Development of a Laser-Range-Finder-Based Human Tracking and Control Algorithm for a Marathoner Service Robot

Eui-Jung Jung, Jae Hoon Lee, Byung-Ju Yi, *Member, IEEE*, Jooyoung Park, Shin'ichi Yuta, *Fellow, IEEE*, and Si-Tae Noh

***Abstract*—This paper presents a human detection algorithm and an obstacle avoidance algorithm for a marathoner service robot (MSR) that provides a service to a marathoner while training. To be used as a MSR, the mobile robot should have the abilities to follow a running human and avoid dynamically moving obstacles in an unstructured outdoor environment. To detect a human by a laser range finder (LRF), we defined features of the human body in LRF data and employed a support vector data description method. In order to avoid moving obstacles while tracking a running person, we defined a weighted radius for each obstacle using the relative velocity between the robot and an obstacle. For smoothly bypassing obstacles without collision, a dynamic obstacle avoidance algorithm for the MSR is implemented, which directly employed a real-time position vector between the robot and the shortest path around the obstacle. We verified the feasibility of these proposed algorithms through experimentation in different outdoor environments.**

***Index Terms*—Human detection, machine learning, mobile robot, obstacle avoidance.**

I. Introduction

A LARGE number of people participate in marathon races and more than 500 marathons are organized worldwide annually [1]. Amateur marathoners often have difficulty transporting their personal belongings while they are training for a race. However, if a mobile robot follows a marathoner and carries essential items such as water, food, and clothes, amateur marathoners could enjoy their runs more. Additionally, providing a running path on a map that included running distance, time, and speed would be helpful. This robot is called a "Marathoner Service Robot" (MSR). In order to realize these objectives, this mobile robot should recognize a running marathoner in a real outdoor environment and follow the target person at human running speed (max: 18 km/h or 5 m/s). It is noted that the average speed of a normal amateur marathoner is less than 16 km/h.

A variety of human tracking approaches have been developed with most based on visual tracking [2]–[6] and other approaches such as laser range finder-based human tracking [7]–[11], 3-D sensor tracking [12], [13], RGB-D sensor tracking [14], and camera and laser fusion tracking [15], [16].

Most tracking systems or mobile robots have been designed to operate in indoor environments at human walking speed. A robotic system that not only tracks a human at running speed but also avoids moving obstacles in a cluttered environment has rarely been investigated.

When a human tracking algorithm is implemented in a mobile platform in an outdoor environment, sensors will be exposed to more noise than indoors, and the human target can be occluded or temporarily lost. In such unstructured environments, following a running human is an enormously difficult task.

To improve tracking performance, many researchers have investigated robustly recognizable parts of the human body. Lee *et al.* [10] and Arras *et al.* [11] proposed a human tracking algorithm by scanning two legs, and Glas *et al*. [17] scanned the torso part of a human. However, these works were implemented in an indoor environment. Jung *et al*. [18] analyzed the optimal height of the laser sensor for detecting a human, and found that the torso height was appropriate to detect the marathoner's position in an outdoor environment.

This paper is an extension of Jung *et al*. [18]. Compared to the previous work [18], a human detection algorithm and an avoidance algorithm have been newly investigated to enhance human tracking performance in this paper. Initially, this MSR employs a support vector data description (SVDD) [19] to detect a human by using an LRF. Second, a weighted radius algorithm considering the relative velocity between the robot and an obstacle is applied for obstacle avoidance. Finally, a dynamic obstacle avoidance algorithm for the MSR is implemented, which directly

Manuscript received February 20, 2013; revised September 6, 2013; accepted October 30, 2013. Date of publication December 20, 2013; date of current version June 13, 2014. Recommended by Technical Editor M. Moallem. This work was supported in part by the BK21 Plus Program (future-oriented innovative brain raising type, 22A20130012806) funded by the Ministry of Education (MOE, Korea) and National Research Foundation of Korea (NRF), in part by the Technology Innovation Program (10040097) funded by the Ministry of Trade, Industry and Energy Republic of Korea (MOTIE, Korea), in part by the GRRC Program of Gyeonggi Province (GRRC HANYANG 2013-A02), in part by the Ministry of Trade, Industry and Energy (MOTIE) and Korea Institute for Advancement in Technology (KIAT) through the Workforce Development Program in Strategic Technology, and in part by the MOTIE, Korea, under the Robotics-Specialized Education Consortium for Graduates support program supervised by the NIPA (National IT Industry Promotion Agency) (H1502-13-1001).

E.-J. Jung and B.-J. Yi are with the Department of Electronic, Electrical, Control and Instrumentation Engineering, Hanyang University, Ansan 426-791, Korea (e-mail: jeuij79@naver.com; bj@hanyang.ac.kr).

J. H. Lee is with the Graduate School of Science and Engineering, Ehime University, Matsuyama 790-8577, Japan (e-mail: jhlee@eng.ehime-u.ac.jp).

J. Park is with the Department of Control and Instrumentation Engineering, Korea University, Sejong-Ro 2511, Korea (e-mail: parkj@korea.ac.kr).

S. Yuta is with the Department of Electrical Engineering, Shibaura Institute of Technology, Tokyo 135-8548, Japan (e-mail: yuta@ieee.org).

S. T. Noh is with the Division of Materials and Chemical Engineering, Hanyang University, Ansan, 426-791, Korea (e-mail: stnoh@hanyang.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMECH.2013.2294180

employs a real-time position vector between the robot and the shortest path around the obstacle.

This paper is organized as follows. In Section II, we describe human feature extraction, a human detection method using SVDD, and some fundamental estimation methods. The control and tracking algorithms for the MSR are described in Section III. In Section IV, we discuss the experimental results for a MSR operated in an outdoor environment.

II. Human Tracking

A. Detection of Human Bodies by SVDD

A laser range finder has been used to detect objects and to draw a map of the environment in many robotic systems [7]–[11], [20]. In this paper, we use a laser range finder to detect a human, and according to Jung *et al*. [18], the laser sensor is placed on the robot at torso height and scans an environment at this height. In this section, we discuss pattern recognition of various shapes in clustered data scanned from a human torso.

To extract features of the torso from the scanned data, we apply SVDD [19] which is a support vector learning method for a one-class classification problem. The state-of-the-art detection method of people in 2-D range data uses AdaBoost [11], which utilizes a machine learning approach. The AdaBoost algorithm finds optimal offsets of classifiers and optimal features. However, the AdaBoost algorithm hardly describes the detailed boundary between normal and abnormal data because it uses straight lines for the classifiers. On the other hand, the SVDD algorithm can describe a more detailed boundary because the SVDD uses curved lines for the classifiers. Therefore, with respect to the classification, the SVDD algorithm can perform better than the classifier used in the AdaBoost algorithm [11], [19]. However, the features and some parameters in the SVDD algorithm should be selected manually to get better results.

When the SVDD algorithm is applied in one-class classification problems, training data for only the normal class are provided. After the training phase is finished, the decision has to be made whether each test vector belongs to the normal or the abnormal class. The SVDD method, which approximates the support of objects belonging to the normal class, is derived as follows. Consider a ball B with center $\underline{\mathbf{a}} \in \mathbb{R}^d$ and radius R, and the training dataset D consisting of objects $\underline{\mathbf{x}}_i \in \mathbb{R}^d, i = 1, \ldots, N$. The main idea of SVDD is to find a ball that can achieve two conflicting goals (it should be as small as possible and contain as much training data as possible) simultaneously by solving

$$\begin{aligned} &\min L(R, \underline{\mathbf{a}}, \underline{\boldsymbol{\xi}}) = R^2 + C \sum_{i=1}^{N} \xi_i \\ &\text{s.t.} \quad \|\underline{\mathbf{x}}_i - \underline{\mathbf{a}}\|^2 \leq R^2 + \xi_i, \quad \xi_i \geq 0, \quad i = 1, \ldots, N \end{aligned} \tag{1}$$

where the slack variable ξ_i represents the penalty associated with the deviation of the ith training pattern outside the ball, and C is a tradeoff constant controlling the relative importance of each term. To make a decision with any test vectors, only training data on the boundary are used, therefore the calculation is not very complicated. A detailed description of this algorithm can be found in Tax and Duin [19].

Fig. 1. Examples of scanned parts of a human body. The left figure shows the scanned part from behind the person at a height of 1.3 m. The right figure shows the scanned part from the flank of the person at a height of 1.3 m.

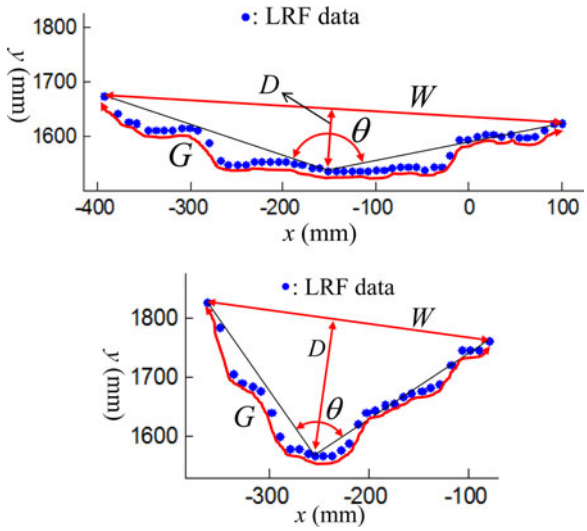


Fig. 2. Definition of four features in clustered data from an LRF.

The shape of the LRF data on a human body cannot be simply defined because the surface of the clothes changes all the time. To analyze the characteristics of the clustered LRF data from a human body, we collected 539 sample datasets to train the SVDD. The sample data were collected from the torsos of five people who were positioned between 1–5 m away from the LRF because we designed the MSR to follow a human at a distance of 1–5 m. Subjects heading in various directions were scanned with respect to the LRF reference frame as shown in Fig. 1, and the height of the scanned region ranged from 0.9 to 1.3 m.

Fig. 1 shows examples of scanning parts of a human when sample data are collected. Fig. 2 shows LRF data obtained by scanning the human body in Fig. 1. We initially defined four features for clustered LRF data to describe the shapes of the LRF data for a human body as shown in Fig. 2. W denotes the *Width* of the clustered LRF data, which is the distance between the first and the last points of the clustered data, G denotes the *Girth* of the clustered data, D denotes the *Depth* of the data, i.e., the distance between the straight line connecting the first and the last points of the clustered data and the farthest point from the line, and θ denotes the *angle* between the straight lines connecting the farthest point to the first and the last points. Additionally, we defined one more feature, *Width*/*Girth*, which

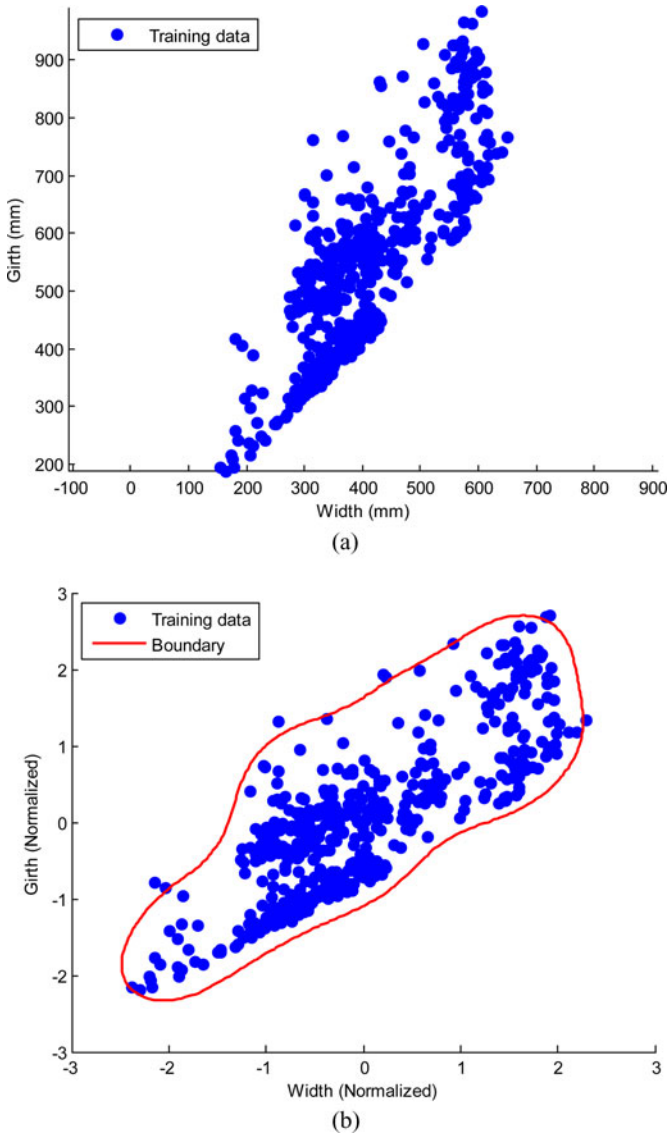


Fig. 3. Training dataset and a result of SVDD in *Width-Girth* domain. (a) Original training data set (sample) in *Width-Girth* domain. (b) A result of SVDD after normalization.

is the ratio of the *Width* to the *Girth*. This feature measures roundness of the shape of the clustered data.

SVDD uses balls defined on a featured space to distinguish a set of normal data from all other possible abnormal objects. The goal of applying SVDD is to obtain the decision boundary for a ball that includes most of the normal sample datasets. We employ a Gaussian Kernel to express the boundary of the sample data as shown in Fig. 3. The Gaussian Kernel makes the surface of the boundary irregular in the input space.

When more than two features are considered in SVDD, it is not easy to show the result graphically. For convenience and simplicity, we show an example of the decision boundary from the SVDD analysis in the *Width-Girth* domain as shown in Fig. 3. Only the *Width* and the *Girth* are considered in the figures. All clustered LRF data inside the boundary of the SVDD are considered human data, and all clustered LRF data outside the boundary are considered nonhuman data.

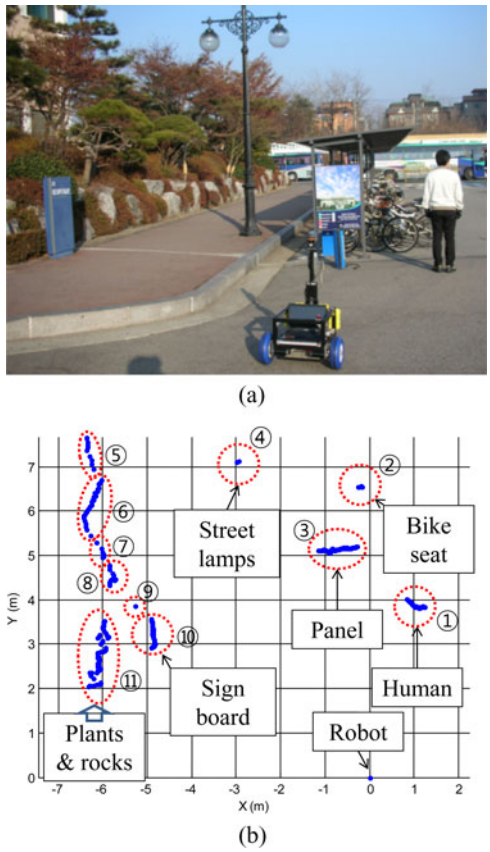


Fig. 4. Experimental environment and LRF dataset. (a) Experimental environment. (b) LRF data (blue dots) from the environment. Each clustered dataset is indicated in a dotted circle and is marked with a circled number.

B. Feature Extraction From an Environment

In this section, we test the classifiers trained by SVDD in an outdoor environment. To see the features of the clustered data, we obtained LRF data from the experimental environment as shown in Fig. 4. The data were obtained by the LRF on the robot, and the robot located at (0, 0) in Fig. 4(b). Then, we extracted the five features from the clustered datasets. A clustered dataset is built by including two neighboring points within 200 mm successively in the scanned data array. Eleven clustered datasets were built from the experimental environment of Fig. 4(a), and the results of the clustering are shown in Fig. 4(b). The circled numbers in Fig. 4(b) denote the clustered data number. Five features of the clustered LRF data are extracted according to the feature definitions above. Then, the results of the feature extraction are summarized in Table I in which the first column represents the five features. The features of clustered datasets number 4 and 9 are ignored (set to zero in Table I) because the number of clustered data points was not sufficient to extract the features defined above.

To determine whether the clustered data are from a human or not, we use the boundary obtained from the training data above. We first employ sets of two features among the five to visualize the results simply. In Fig. 5, two sets of the *Width-Angle* and *Width-Width/Girth* after normalization are used to determine the decision boundaries. In the graphs, the blue dots denote training

TABLE I
EXTRACTION OF FIVE FEATURES FROM THE CLUSTERED DATASETS

Features \ Cluster	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪
Width(mm)[A]	447	86	888	0	712	1261	366	447	0	656	1505
Girth(mm)[B]	517	95	1005	0	727	1440	389	577	0	720	2620
Angle(rad)[C]	2.45	2.31	2.72	0.00	2.83	2.28	2.47	2.01	0.00	2.25	1.89
Depth(mm)[D]	80.2	17.1	70.0	0.0	52.3	241.3	55.0	127.8	0.0	70.4	232.4
Width/Girth[E]	0.86	0.90	0.88	0.00	0.98	0.88	0.94	0.77	0.00	0.91	0.57

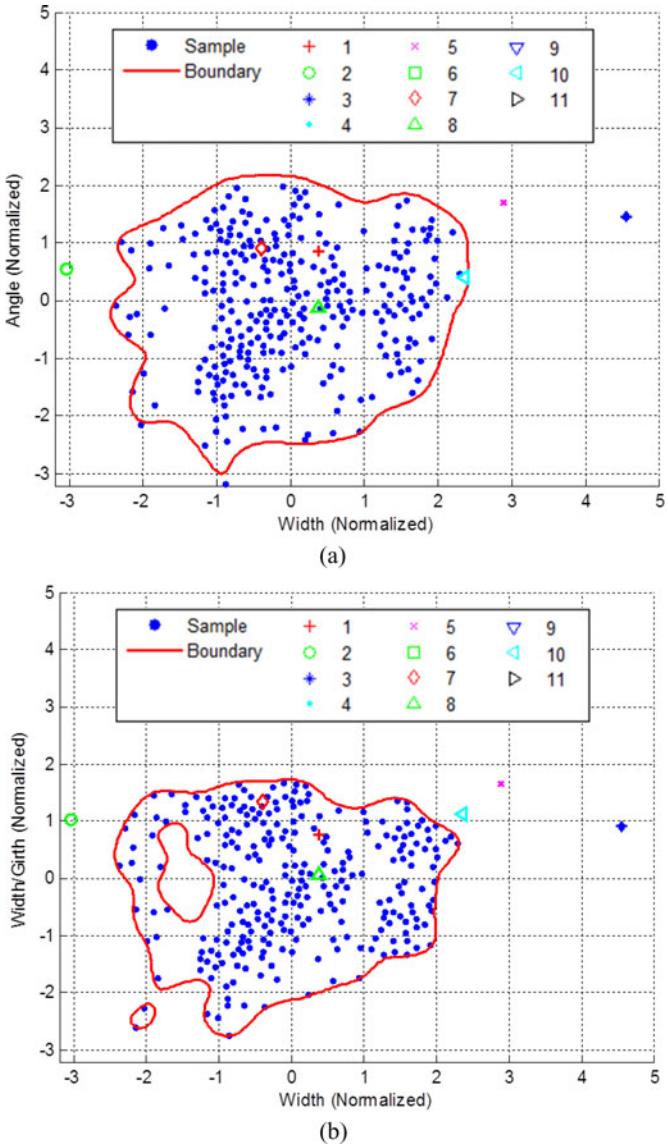


Fig. 5. Classification of the objects in Fig. 4. (a) Classification in *Width* and *Angle* domain. (b) Classification in *Width* and *Width/Girth* domain.

sample data from people. Boundaries that contain almost all of the training samples are obtained in the training phase of the SVDD algorithm.

As a result, four sets of clustered data, numbers 1, 7, 8, and 10, are inside the boundary in Fig. 5(a), and three sets of clustered data, numbers 1, 7, and 8, are inside the boundary in Fig. 5(b). Therefore, clustered dataset number 10, which is the signboard in Fig. 4, can be distinguished from a human by using *Width* and *Width/Girth* features in SVDD. However, the clustered datasets 7 and 8, which correspond to plants and rocks in Fig. 4, respectively, cannot be distinguished by using these two kinds of feature spaces because the features of the clustered datasets 1, 7, and 8 are almost the same as those of the training sample data from people. Because some trees and rocks are similar in shape to humans, it is not easy to distinguish them from humans in 2-D range data.

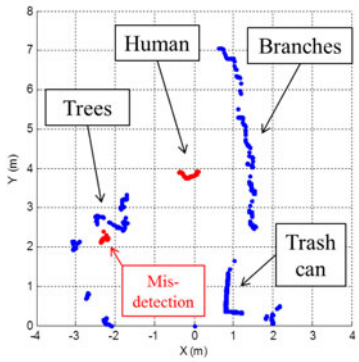
The above analyses indicate that the recognition success rate for a human body differs according to certain features. In the next section, we describe how we selected features for SVDD to detect a human.

C. Optimal Feature Selection

In this section, we describe how to choose a set of features, which would allow the optimal recognition of the human body. To evaluate our proposed approach to detect a human using LRF data, we performed an experiment to diagnose the correctness of human detection. In the experimental environment shown in Fig. 6(a), where there are many trees and branches that have a similar shape to a human, 240 frames of the LRF scan data were obtained while following the person in the experimental environment. As shown in Fig. 6(b), even though there was only one person in each frame, sometimes trees or other objects, which have the same shapes as a human in the LRF data, can be considered to be human-like (see Fig. 5). From the 240 frames of scanned data, 5575 clustered datasets were collected, and we counted how many times a human was correctly detected by applying SVDD with various combinations of the five features. We also counted the number of incorrect detections of a human. Correct human detection (CHD) refers to when the SVDD considered the clustered data from a person to be a human, and incorrect human detection (IHD) refers to when the SVDD considered the clustered dataset of an object to be a human.

Table II shows some results of human detection that use various combinations of features in the outdoor environment shown in Fig. 6. Letters A to E in the first row of Table II represent features listed in the first column of Table I, and the number of letters represents the number of the features used for SVDD. For example, BE means that the results shown in column BE were obtained using two features, *Girth* and *Width/Girth*. The

(a)



(b)

Fig. 6. Experimental environment and LRF data from the environment. (a) Experimental environment. (b) LRF data from the environment.

TABLE II
RESULTS OF SVDD WITH VARIOUS COMBINATIONS OF FEATURES

	A	B	BE	CD	ACE	ABC	ABE	ABCD	ACDE	ABCDE
CHD	223	129	205	214	236	215	222	232	221	202
%	93	54	85	89	98	90	93	97	92	84
IHD	662	795	427	1436	414	867	383	431	1311	490
%	12	14	8	26	7	16	7	8	24	9

ten combinations of the five features that outperformed the other combinations are listed in the first row of Table II.

CHD and IHD denote the number of correct detections of humans in the data (human datasets in the 240 frames) and the number of IHD of humans among 5575 clustered datasets, respectively. When many features were used in SVDD, the results were relatively good. However, inclusion of many features in SVDD increases computational complexity. Therefore, we chose the combination ACE (*Width*, *Angle*, and *Width/Girth*) to detect a human. This combination comprised a relatively small number of features and showed good performance. It successfully detected humans 98% of the time with IHD frequency of only about 7% as shown in the sixth column in Table II. For human tracking, the CHD rate is much more important than the IHD rate. The cases of IHD can be handled with a data association method because the location of the human candidate detected by IHD is different from that of the target human.

To show how well SVDD with the selected feature set (ACE) works, we compare it with the case of using only the feature *Width*. Fig. 7 shows the number of human candidates detected in each set of scan data with different feature selections in the same environment as Fig. 6. Fig. 7(top) shows the results of human detection when only the *Width* feature is considered. In this experiment, when the width of a clustered dataset was between 200 and 700 mm, the dataset was considered to contain a human candidate. There was only one person, but there were many trees and branches that are similar in width to a human. Therefore, the average number of detected human candidates per scan was 4.66. Fig. 7(bottom) shows the result of human detection when SVDD was applied using three features: *Width*, *Angle*, and *Width/Girth*. When SVDD was applied in the same environment, the average number of detected human candidates per scan was 3.17. Even though there were many trees and branches similar in shape as a human, IHD occurred less when the three selected features were used in the SVDD analysis.

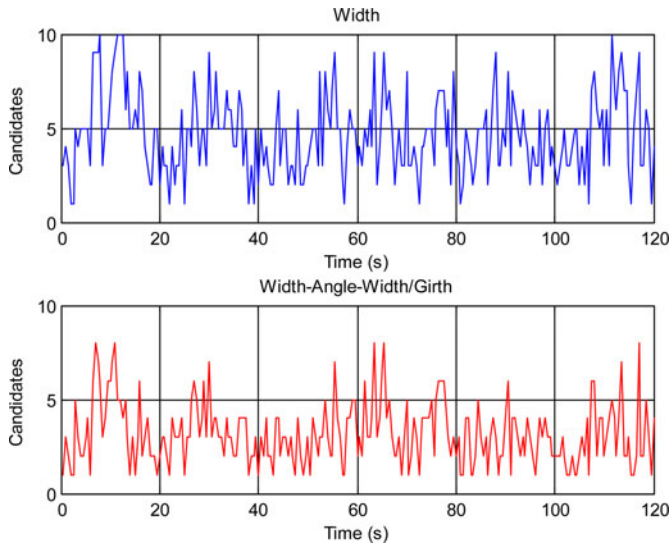


Fig. 7. Comparison of the number of detected human candidates with different feature selections. The top graph shows the number of detected human candidates when only *Width* feature is used for SVDD. The bottom graph shows the number of detected human candidates when *Width*, *Angle*, and *Width/Girth* features are used.

These experimental results indicate that *Width*, *Angle*, and *Width/Girth* can be used to describe human data collected by a LRF. Therefore, we applied SVDD with these three features for human tracking with an LRF.

In some cases, objects that have a similar size and shape to humans can be mistakenly recognized as human bodies. However, counting the exact number of human bodies is not the primary goal of this paper. Tracking a single marathoner and avoiding collisions with other marathoners or the environment are our primary focus. Small errors can be overcome by the following estimation and data association methods.

D. State Estimation

Tracking a running person is a challenging task. The difficulties in tracking a human with a mobile robot equipped with a laser range finder are due to uncertainty and sensor noise created by fast movement of the human body in an unstructured outdoor environment. To track human bodies robustly against these difficulties, we used a Kalman filter [21], [22], a well-known

Bayesian estimator, to estimate the state variables of human movement. The state variable vector $\underline{\mathbf{x}}$ is given by

$$\underline{\mathbf{x}} = (\, x \quad y \quad \dot{x} \quad \dot{y} \,)^T \tag{2}$$

and this equation includes positional and velocity information about a human body. x and y denote the x- and y-directional components in the global reference frame, respectively.

Modeling the motion of the human body is a difficult task because most of the time, a person's behavior is unpredictable. Many human tracking applications are based on a simple Brownian model [23], but to handle occlusions, a constant velocity model is a better choice [24], [25]. The body motion model was designed as a constant velocity model in discrete time, and is written as follows:

$$\underline{\mathbf{x}}_{k+1} = \mathrm{F}_k \underline{\mathbf{x}}_k + \underline{w}_k \tag{3}$$

where F_k is the state transition function and is given as

$$\mathrm{F}_k = \begin{bmatrix} \mathrm{I} & T\mathrm{I} \\ \underline{\mathbf{0}} & \mathrm{I} \end{bmatrix} \tag{4}$$

where T denotes the sampling time and $\underline{w}_k$ describes the process noise that is assumed to be zero mean Gaussian white noise with variance σ_w^2; $\underline{w}_k \sim N(0, \sigma_w^2)$. To get the parameter σ_w^2, we performed an experiment to measure the acceleration of human running motions by using the odometry and the LRF of the MSR. Therefore, the obtained acceleration includes odometry errors. Because the normal odometry noise is low frequency [26], the odometry errors are negligible when updating the states with typical sampling times of $T < 10$ ms in our practical experience.

To observe the sequential movement of subjects in a cluttered environment, data from the sensor should be correctly assigned to the proper human body, which is called data association. Then, the positions of the estimated human bodies are updated at each sampling time. The general scheme of data association has been illustrated [27]. We applied the nearest neighbor (NN) standard filter [28], which consists of choosing the closest pairs. That is, one measurement was assigned to one prediction at each update step.

III. Control Algorithm

A. Target Setup

Because there could be many human body candidates in the scanning range, the target body needs to be recognized among the candidates. Therefore, we initially set the target before tracking. The MSR estimates the positions of all candidates by applying a separate Kalman filter for each candidate. If a candidate, far from the desired tracking distance, is set as the target, the MSR might move to the target immediately at high acceleration and high speed to compensate for the tracking distance error. To ensure safe movement of the robot in the target setup process, we recognize the target when any candidate is located at 1 ± 0.1 m in front of the MSR. One meter is the desired tracking distance in our system. After the target setup process, the candidate becomes a target, and the target can be identified by the data association method (NN standard filter). Therefore, even though there are other people around the target person, the MSR can track the only target person.

B. Human Following

To follow a person, the MSR needs to be able to control its linear velocity and angular velocity. The distance between the MSR and the target person is controlled by the linear velocity to keep the desired safe distance. The angle between the heading direction of the MSR and the direction of the target person with respect to the MSR local coordinate is controlled by the angular velocity of the MSR.

To maintain the desired tracking distance between the MSR and the target, a typical PID control, Fuzzy control [29], adaptive control [30], and robust control [31] can be used. However, a typical proportional controller may be suitable for the marathoner following task. It is well known that applying only a proportional controller does not guarantee settling at the desired value, because steady-state errors are retained. Therefore, we used these steady-state errors as the safe distance between the target marathoner and the MSR. The following equation describes the proportional controller applied to the mobile robot

$$v = K_{Pv}(D_a - D_d) \tag{5}$$

where v is the linear velocity of the MSR, K_{Pv} is the proportional gain, and D_a and D_d denote the actual distance between the MSR and the target and the desired tracking distance, respectively. K_{Pv} is an indicator of safety margin since a small K_{Pv} results in a large tracking distance. Equation (5) indicates that the linear velocity of the mobile robot increases as the tracking distance increases. Conversely, tracking distance increases as the linear velocity increases. By using this fact, the mobile robot automatically guarantees a longer safe tracking distance as the target moves faster. When the marathoner walks or runs slowly, the MSR maintains a short tracking distance, and when the person runs fast, the MSR maintains a long tracking distance for safety. Even if the MSR has sufficient acceleration capacity to stop immediately, fast movement of the robot behind the person may frighten the person. Therefore, we used a proportional controller to ensure maintenance of a safe tracking distance.

The main function of the MSR is to follow a marathoner. Therefore, the angle between the heading direction of the MSR and the direction of the marathoner should be minimized. To compensate for angle error, a typical PID control was applied to the system. In addition, by approaching a hand to the LRF within 20 cm, the user can make the MSR stop tracking.

C. Obstacle Avoidance

To use the MSR in a real application, the robot should not only follow a marathoner, but also avoid other objects and other people for safety. Obstacle avoidance algorithms for a robot that moves at a human running speed should consider the motion of the robot and the objects in the environment as [32], [33]. Therefore, to ensure safety, we developed an obstacle avoidance algorithm that takes into account the velocity of obstacles relative to the MSR. For the marathoner following task, we assumed that the marathoner runs on a flat paved road or a jogging track.

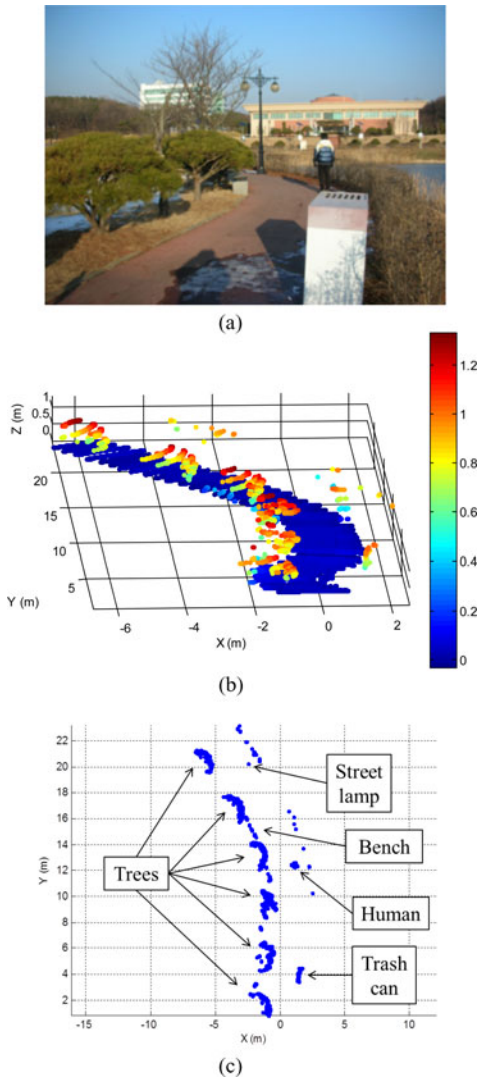


Fig. 8. Obstacle detection in 3-D. (a) Original scene image. (b) Obtained data from a leaned LRF in the outdoor environment shown in (a). (c) Projected data of (b) on the ground after cutting off data under 5 cm in height.

1) Obstacle Detection in 3-D: An original image of the test environment is shown in Fig. 8(a). To detect obstacles in 3-D, we employed an additional laser range finder inclined downward by 30° from the horizontal plane as shown in Fig. 12. This sensor detects bumps or small obstacles on the ground, while the original LRF detects persons and some objects in the test environment. Fig. 8(b) shows 3-D data of a pedestrian passage obtained from the additional laser range finder. We select the data 5 cm higher relative to the flat ground considering the sensor errors. Then, we project this data onto the horizontal plane of the original laser range finder, and merge these two LRF datasets. Therefore, all obstacles in 3-D near the MSR can be obtained in 2-D, as shown in Fig. 8(c). Then, we applied the obstacle avoidance algorithm to the projected 2-D map.

2) Weighted Radius of an Obstacle: There can be many obstacles around the MSR and the marathoner, but from the point of view of the MSR, the goal point is 1 m behind the marathoner in the marathoner following task. Therefore, all objects farther than the marathoner do not need to be considered, and if there is no obstacle between the MSR and the marathoner, the MSR just follows the marathoner. In a real environment, there are various obstacles that have various shapes and sizes and move at various velocities. To deal with the various obstacles that may be encountered by a mobile robot that moves at a human running speed, we applied the same Kalman filter model to each clustered dataset in order to estimate its state vectors (positions and velocities). If the position and the velocity of each obstacle are considered, a mobile robot, which moves at a human running speed, can generate a safer path compared to considering only positions of obstacles.

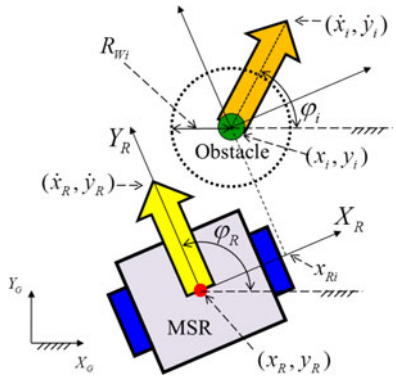


Fig. 9. Definition of velocities and positions of the MSR and an obstacle with respect to the global reference frame.

To make a safe path around obstacles, we define a weighted radius R_{W_i} in Fig. 9. The weighted radius is a virtual radius of an obstacle that depends on the relative velocity between the MSR and each obstacle. In previous work [18], [33], only the y-directional relative velocity was considered for the weighted radius. However, the x-directional relative velocity is also important to ensure safety because the moving obstacle may change its motion in any direction. Thus, in this paper, we modified the weighted radius that considers both x- and y-directional velocities for obstacle avoidance. The relative velocity components of the ith obstacle in the x- and y-directions of the MSR coordinate system are expressed respectively by

$$\begin{cases} v_{yi} = v_R - v_i \cos(\varphi_R - \varphi_i) \\ v_{xi} = v_i \sin(\varphi_R - \varphi_i) \end{cases} \tag{6}$$

where v_R and v_i, respectively, represent the robot velocity and the obstacle velocity given by

$$\begin{cases} v_R = \sqrt{\dot{x}_R^2 + \dot{y}_R^2} \\ v_i = \sqrt{\dot{x}_i^2 + \dot{y}_i^2} \end{cases}. \tag{7}$$

In (6), φ_R and φ_i are the angles of the heading direction of the MSR and the angle of the velocity vector of the obstacle, respectively, as shown in Fig. 9. $\dot{x}_R$ and $\dot{y}_R$ are the velocities of the MSR in the x- and y-directions, respectively, with respect to the global coordinate system. $\dot{x}_i$ and $\dot{y}_i$ are the velocities of the ith obstacle in the x- and y-directions, respectively, with respect to the global coordinate system.

If the Y_R-directional velocity $(v_i \cos(\varphi_R - \varphi_i))$ of an obstacle is smaller than that of the robot ($v_{yi} < 0$), the obstacle

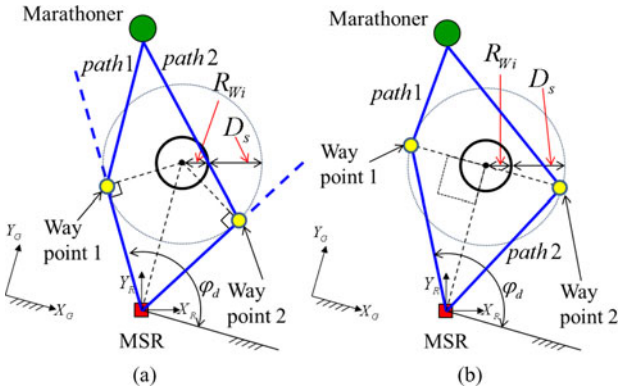


Fig. 10. Desired angle of the MSR with a weighted radius. In the previous obstacle avoidance algorithm, tangential lines of the circle are used for the desired heading direction of the MSR. In the modified obstacle avoidance algorithm, the way points are the intersections of the circle with a radius of $R_{W_i} + D_s$ and the line that crosses the center of the circle and is perpendicular to line from the MSR to the center of the circle.(a) Previous obstacle avoidance algorithm in [18]. (b) Modified obstacle avoidance algorithm.

should be avoided. On the other hand, an obstacle moving faster than the mobile robot is not considered as an obstacle since it has less chance to collide with the mobile robot. If an obstacle gets closer to the mobile robot in the X_R direction, the radius should be bigger. Otherwise, it should be smaller. Therefore, combining these ideas, the modified weighted radius of the ith obstacle is defined as follows:

$$R_{Wi} = \begin{cases} 0 & v_{yi} \le 0 \\ K_y v_{yi} - K_x v_{xi} \text{sign}(x_{Ri}) + R_i & v_{yi} > 0 \end{cases} \quad (8)$$

where K_y and K_x are scaling factors between the velocity and the radius, and x_{Ri} is the position of the obstacle in the X_R-direction of the robot coordinate system and is calculated as

$$x_{Ri} = \sqrt{(x_R - x_i)^2 + (y_R - y_i)^2}\,\sin(\varphi_R - \varphi_i) \quad (9)$$

where R_i is the original radius of the ith obstacle. When an obstacle is moving faster than the MSR, the weighted radius becomes zero, which means that the obstacle is not considered as an obstacle. If R_{W_i} is smaller than R_i, it means that the obstacle is moving away from the MSR.

We assumed that the original radius of each obstacle is the same as half the width of the clustered data. If the width of the obstacle was larger than 500 mm, we split the clustered data from the test environment into many sections of 500 mm or less width. The dotted circle in Fig. 9 shows an example of the weighted radius applied to an obstacle. Obstacles slower than the MSR or moving to the MSR are assigned to a bigger radius.

3) Obstacle Avoidance: If any obstacle is located between a marathoner and the MSR and is moving slower than the MSR, avoiding the obstacle should be attempted first. Otherwise, the MSR keeps tracking the marathoner.

For obstacle avoidance, we search for an obstacle which is closest to the MSR and calculate the weighted radius R_{W_i} of the obstacle (8). Then, the desired heading direction φ_d of the MSR can be geometrically determined by searching for the shortest path among the two possible paths shown in Fig. 10. D_s denotes the safe distance determined by the size of the MSR.

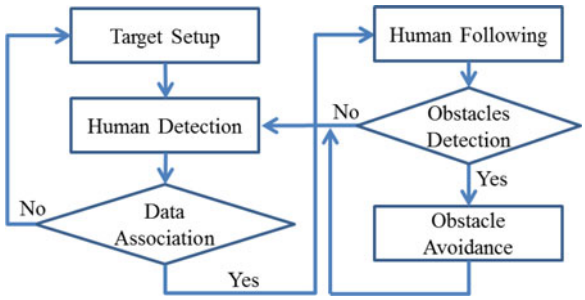


Fig. 11. Flowchart of the control algorithms.

Such paths are not predefined, but these are instantaneously created according to the conditions of the MSR and obstacles. Therefore, the path changes at every sampling time because the velocities and the positions of the MSR and the obstacles always change. Thus, this path is only used for determining the desired heading direction of the MSR at every sampling time. In previous work [18], we used a line that is tangent to the circle with a radius of $R_{W_i} + D_s$ and is connected to the MSR as shown in Fig. 10(a). In this case, as the MSR comes closer to the obstacle, the desired heading angle φ_d is dramatically increased or decreased, which results in a slip in the wheels and odometry errors. To cope with this problem, in this paper, we modified the waypoints on the circle of Fig. 10(b) such that a line connecting two waypoints is perpendicular to another line from the MSR to the center of the circle. In this algorithm, even though the MSR comes close to an obstacle, a dramatic change in the desired heading angle φ_d does not happen.

As velocity of the MSR relative to the obstacle increases, obstacle avoidance action is taken earlier. Consequently, the path of the MSR surrounding the obstacle gets longer, which is explained by a larger weighted radius. Thus, obstacles can be avoided earlier if the weighted radius is large, which means that the relative velocity is large. Then, the MSR should take the shortest path around the obstacle. However, if an obstacle has a zero weighted radius, it is not considered as an obstacle because in this case, the obstacle is faster than the MSR. By applying this obstacle avoidance algorithm with the weighted radius, the MSR can avoid not only static obstacles but also moving obstacles while tracking a marathoner. Fig. 11 shows a flowchart of the control algorithms.

IV. Experiments

A. Marathoner Service Robot

Fig. 12 shows the MSR that we designed to follow an amateur marathoner and to carry personal belongings. We designed the robot to have a maximum speed of 6.0 m/s and to carry loads of up to 5 kg. To allow the robot to absorb some impulses from the irregular terrain in the outdoor environment, we implemented a suspension system at the front caster wheel of the robot platform. The MSR is equipped with one UTM-30LX (HOKUYO Corporation) that covers the angular range of 270° within 30 m for detecting humans, and one URG-04LX (HOKUYO Corporation) installed on the tilt that covers the angular range of 240°

Fig. 12. Marathoner Service Robot equipped with two LRFs for human and obstacle detection.

TABLE III
SPECIFICATIONS OF THE LASER RANGE FINDERS

Model	UTM-30LX (HOKUYO Co.)	URG-04LX (HOKUYO Co.)
Measurement Range	20~30000 mm	60~4000 mm
Measurement Accuracy	±30 mm	±10 mm
Angular Range	270°	240°
Angular Resolution	0.25°	0.36°
Sampling Time	25 ms	100 ms

TABLE IV
SPECIFICATION OF THE MARATHONER SERVICE ROBOT

Motor	Maxon RE50 (200W) × 2
Max Speed	6.0 m/s
Max Acceleration	10 m/s^2
Weight	23 kg (Include batteries)
Payload	5 kg
W × *D* × *H*	560 × 630 × 700-1100 mm
Wheel radius	130 mm
Battery	Lead-acid (12V, 4.0Ah) × 4
Power Consumption	60W (Normal tracking)
Duration	About 3 hours

within 4 m for detecting obstacles on the ground. The specifications of the two laser range finders are shown in Table III and those of the MSR are shown in Table IV.

B. Experiment 1: Tracking Speed

Initially, the tracking performance of the MSR using the proportional controller given in (5) was tested in an outdoor environment. In this experiment, the proportional gain K_{Pv} and the desired tracking distance D_d in (5) are set as 2 and 1 m, respectively. A target person ran in a straight line for 60 s while the MSR kept tracking a target person.

Fig. 13 shows the results of tracking performance at human running speed. In the figures, we can see that as the speed of the target person increased, the tracking distance between the MSR and the target body is increased for safety. The MSR maintained a certain distance according to the tracking algorithm (5). In the beginning of the tracking experiment, the initial distance between the MSR and the target person is set at 1 m. However, the tracking distance increases to more than 3.5 m as the speed

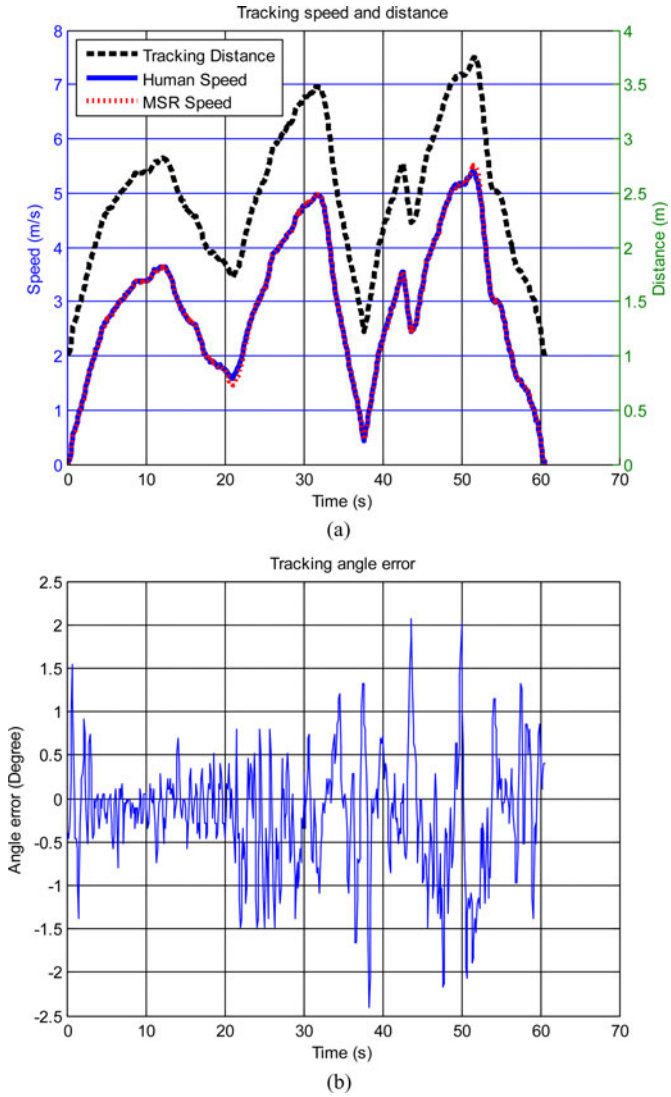


Fig. 13. Experimental result of human tracking at human running speed. (a) Tracking speed and distance of the MSR and the speed of a target person. (b) Tracking angle error of the MSR.

of the target person increases up to 5 m/s at 32 s. The maximum speed of the MSR in this experiment is 5.5 m/s at 52 s, which is faster than our goal speed. There is a little delay between the speed of the MSR and that of the target person, but those speeds are almost identical.

Minimizing the angular difference between the heading angle and the desired angle of the MSR is one of the important measures for tracking a person. The results of minimizing the angular error are shown in Fig. 13(b). It is noted that when the MSR has relatively high acceleration or high deceleration, the tracking angle error increases. This is due to skidding of the wheels of the MSR at those moments. However, the magnitude of tracking angle error of the MSR was less than 3°, which is acceptable in the marathoner following task. Furthermore, a certain distance was maintained between the MSR and the marathoner, which ensured the marathoner's safety.

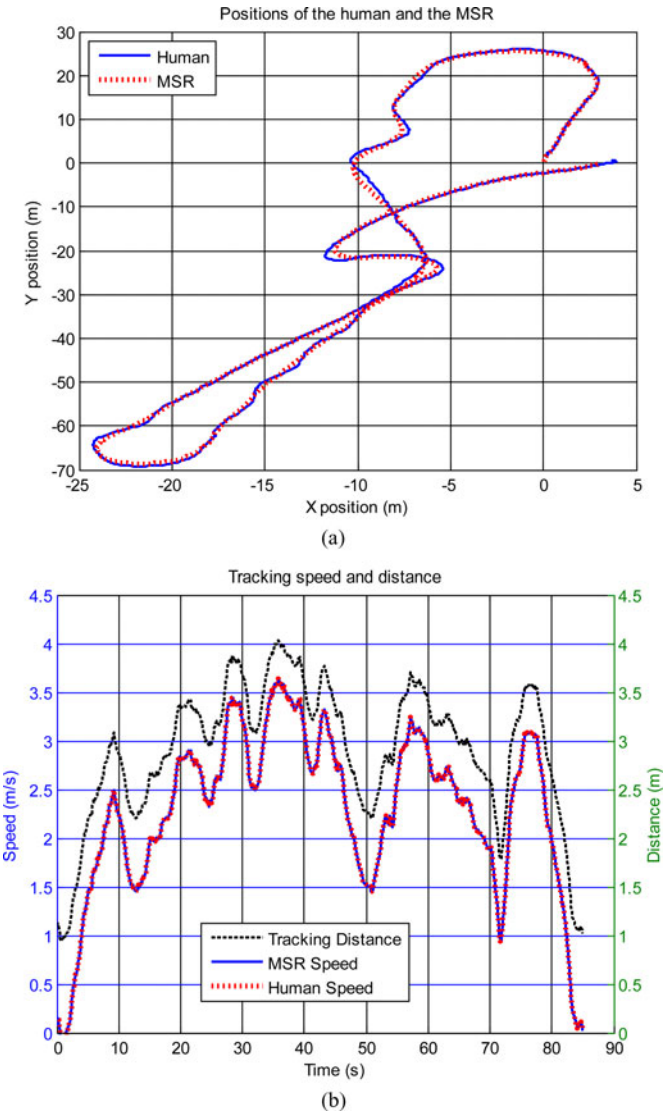


Fig. 14. Experimental result of human tracking with various velocities. (a) Paths of the target person and the MSR. (b) Tracking speed and distance of the MSR and the speed of the target person.

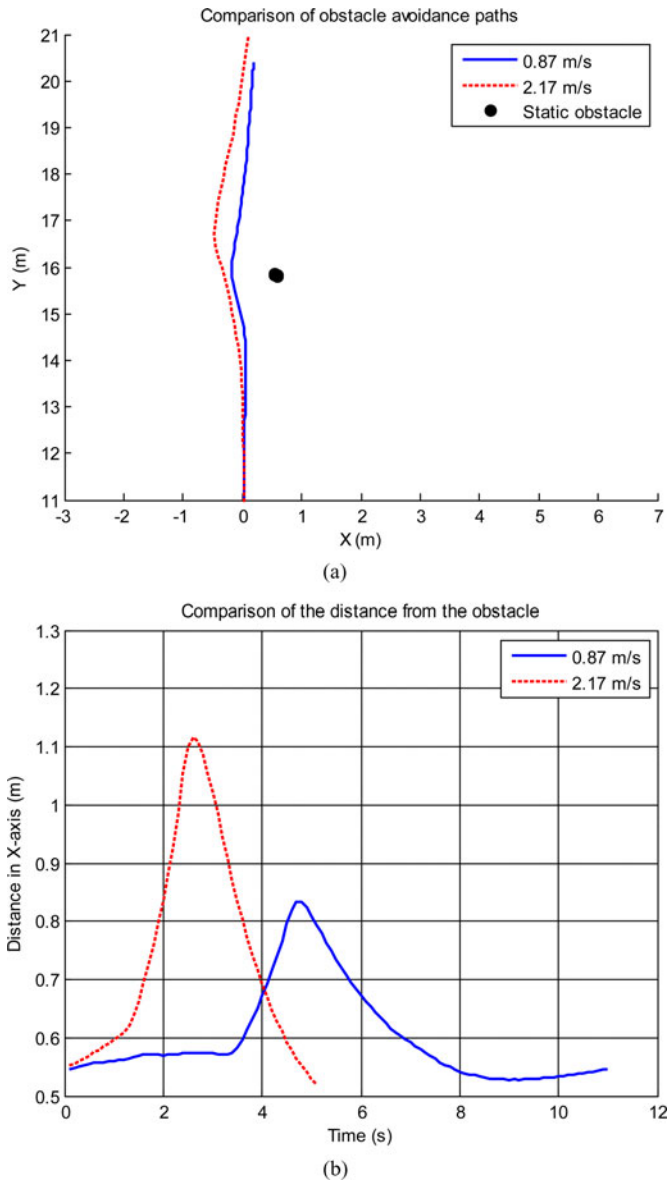


Fig. 15 Obstacle avoidance paths of the MSR and distance from the obstacle while tracking a target person.(a) Paths of the MSR avoiding static obstacles at different speeds. (b) Distance from the obstacle in the x-axis.

C. Experiment 2: Tracking Performance

In the second experiment, a target person ran in a curved line with varying velocities in an outdoor environment to show the tracking performance of the MSR. In this experiment, the proportional gain K_{Pv} in (5) is set at 1.2. Fig. 14 shows the results of the experiment. For 85 s, the target human ran more than 200 m and the maximum tracking speed of the MSR was 3.5 m/s (average: 2.9 m/s). Even though the target person ran in a curved path, the MSR could keep tracking the target person and maintain the tracking distance as the speed of the target person varied.

D. Experiment 3: Static Obstacle Avoidance

To show the effectiveness of the proposed obstacle avoidance algorithm, we perform two experiments that applied the modified weighted radius for two different speeds of a target person. In these experiments, initial positions of the MSR and a target person are set as (0, 0) m and (0, 1) m in the global coordinate of Fig. 15(a), respectively, and the position of a static obstacle is at (0.3, 7.5) m. Then, the goal position of the target person is set as (0, 12) m.

Fig. 15(a) shows the paths of the MSR resulting from target tracking in two experiments. In the first experiment, the target person runs in a straight line at an average speed of 0.87 m/s. The scaling factors of the weighted radius, K_y and K_x in (8), are set as 0.25 and 1, respectively. As a result, the MSR made a small circular path (blue solid line) around the obstacle as shown in Fig. 15(a). It is noted that the MSR passed by the obstacle at most 0.84 m (collision-free distance) from the obstacle as shown in Fig. 15(b).

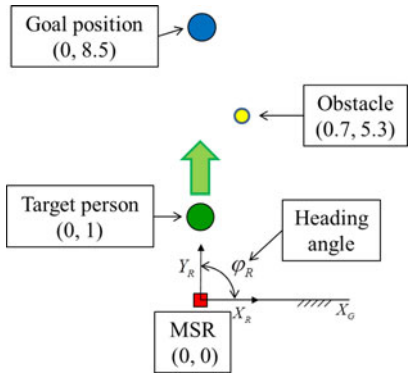


Fig. 16. Experimental environment.

In the second experiment, the target person ran in a straight line at an average speed of 2.17 m/s. As a result, the MSR made a bigger circular path (red dashed line) with a collision-free distance of 1.12 m around the obstacle as shown in Fig. 15(b), which indicates that as the speed of the target person increases, the MSR passes far away from the obstacle to ensure safety.

The above results confirm that the modified weighted radius algorithm, which considers the relative velocities between the MSR and obstacles, can help prevent collisions between the MSR and other human bodies or other objects in the outdoor environment.

E. Experiment 4: Comparison of Obstacle Avoidance Algorithms

In this section, we compare the two obstacle avoidance algorithms shown in Fig. 10. Fig. 16 shows the experimental environment. A target person moves from (0, 1) to (0, 8.5) m in the global coordinate system and runs at an average speed of 1 m/s. There is one obstacle located at (0.7, 5.3) m. We tested, respectively, the previous obstacle avoidance algorithm [see Fig. 10(a)] and the modified obstacle avoidance algorithm [see Fig. 10(b)] five times. The modified weighted radius of (8) is applied to both algorithms. In these experiments, the scaling factors of the weighted radius, K_y and K_x, are set as 0.25 and 1, respectively.

In these experiments, initially the desired heading angle of the MSR with respect to the global coordinate is 90° as shown in Fig. 16. If an obstacle is detected between the target person and the MSR, the MSR finds a new desired heading angle to avoid the obstacle using the obstacle avoidance algorithms.

Fig. 17 shows desired heading angles and paths of the MSR for the above two algorithms. In the figures, "Plan1" and "Plan2" stand for the previous obstacle avoidance algorithm and the modified obstacle avoidance algorithm, respectively. The speed in parentheses denotes the average speed of a target person during each test.

Fig. 17(a) describes the change in the desired heading angle of the MSR with respect to the global reference frame for each experiment. When the previous obstacle avoidance algorithm is applied, the desired direction is dramatically changed. Thus, there will be more chance for the MSR to have slip at the wheels and result in odometry error. On the other hand, when the modified obstacle avoidance algorithm is applied, the desired direction is changed smoothly.

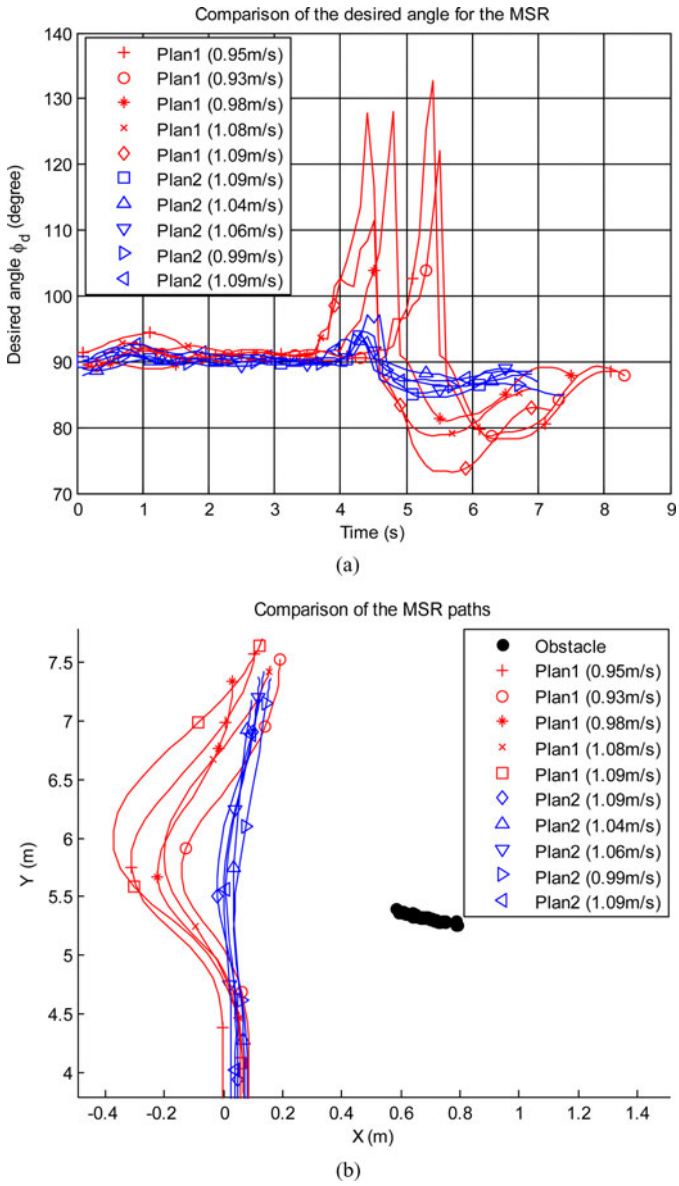


Fig. 17. Results of obstacle avoidance paths of the MSR and desired heading angle of the MSR while avoiding a static obstacle. (a) Comparison of the desired heading angle of the MSR when an obstacle is detected. (b) Comparison of the paths of the MSR with two obstacle avoidance algorithms to avoid static obstacles.

Fig. 17(b) demonstrates the paths of the MSR for two obstacle avoidance algorithms. The MSR made a big circular path around the obstacle when the previous obstacle avoidance algorithm was applied, but when the modified obstacle avoidance algorithm was applied, the MSR passed by the obstacle safely and without collision.

F. Experiment 5: Avoidance of Moving Obstacle

In experiment 5, we extend the concept of the modified weighted radius to a moving obstacle. The scenario is as follows.

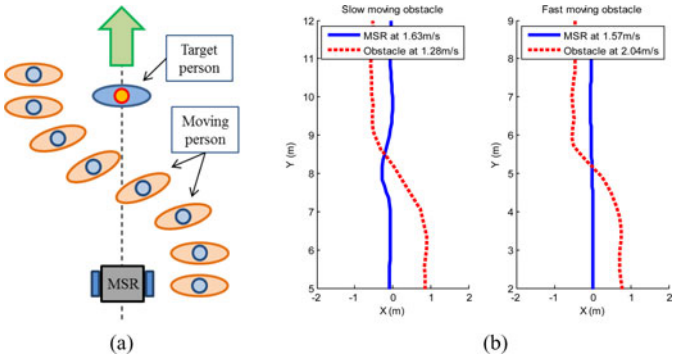


Fig. 18. Moving obstacle avoidance experiments. A target person runs straight and the MSR follows the target. Then, an intruder crosses between the target person and the MSR, which are proceeding at different speeds. (a) Conditions of the experiments. (b) Moving path of the intruder and tracking path of the MSR.

When the MSR is following a target person running in a straight line, an intruder crosses the path between the target person and the MSR as shown in Fig. 18(a). We consider two cases in which the intruder is moving either faster or slower than the MSR. Then, we compare paths of the MSR.

Fig. 18(b) demonstrates the paths of the MSR for the two cases. In the first case, the MSR moves at an average speed of 1.63 m/s, and the intruder moves at an average speed of 1.28 m/s. The MSR considers the intruder as an obstacle according to the modified weighted radius (8) because the intruder is slower than the MSR. The left graph of Fig. 18(b) shows that when the intruder comes closer, the MSR turns left to avoid the intruder. After the intruder crosses the path between the target person and the MSR, the MSR turns right to avoid the intruder according to the modified obstacle avoidance algorithm.

In the second case, the MSR moves at an average of 1.57 m/s, and the intruder moves at an average of 2.04 m/s. The right graph of Fig. 18(b) shows the result of the second experiment. When the intruder is faster than the MSR, the MSR does not consider the intruder as an obstacle. Therefore, the MSR did not show any avoidance behavior. By applying the modified weighted radius to the moving obstacle, the MSR can avoid moving obstacles while tracking a marathoner.

G. Experiment 6: Marathoner Tracking

In this experiment, the MSR was tested in the outdoor environment around the Hanyang university campus in a 700 × 500 m area. Fig. 19 demonstrates that the MSR was able to track a runner for a long distance, showing good tracking performance. The MSR provides the marathoner with useful training information such as running distance, GPS log data, average speed, maximum speed, and running time. The GPS log data in map format are shown in Fig. 19, and Table V shows the information obtained from this experiment. The marathoner ran 2378.24 m in 815.1 s with average and maximum speeds of 2.92 and 4.75 m/s, respectively.

There are a few cases in which the MSR fails to track a marathoner. On a real road, there are various speed bumps. The MSR can miss a target at a big bump because the human

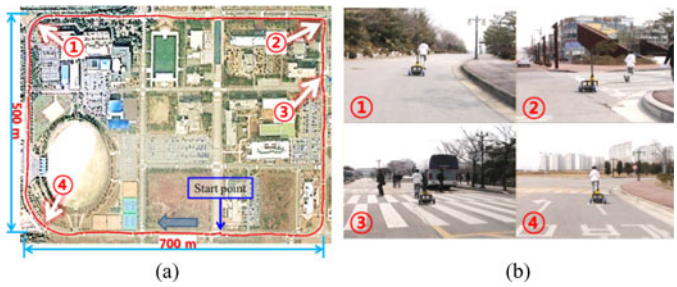


Fig. 19. Red line on the map denotes GPS log data while the MSR follows a running person. The path of this experiment is longer than 2 km. (a) GPS log data of the running path. (b) Detailed tracking environment.

TABLE V
TRAINING LOG DATA FROM THE MSR

Running Distance	2378.24 m (2.378 km)
Average speed	2.92 m/s (10.5 km/h)
Maximum speed	4.75 m/s (17.1 km/h)
Running Time	815.1 s (13 m 35.1 s)

tracking algorithm of the MSR in the global coordinate is based on the odometry. In our experiments at speed bumps, the rate of the failure was 3%. By using a gyro sensor, the tracking performance of the MSR can be improved. Another case occurs when the mobile robot misses the target person when another person blocks the robot for a long time. Even though this case does not happen often during a training, when the MSR loses a target, it stops tracking for safety and waits on the target setup step.

V. CONCLUSION

We investigated a marathoner service robotic system moving at a human running speed. The first contribution is the SVDD method-based feature selection using 2-D data from a laser range finder to perform better human detection in an uncertain and cluttered outdoor environments. The second contribution is the implementation of a weighted radius, which enables the MSR to avoid moving obstacles by considering the relative velocity between the MSR and the closest obstacle. The last contribution is the integration of the core technologies mentioned above and applying them to a MSR. Using a marathoner following experiment in an outdoor environment, we have shown that our MSR has satisfactory maximum speed and performs a human tracking task stably without collisions. Our future work involves enhancing the mechanical reliability of our robot platform and upgrading the software algorithms for enhanced safety.

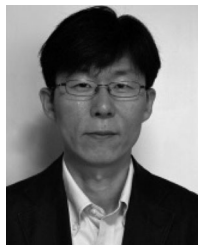
REFERENCES

[1] MarathonWorld. (2012). [Online]. Available: http://www.marathon-world.com
[2] I. J. Cox and S. L. Hingorani, "An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 2, pp. 138–150, Feb. 1996.
[3] J. K. Aggarwal and Q. Cai, "Human motion analysis: A review," in *Proc. IEEE Proc. Nonrigid Articulated Motion Workshop*, 1997, pp. 90–102.

[4] J. MacCormick and A. Blake, "A probabilistic exclusion principle for tracking multiple objects," in *Proc. 7th IEEE Int. Conf. Comput. Vision*, 1999, pp. 572–578.
[5] C. Rasmussen and G. D. Hager, "Probabilistic data association methods for tracking complex visual objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6,, pp. 560–576, Jun. 2001.
[6] B. Jung and G. S. Sukhatme, "Detecting moving objects using a single camera on a mobile robot in an outdoor environment," in *Proc. 8th Conf. Intell. Auton. Syst.*, 2004, pp. 980–987.
[7] B. Kluge, C. Kohler, and E. Prassler, "Fast and robust tracking of multiple moving objects with a laser range finder," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2001, pp. 1683–1688.
[8] A. Fod, A. Howard, and M. A. J. Mataric, "A laser-based people tracker," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, pp. 3024–3029.
[9] H. Zhao and R. Shibasaki, "A novel system for tracking pedestrians using multiple single-row laser-range scanners," *IEEE Trans. Syst., Man Cybern., A, Syst. Humans*, vol. 35, no. 2, pp. 283–291, Mar. 2005.
[10] J. H. Lee, T. Tsubouchi, K. Yamamoto, and S. Egawa, "People tracking using a robot in motion with laser range finder," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2006, pp. 2492–6392.
[11] K. O. Arras, O. M. Mozos, and W. Burgard, "Using boosted features for the detection of people in 2D range data," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2007, pp. 3402–3407.
[12] R. Urtasun, D. J. Fleet, and P. Fua, "3D People tracking with Gaussian process dynamical models," in *Proc IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit.*, 2006, pp. 238–245.
[13] L. Spinello, M. Luber, and K. O. Arras, "Tracking people in 3D using a bottom-up top-down detector," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 1304–1310.
[14] C. Wongun, C. Pantofaru, and S. Savarese, "Detecting and tracking people using an RGB-D camera via multiple detector fusion," in *Proc. IEEE Int. Conf. Comput. Vision Workshops*, 2011, pp. 1076–1083.
[15] M. Kobilarov, G. Sukhatme, J. Hyams, and P. Batavia, "People tracking and following with mobile robot using an omnidirectional camera and a laser," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 557–562.
[16] N. Bellotto and H. Huosheng, "Multisensor-based human detection and tracking for mobile service robots," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 39, no. 1, pp. 167–181, Feb. 2009.
[17] D. F. Glas, T. Miyashita, H. Ishiguro, and N. Hagita, "Laser tracking of human body motion using adaptive shape modeling," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2007, pp. 602–608.
[18] J. Eui-Jung, L. Jae Hoon, Y. Byung-Ju, S. Il Hong, S. Yuta, and N. Si Tae, "Marathoner tracking algorithms for a high speed mobile robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2011, pp. 3595–3600.
[19] D. M. J. Tax and R. P. W. Duin, "Support vector domain description," *Pattern Recognit. Lett.*, vol. 20, pp. 1191–1199, 1999.
[20] F. Guoqiang, P. Corradi, A. Menciassi, and P. Dario, "An integrated triangulation laser scanner for obstacle detection of miniature mobile robots in indoor environment," *IEEE/ASME Trans. Mechatronics*, vol. 16, no. 4, pp. 778–783, Aug. 2011.
[21] G. Welch and G. Bishop, *An Introduction to the Kalman Filter*. Chapel Hill, NC, USA: Univ. of North Carolina Press, vol. 7, 1995.
[22] S. Xiaojing, L. D. Seneviratne, and K. Althoefer, "A Kalman filter-integrated optical flow method for velocity sensing of mobile robots," *IEEE/ASME Trans. Mechatronics*, vol. 16, no. 3, pp. 551–563, Jun. 2011.
[23] M. Montemerlo, S. Thrun, and W. Whittaker, "Conditional particle filters for simultaneous mobile robot localization and people-tracking," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, pp. 695–701.
[24] D. Beymer and K. Konolige, "Tracking people from a mobile platform," *Exp. Robot. VIII*, vol. 5, pp. 234–244, 2003.
[25] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers, "People tracking with mobile robots using sample-based joint probabilistic data association filters," *Int. J. Robot. Res.*, vol. 22, pp. 99–116, 2003.
[26] J. Borenstein and L. Feng, "Measurement and correction of systematic odometry errors in mobile robots," *IEEE Trans. Robot. Autom.*, vol. 12, no. 6, pp. 869–880, Dec. 1996.
[27] D. L. Hall and S. A. H. McMullen, *Mathematical Techniques in Multisensor Data Fusion*. Norwood, MA, USA: Artech House, 2004.
[28] Y. Bar-Shalom and X. R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*. Storrs, CT, USA: Univ. of Connecticut, 1995.
[29] K. Tanaka, M. Tanaka, H. Ohtake, and H. O. Wang, "Shared nonlinear control in wireless-based remote stabilization: A theoretical approach," *IEEE/ASME Trans. Mechatronics*, vol. 17, no. 3, pp. 443–453, Jun. 2012.
[30] Y. Bin, H. Chuxiong, and W. Qingfeng, "An orthogonal global task coordinate frame for contouring control of biaxial systems," *IEEE/ASME Trans. Mechatronics,*, vol. 17, no. 4,, pp. 622–634, Aug. 2012.
[31] M. Fallah, R. B. Bhat, and X. Wen Fang, "Optimized control of semiactive suspension systems using H-infinity robust control theory and current signal estimation," *IEEE/ASME Trans. Mechatronics*, vol. 17, no. 4, pp. 767–778, Aug. 2012.
[32] A. R. Willms and S. X. Yang, "An efficient dynamic system for real-time robot-path planning," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 36, no. 4, pp. 755–766, Aug. 2006.
[33] I. Mas and C. Kitts, "Obstacle avoidance policies for cluster space control of nonholonomic multirobot systems," *IEEE/ASME Trans. Mechatronics*, vol. 17, no. 6, pp. 1068–1079, Dec. 2012.

Eui-Jung Jung received the B.S. degree in electrical engineering from Hanyang University, Ansan, Korea, in 2006, and the M.S. and Ph.D. degrees from the Department of Electronics, Electrical, Control and Instrumentation Engineering, Hanyang University, in 2008 and 2013, respectively.

His research interests include mobile robots, kinematics, and human tracking.



Jae Hoon Lee received the B.S. degree in mechanical and automatic control engineering from the Korea University of Technology and Education, Cheonan, Korea, in 1996, and the M.S. and Ph.D. degrees in electric, electrical, control and instrumentation engineering from Hanyang University, Ansan, Korea, in 1998 and 2003, respectively.

From 2004 to 2006, he was a Postdoctoral Fellow at the Intelligent Robot Laboratory, University of Tsukuba, Japan. From 2007 to 2008, he was a JSPS Fellow with the Ubiquitous Functions Research Group, Intelligent Systems Research Institute, AIST, Japan. He is currently an Associate Professor in the Department of Mechanical Engineering, Graduate School of Science and Engineering, Ehime University, Matsuyama, Japan. His current research interests include the collaborative navigation of mobile robots in the daily environment, the development of sensory systems to monitor bicycle riding situations, and the design and control of bioinspired robotic systems.

Byung-Ju Yi (M'89) received the B.S. degree from Hanyang University, Seoul, Korea, in 1984, and the M.S. and Ph.D. degrees from The University of Texas at Austin, Austin, TX, USA, in 1986 and 1991, respectively, all in mechanical engineering.

From January 1991 to August 1992, he was a Postdoctoral Fellow with the Robotics Group, The University of Texas at Austin. From September 1992 to February 1995, he was an Assistant Professor in the Department of Mechanical and Control Engineering, Korea Institute of Technology and Education, Cheonan, Chungnam, Korea. In March 1995, he joined the Department of Control and Instrumentation Engineering, Hanyang University, Seoul, Korea. Currently, he is a Professor with the Department of Electronic Systems Engineering, Hanyang University. He was a Visiting Professor at The Johns Hopkins University, USA, in 2004 and a JSPS Fellow at Kyushu University, Japan, in 2011. His research interests include general robot mechanics with application to surgical robotic systems (ENT, neurosurgical, and needle insertion areas), pipeline inspection robots, and ubiquitous sensor network-based robotics.

Dr. Yi is a member of the IEEE Robotics and Automation Society and served as an Associate Editor of the IEEE TRANSACTIONS ON ROBOTICS from 2005 to 2008. Currently, he is serving as a Board Member of the Korean Robotics Society, Korean Society of Medical Robotics, and International Society of Computer-Aided Surgery.

Jooyoung Park received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1983 and the Ph.D. degree in electrical and computer engineering from the University of Texas at Austin, Austin, TX, USA, in 1992.

He joined Korea University, Seoul, Korea, in 1993, where he is currently a Professor in the Department of Control and Instrumentation Engineering. His recent research interests include the area of kernel methods, reinforcement learning, and control theory.

Shin'ichi Yuta (F'00) received the Ph.D. degree in electrical engineering from Keio University, Tokyo, Japan, in 1975.

In 1978, he had been at the University of Tsukuba and a Full Professor after 1992. He served as the Vice-President for research, in 2004–2006, and the Director of Tsukuba Industrial Liaison and Cooperative Research Center. In 2012 March, he retired from the University of Tsukuba, and is currently an Adjunct Professor in Shibaura Institute of Technology, Tokyo, Japan, as well as an Emeritus Professor at University of Tsukuba, Tsukuba, Japan. As an expert in robotics, he has conducted an autonomous mobile robot project, since 1980, and published more than 500 technical papers in this field. He has been keeping a close relationship and collaboration with many industries, for the developments of practical mobile robot systems and devices for intelligent robots. The typical achievements include the development of small size scanning laser range sensors for mobile robots, which is produced by Hokuyo, and now widely used in the world for robotics applications.

Dr. Yuta is a Fellow of the Robotics Society of Japan.

Si-Tae Noh received the B.S. and M.S. degrees from Hanyang University, Seoul, Korea in 1976 and 1978, respectively, all in polymer engineering, and also the Ph.D. degree in industrial chemistry in 1981.

From March 1987 to February 1992, he was a Postdoctoral Fellow with the University of Cincinnati. Since March 1982, he has been a Professor in the Department of Chemical Engineering, Hanyang University ERICA, Ansan, Gyeonggi, Korea. His research interests include synthesis of binder material for solid propellant, block copolymer for organic electric device, UV curable high refractive oligomer, and the manufacture of specialty polymers desirable computer control and monitoring minipilot reaction system design.

Dr. Noh is a Member of the Korean Society of Industrial Engineering Chemistry and the Polymer Society of Korea.