system, and θ_g is its orientation. The indexes 1 and 2 represent the

global and obtained local coordinates respectively, while μ 's stand for mean values; for example, μ_{x1} stands for the mean value of the measured x coordinates in the global coordinate system:

$$\mu_{x1} = \frac{1}{n} \sum_{i=1}^n x_{i1} \quad (21)$$

The calibration procedure can also be automated, for example by tracking the mobile robot moving in the space and using these data for calibration, as described in [23].

3.7. Experiment – Tracking the Robot and Humans

An example of results obtained using the described tracking method is shown in Figure 6. The obtained position is compared with the estimation using the global positioning ultrasound system in our lab. The position estimation with the ultrasound system is done by tracking a tag on the object (or two in the robot tracking case). In the robot tracking case this measurement is also combined with the output from the robot's wheel encoders to obtain a smooth and stable measurement.

Due to the scanning speed of the used laser range finders of 100 ms, the sampling time of the tracking algorithm was also fixed to that value. This also proved to be large enough to do all the necessary calculations.

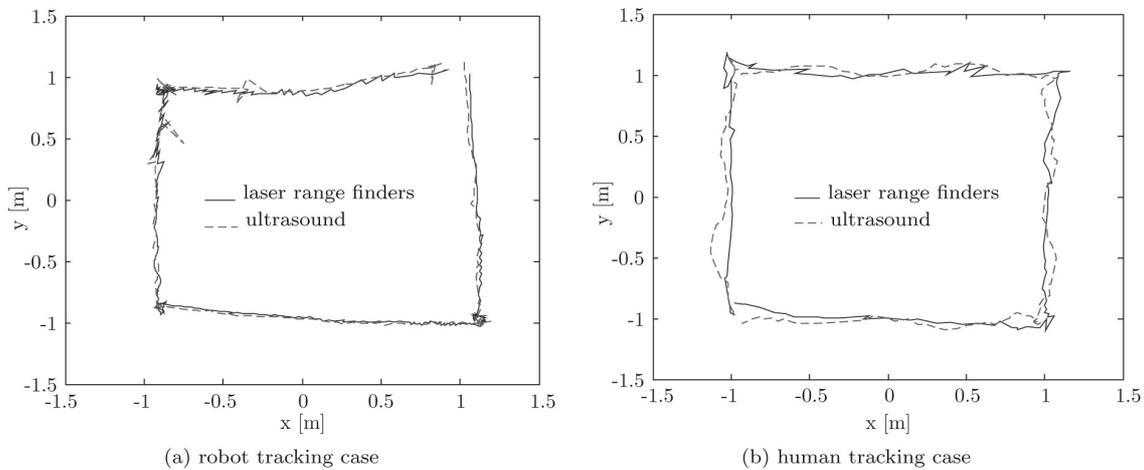


Figure 6. Comparison of the tracking results using laser range finders with the ultrasound measurement. The robot and human tracking case are shown.

It can be seen from the obtained robot tracking results (Figure 6a) that tracking using laser range finders gives very similar results to the one obtained using the ultrasound system (which actually has more outliers). In the human tracking case (Figure 6b) the difference between the two estimates is somewhat larger. Since in this case the ultrasound system measurements are directly used, the larger difference is partly due to the human swinging during walking – since the tag is attached to the human, this swinging is reflected in the measurement, as can be noticed in the figure. However, the difference between the estimates is less than 10 cm most of the time, which confirms the adequacy of the laser range finders-based tracking system for application in iSpace. Also, although not shown in the results, the tracking using distributed laser range finders can, naturally, also be used for tracking multiple objects.

4. Control of the Mobile Robot

Using the method described in the previous section, at any moment estimates of the position of robot and humans and a map of the space are available from the position and mapping servers. By using this information, it is possible to implement the control of the physical agent inside the iSpace. The flow of the control process is shown in Figure 7. It consists of two steps: path planning, and path tracking and obstacle avoidance.

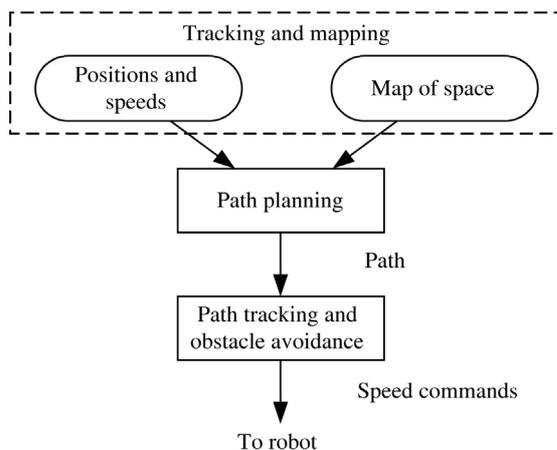


Figure 7. Flow of the mobile robot control process.

In our implementation, all calculations are performed by the iSpace. It sends the calculated

speed commands to the robot, whereas the robot acts just as a mobile platform and does not participate in the control process. The steps of the control algorithm are described below.

4.1. Path Planning

Path planning is the problem of finding the best way for the robot to move through the space based on the map of the obstacles in the space. Since the map obtained during the mapping process is represented as an occupancy grid, the most often used path planning methods are various graph search methods.

One such method that has been very popular for mobile robot path planning applications is described in [24]. This planning method is named Dynamic A* or D*, and it uses advanced computing techniques to speed up the planning time. This way, fast path planning even in large maps became possible. Since the publication of the original D* paper, several improvements have appeared.

However, the main disadvantage of most planning methods including D* is that, due to the grid representation of the map, the headings of the obtained path are restricted to multiples of 45° . But, recently, planning techniques that give smoother paths and can have arbitrary headings have been developed, such as the Field D* method [25]. The main idea of Field D* is to calculate not the cost of grid cells, but the cost of the corner cells, and to linearly interpolate the cost on the cell edges from the costs of the cell corners. Although this linear interpolation approach is not exact, it gives a good approximation of the real cost. By using this method it is possible to calculate the cost of both the points on the edge and inside the cell. Further details of the method and its implementation can be found in [25].

The whole planning process goes as follows. Before planning, the obstacles in the map (both static and moving, i.e. humans) are enlarged by the radius of the robot. Also, in order to keep the robot from moving too close to the obstacles, in the vicinity of obstacles the cost of traversing the cells is higher (the closer to the obstacle the higher the cost). The Field D* method is applied on the obtained grid, which

gives a smooth path from the robot to the goal. This process is repeated in every step.

An example of a path calculated using this method is shown in Figure 8. The figure shows the top view of the space, and an obstacle and one human are present in the space. Two obtained paths are shown – one using the D* method and the other using Field D*. While in the D* case the path is confined to headings of 45°, Field D* gives a very smooth path and moves quickly towards the goal. These good characteristics are the reason why we use Field D* in our implementation.

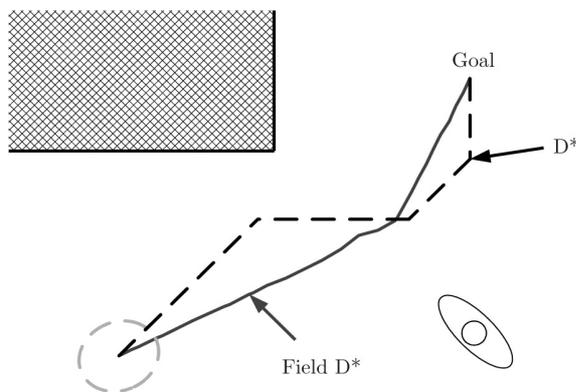


Figure 8. Comparison of paths planned using D* and Field D*. The Field D* gives a smoother path.

4.2. Path Tracking and Obstacle Avoidance

In order for the robot to follow the calculated path and at the same time avoid bumping into obstacles, a control algorithm that takes into consideration the state of the space around the robot is used. Although a variety of approaches exist, in our implementation we use the Dynamic Window Approach (DWA) [26]. The reason for choosing DWA is that it has a physically meaningful interpretation and has been used in several successful implementations in real environments, e.g. [27].

The DWA is used to generate the mobile robot speed commands ensuring that the robot does not collide with obstacles and that it follows the desired path. Moreover, the obtained commands are calculated in such way that they do not violate the dynamic constraints of the robot, i.e. its maximal transversal and rotational accelerations.

In order to achieve that, calculations are done directly in the velocity space of the robot, that is, the space defined by its speed and angular speed. First, the combinations of transversal and angular speed can be reached in the next step based on the allowed maximal accelerations are determined. Then an objective function is used to determine which speed combination is most appropriate to be applied in the next time step. The objective function consists of three sub-objectives – clearance, speed and heading of the robot [26]:

$$G = g_{clear} + g_{speed} + g_{head} \quad (22)$$

Clearance g_{clear} gives a measure of how good the given speed combination is for avoiding the obstacles. It does this by considering the time it would take the robot to hit an obstacle if the given speeds are applied. The speeds for which this collision time is small have lower clearance objective values, while speeds that would certainly lead to collision are regarded as non-admissible and are not considered further.

The speed objective g_{speed} is usually set to have higher values for higher forward speeds so that the robot travels as fast as possible. But, sometimes turning in place might be desired, in which case lower speeds are given higher objective values. We use this in the case when the robot is facing away from the goal, so that the robot will first turn towards the goal before starting to move.

The heading objective g_{head} is used to make the robot move toward the goal, or follow a path. This is done by first predicting the position and orientation that the robot would have after applying the considered speeds for one time step and then breaking until stop. The heading objective value is calculated from the difference between the achieved orientation and the heading towards the goal – the lower the difference the higher the value.

In order for the robot to follow a path, in the heading calculation instead of the real goal one point on the path (path goal point) is used as a temporary goal. This has a convenient property. By changing the difference between the path cost of the robot's current position and the path cost of the path goal point, the maximum speed of the robot can be regulated. The reason for this is that for higher speeds the predicted position of the robot will exceed the path goal point

and thus make the heading objective small. This can be useful, for example, in the case when the robot has to move in the vicinity of obstacles or in areas where there is large uncertainty in the map. The cost of traversing these areas is higher and thus the path cost is changing faster. As a consequence, the path goal point will be closer, and the robot will slow down. In other words, the robot will move slower in those situations, which is exactly the behavior we would wish to have and another reason why we chose this control method. This behavior is depicted in Figure 9. It can be seen that the robot slowed down when close to the obstacles.

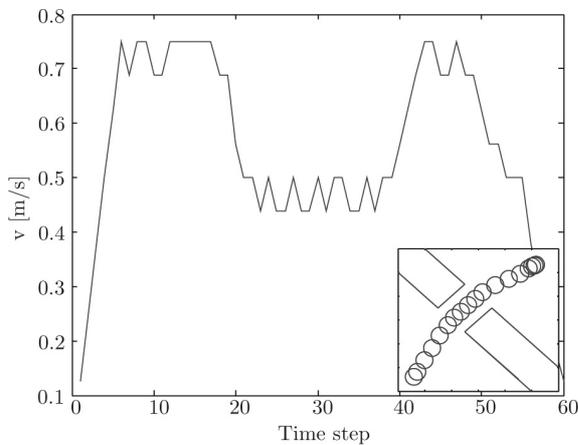


Figure 9. Simulation of the robot slowing down in the vicinity of obstacles: velocity profile. The robot path is shown in the lower right.

In the calculation of the objective function, the calculations of the clearance and heading objective are particularly involved because they

include calculations of the time until collision with an obstacle and prediction of the robot pose after movement and breaking, respectively. Since these calculations have to be done for every speed combination, they can be very expensive. For this reason, similarly to [27], we adopt the use of look-up tables which results in a considerable speed up in the calculation.

4.3. Experiment – Control of Mobile Robot

We tested the developed system in our experimental space. Three laser range finders were arranged in the space and, using them, a map of the space was built and humans and a mobile robot were continuously tracked, as explained in Section 2. Based on the obtained map and positions, the mobile robot was driven in the environment by the iSpace. The mobile robot used was an ActivMedia Pioneer DX2.

The robot was controlled with a sample time of 100 ms. We found out that this is enough to obtain smooth and fast movement of the robot, even in the presence of humans moving in the space. As shown in Figure 7, the robot is controlled with the transversal and angular speed obtained from DWA (the speeds are regulated using Pioneer's onboard PID controllers).

Figure 10 shows an example of the obtained robot navigation results. In this situation the iSpace was given a command to drive a robot from the start position (-2, -2) to the point with coordinates (1, 0). In the first case (Figure 10a) there was no obstacle in the robot path. In that

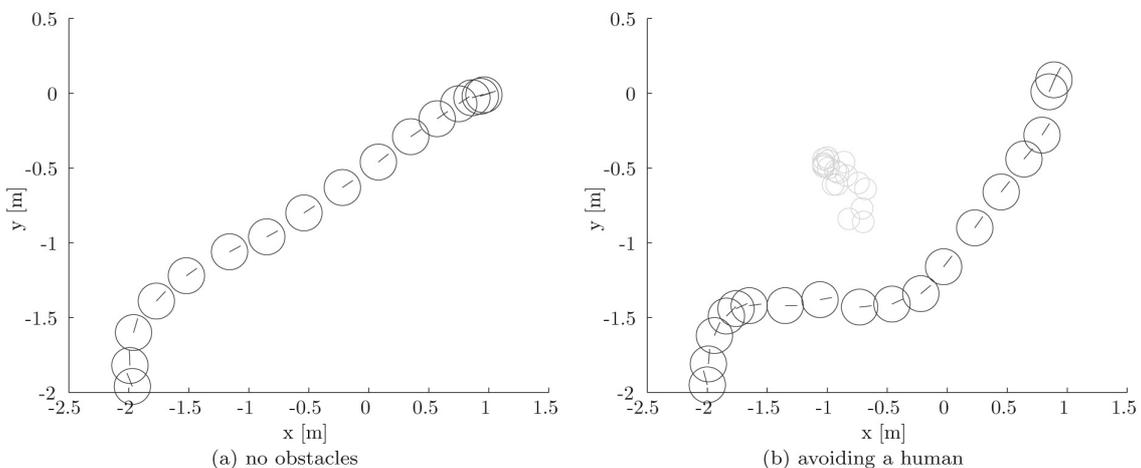


Figure 10. Mobile robot control experiment.

case the robot moved smoothly and on a direct path towards the goal. In the case shown in Figure 10b, there was a human between the robot's current position and the goal. The position of the human was continuously tracked by the iSpace, which enabled appropriate navigation of the robot. It drove around the human and then proceeded towards the goal.

5. Conclusion

In this paper the problem of utilizing a mobile robot as physical agent inside Intelligent Space, a space that can monitor its state using multiple distributed sensors, was considered. As noted, this problem consists of guiding the mobile robot to desired locations in the space, and it can be divided to two main tasks: sensing and robot navigation.

Sensing consists of tracking the position of the robot and humans in the space, and building the map of the space, since both of these tasks are indispensable for the implementation of robot control. We presented an implementation of tracking and mapping using multiple laser range finders distributed in the space. We have shown that laser range finders are an attractive and easy solution for these tasks. Furthermore, a description of the calibration and fusion processes is given. Based on the results of the tracking and mapping processes control of the mobile robot is implemented. The description of the mobile robot control algorithm used was given, and obtained experimental results were presented.

The presented implementation is an example of how to introduce physical agents inside Intelligent Space by utilizing the distributed sensors present in the space. It gives a practical solution that uses easily available sensors and robust control algorithms. Several improvements are also possible, such as the utilization of robot's onboard sensors, which will be considered in our future work.

References

- [1] J.H. LEE, H. HASHIMOTO. Intelligent Space – Concept and Contents. *Advanced Robotics*, 16/3 (2002), 265-280.
- [2] J. KRUMM, S. HARRIS, B. MEYERS, B. BRUMITT, M. HALE, S. SHAFER. Multi-camera Multi-person Tracking for EasyLiving. Presented in the *Proceedings of the 3rd IEEE International Workshop on Visual Surveillance (VS)*, (2000) Washington, D.C., USA.
- [3] L. RUDOLPH. Project Oxygen: Pervasive, Human-centric Computing – An Initial Experience. Presented in the *Proceedings of the 13th International Conference on Advanced Information Systems Engineering (CAiSE)*, (2001) London, UK.
- [4] B. JOHANSON, A. FOX, T. WINOGRAD. The Interactive Workspaces Project: Experiences with Ubiquitous Computing Rooms. *IEEE Pervasive Computing* 1/2 (2002), 67-74.
- [5] B.K. KIM, N. TOMOKUNI, K. OHARA, T. TANIKAWA, K. OHBA, S. HIRAI. Ubiquitous Localization and Mapping for Robots with Ambient Intelligence. Presented in the *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, (2006) Beijing, China.
- [6] T. HORI, A. KANEKO, K. NAGASHIMA, Y. NISHIDA, H. AIZAWA, H. MIZOGUCHI. Delivery Task by a Humanoid Robot in the Sensorized Environment. Presented in the *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC)*, (2004) The Hague, The Netherlands.
- [7] J.H. KIM, Y.D. KIM, K.H. LEE. The Third Generation of Robotics: Ubiquitous Robot. Presented at the *2nd International Conference on Autonomous Robots and Agents*, (2004) Palmerston North, New Zealand.
- [8] M. BROXVALL, M. GRITTI, A. SAFIOTTI, B. SEO, Y. CHO. PEIS Ecology: Integrating robots into smart environments. Presented in the *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, (2006) Orlando, FL, USA.
- [9] Y. TOSHIMA, N. ANDO, H. HASHIMOTO. Information Display System using Active Projector in Intelligent Space – Integration of distributed devices based on RT-Middleware. Presented in the *Proceedings of the 12th International Symposium on Artificial Life and Robotics (AROB)*, (2007) Oita, Japan.
- [10] S. THRUN, W. BURGARD, D. FOX. *Probabilistic Robotics* The MIT Press, Cambridge, 2005
- [11] J. HIGHTOWER, G. BORRIELLA. Location Systems for Ubiquitous Computing. *IEEE Computer*, 34/8 (2001), 57-66.
- [12] T. MORI, Y. SUEMASU, H. NOGUCHI, T. SATO. Multiple people tracking by integrating distributed floor pressure sensors and RFID system. Presented in the *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC)*, (2004) The Hague, The Netherlands.
- [13] R.J. ORR, G.D. ABOWD. The Smart Floor: A Mechanism for Natural User Identification and Tracking. Presented in the *Proceedings of the Conference on Human Factors in Computing Systems (CHI)*, (2000) The Hague, The Netherlands.

- [14] W. HU, T. TAN, L. WANG, S. MAYBANK. A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on System, Man, and Cybernetics C*, 34/3 (2004), 334-352.
- [15] K. MORIOKA, Y. KURODA, S. TAKATSUKA, H. HASHIMOTO. Seamless object tracking in distributed vision sensor network. Presented in the *Proceedings of the SICE Annual Conference*, (2004) Sapporo, Japan.
- [16] C. KEE, H. JUN D. YUN. Indoor Navigation System using Asynchronous Pseudolites. *Journal of Navigation*, 56/3 (2003), 443-456.
- [17] A. FOD, M. MATARIĆ, G. SUKHATME. Laser Tracking and Classification of Multiple Objects. Presented in the *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, (2002) Washington, D.C., USA.
- [18] H. ZHAO, R. SHIBASAKI. A novel system for tracking pedestrians using multiple single-row laser-range scanners. *IEEE Transactions on System, Man, and Cybernetics A*, 35/2 (2005), 283-291.
- [19] D. BRŠČIĆ, H. HASHIMOTO. Tracking of Objects in Intelligent Space Using Laser Range Finders. Presented in the *Proceedings of the IEEE International Conference on Industrial Technology*, (2006) Mumbai, India.
- [20] Y. BAR-SHALOM, T.E. FORTMANN. *Tracking and data association* Academic Press, New York, 1988
- [21] M. RIBO, A. PINZ. A comparison of three uncertainty calculi for building sonar-based occupancy grids. *International Journal of Robotics and Autonomous Systems*, 35/3 (2001), 201-209.
- [22] G. ORIOLO, G. ULIVI, M. VENDITELLI. Real-time map building and navigation for autonomous robots in unknown environments. *IEEE Transactions on System, Man, and Cybernetics A*, 28/3 (1998), 316-333.
- [23] T. SASAKI, D. BRŠČIĆ, H. HASHIMOTO. Implementation of Distributed Sensor Network for Intelligent Space. Presented in the *Proceedings of the IEEE International Conference on Mechatronics (ICM)*, (2007) Kumamoto, Japan.
- [24] A. STENTZ. Optimal and Efficient Path Planning for Partially-known Environments. Presented in the *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, (1994) San Diego, CA, USA.
- [25] D. FERGUSON, A. STENTZ. The Field D* Algorithm for Improved Path Planning and Replanning in Uniform and Non-uniform Cost Environments. Tech. Report, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA, 2005.
- [26] D. FOX, W. BURGARD, S. THRUN. The Dynamic Window Approach to Collision Avoidance. *IEEE Robotics and Automation Magazine*, 4/1 (1997).
- [27] R. PHILIPPSEN, R. SIEGWART. Smooth and Efficient Obstacle Avoidance for a Tour Guide Robot. Presented in the *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, (2003) Taipei, Taiwan.

Received: November, 2007

Revised: April, 2008

Accepted: May, 2008

Contact address:

Dražen Brščić
Institute of Industrial Science

The University of Tokyo

4-6-1 Komaba, Meguro-ku

153-8505 Tokyo, Japan

e-mail: brscic@hlab.iis.u-tokyo.ac.jp

DRAŽEN BRŠČIĆ received his BSc and MSc degrees in electrical engineering from the University of Zagreb, Croatia, in 2000 and 2004, respectively. He is currently working towards his PhD degree at the Institute of Industrial Science, The University of Tokyo. His research interests include mobile robotics, object tracking and intelligent spaces. Mr. Brščić is a member of the IEEE and RSJ.

HIDEKI HASHIMOTO received the BSc, MSc and DrEng degrees in electrical engineering from The University of Tokyo, Japan in 1981, 1984 and 1987, respectively. He is currently an associate professor with the Institute of Industrial Science, The University of Tokyo. From 1989 to 1990 he was a visiting researcher at the Massachusetts Institute of Technology, Cambridge, USA. His research interests are control and robotics, especially advanced motion control and intelligent control. Dr. Hashimoto is a member of the SICE, IEEE, RSJ and IEEJ.
