Toward Optical Guidance of Mobile Robots

Igor E. Paromtchik, Hajime Asama
The Institute of Physical and Chemical Research (RIKEN)
Hirosawa 2-1, Wako-shi, Saitama 351-0198, Japan

Abstract

This paper describes our research toward the development of an optical guidance system for mobile robots. The guidance system projects a light from a laser pointer onto a floor surface to indicate a target position for the mobile robot. The light dot on the surface is detected by means of processing image data from a color CCD camera mounted on the robot. The robot's control system ensures the accurate motion of the robot to the indicated target position.

Keywords: Mobile Robot, Optical Guidance, Motion Control.

1 Introduction

Explicit robot localization based on internal sensors such as odometry or gyroscopes results in accumulation of positional errors because of inaccurate modeling and wheel slippage. External means (such as landmarks, beacons, etc.) and a fusion of sensor data are necessary in order to provide better accuracy. Various optical guidance methods are known: using reflective beacons, tracking stationary light sources, drawing a guidance line on the floor or ceiling surface, or using a scanning laser on the vehicle to measure distances to surrounding objects [1].

We propose an optical guidance system for the mobile robots that makes use of a laser light. The guidance system indicates the numerical coordinates of the target positions by means of projection of a laser light onto a floor surface. The on-board vision system of the robot processes the color image to detect the light dot and evaluate its coordinates relative to the mobile robot. This visual feedback ensures the accurate following of the indicated target positions by the mobile robot.

The main advantage of the proposed guidance system is the improved accuracy. The system allows for an implicit localization of the mobile robot in the environment: when the robot has reached its indicated target position, the robot's coordinates in the environmental model are known. At the same time, the transformation between the coordinate systems of the environmental model and that of the robot becomes irrelevant; the robot's control system operates with the relative coordinates of the target positions obtained from image processing.

Our paper focuses on the control architecture of the mobile robot and the motion generation approach developed for vision-based guidance by a laser pointing device. The paper is organized as follows. The key idea of the proposed optical guidance system is described in section 2. The mobile robot and its control architecture are presented in section 3. The motion generation approach is considered in section 4. The experimental results obtained on visual tracking of the mobile robots and its further extension to tracking a projected laser light are discussed in section 5. The conclusions are given in section 6.

2 The Optical Guidance System

The proposed optical guidance system indicates positions by means of projection of a laser light onto the coorresponding spot in the environment. The guidance system is sketched in Fig. 1 and involves a central computer and a laser pointer that directs an optical axis of the laser to any spot on the floor surface. The coordinates of the target position are computed by the central computer from the environmental model or are set by a human operator. The guidance system relies on a client-server wireless communication model between the central computer and the on-board control system of the robot. The robot's vision system detects the laser light dot and evaluates its relative coordinates.

Let (X_w, Y_w) denote a common coordinate system in Fig. 1, where (x_t, y_t) are coordinates of a target position for a mobile robot, and (x_0, y_0) are coordinates of the laser pointer. Let the laser pointer be situated at a hight h_0 from the floor surface and have at least two degrees-of-freedom (θ_1, θ_2). The light beam from the laser pointer will indicate the coordinates (x_t, y_t) on the floor surface, if an orientation (θ_1, θ_2) of the laser pointer is obtained from the following expressions:

$$\tan\theta_1 = (y_0 - y_t)/(x_0 - x_t), \tag{1}$$

$$\tan\theta_2 = \sqrt{(x_0 - x_t)^2 + (y_0 - y_t)^2}/h_0. \tag{2}$$

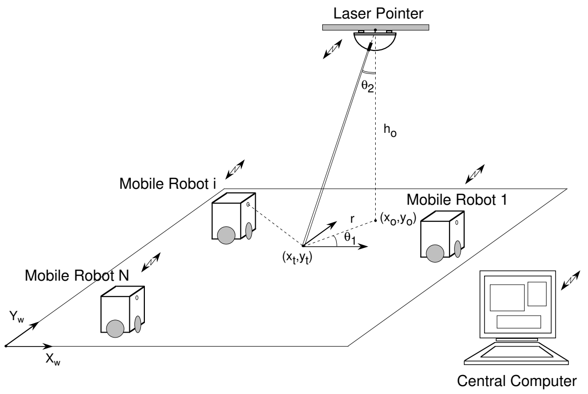

Figure 1: A sketch of the optical guidance system

The guidance system operates according to the following basic algorithm in order to indicate a target position for a mobile robot:

1. Establish a connection between the central computer and a control system of the mobile robot.

2. Transmit a request from the central computer whether the robot's control system is ready to process a new target position.

3. If the mobile robot is ready to get a new target position, then set the laser pointer in the appropriate orientation (θ_1, θ_2). Otherwise, go to step 2.

4. Turn on the light of the laser pointer in order to indicate the target position on the floor surface.

5. If the robot's control system confirms detection of the indicated target position, then turn off the laser light. Otherwise, wait until the confirmation or failure response is received from the robot's control system.

6. If the indicated target position could not be detected (failure response), then proceed to failure analysis and its compensation, e.g. by means of setting a target position closer to the mobile robot.

7. If another target position must be set, then go to step 2, otherwise stop.

The proposed optical guidance system has the following features:

- target positions can be set precisely in the environment by means of an optical pointing device (e.g. a laser pointer).

- close-loop control based on visual feedback instead of open-loop control will provide better positioning accuracy of the mobile robot.

- path to follow can be indicated as a sequence of the target positions.

- TCP/IP and wireless Ethernet are used for communication based on a client-server model between the mobile robot and the guidance system.

- accumulation of odometry errors will not influence localization of the mobile robot in the environment because localization is performed when the robot has reached its indicated target position.

- one laser pointing device can indicate target positions for multiple mobile robots within the environment.

3 The Mobile Robot

The mobile robot and its overall control architecture are shown in Fig. 2. The robot is equiped with four omnidirectional wheels which allow it to perform motions in two directions and rotate simultaneously. Three DC motors and a transmission mechanism provide the omnidirectional motion [2]. Three servo-systems execute the motion control commands issued by the control system of the robot and ensure attaining the commanded position, orientation and velocity.

The sensor system of the robot involves a color CCD camera and eight infrared and sonar range sensors. These sensors gather data about the local environment in order to evaluate relative distances between the robot's frame and environmental objects. Processing of the color data about the local environment allows the robot to detect and localize a specified object, e.g. another mobile robot. The sensor system also involves three encoders used in the feedback loop of the servo-systems.

The control system processes the data obtained from the sensor system and the servo-systems and issues the motion control commands which are subsequently executed by the servo-systems. The robot is able to follow a specified object (e.g. a mobile robot) while keeping a safe distance from the object as a function of the motion velocity. The target positions for the mobile robot can also be set by a central computer or a remote human operator from a graphical user interface [3]. The infrared communication system is used for inter-robot communication when the mobile robots are assigned to perform cooperative tasks [4].

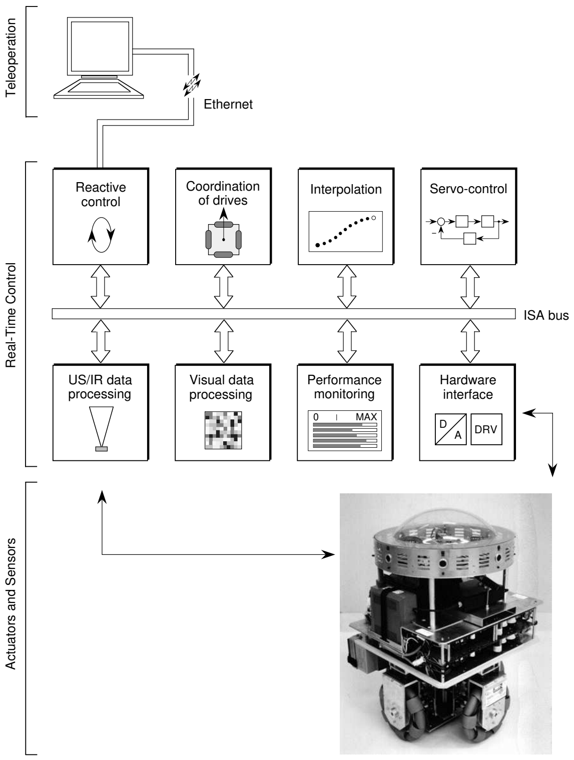

Figure 2: The overall control architecture

The kinematic model of the robot is depicted in Fig. 3. The coordinates of the robot relative to a reference coordinate system $\mathbf{\Sigma}^w$ are denoted as

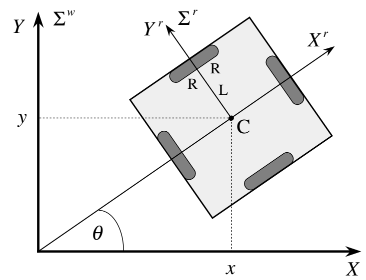

Figure 3: A kinematic model of the mobile robot

$\mathbf{x} = (x, y, \theta)^T$, where $x = x(t)$ and $y = y(t)$ are the coordinates of the intersection point C of the imaginary axes connecting the wheel centers of each pair of the parallel wheels, $\theta = \theta(t)$ is the orientation of the robot, and t is time. Let R denote the wheel radius while L is the distance between the point C and the wheel center. The robot's velocity relative to $\mathbf{\Sigma}^r$ is

$$\dot{\mathbf{x}}^r = \mathbf{J}^r \, \dot{\mathbf{q}}, \tag{3}$$

where $\dot{\mathbf{x}}^r = (\dot{x}^r, \dot{y}^r, \dot{\theta}^r)^T$, $\dot{\mathbf{q}} = (\dot{q}_1, \dot{q}_2, \dot{q}_3)^T$ represents the velocities of the actuators, and $\mathbf{J}^r$ denotes the Jacobi matrix with respect to $\mathbf{\Sigma}^r$:

$$\mathbf{J}^r = \begin{pmatrix} k_1 R & 0 & 0 \\ 0 & k_1 R & 0 \\ 0 & 0 & k_2 \frac{R}{L} \end{pmatrix} \tag{4}$$

where k_1 and k_2 are the gear coefficients. Because $\mathbf{J}^r$ is diagonal, the translation along each direction and rotation with respect to $\mathbf{\Sigma}^r$ are decoupled and can be driven by the corresponding actuators [2].

The robot's velocity relative to $\mathbf{\Sigma}^w$ is

$$\dot{\mathbf{x}} = \mathbf{J}(\theta) \, \dot{\mathbf{q}}, \tag{5}$$

where $\mathbf{J}(\theta)$ is the Jacobi matrix with respect to $\mathbf{\Sigma}^w$:

$$\mathbf{J}(\theta) = \begin{pmatrix} k_1 R \cos\theta & -k_1 R \sin\theta & 0 \\ k_1 R \sin\theta & k_1 R \cos\theta & 0 \\ 0 & 0 & k_2 \frac{R}{L} \end{pmatrix}. \tag{6}$$

The Eq. (5) allows for evaluation of the translational and rotational velocities of the robot and estimation of its location with respect to Σ^w. Eq. (5) is valid for a vehicle moving on flat ground with pure rolling contact without slippage between the wheels and the ground.

The overall control architecture of the mobile robot is shown in Fig. 2. It involves three main parts: the vehicle with its actuators and sensors, the on-board real-time control system, and the remote control interface.

The reactive control ensures an autonomous operation of the robot in a dynamic environment [4, 5]. The collision avoidance algorithm operates with a rule matrix obtained from an adaptive behavior acquisition scheme which is based on reinforcement learning. The reactive control processes data gathered by the ultrasonic and infrared sensors as well as the color CCD camera. Based on the sensors configuration, eight possible directions of motion are considered [5, 6]. The reactive control algorithm ensures a collision-free motion to a given target position.

The motion controller provides coordinated translation and rotation of the omnidirectional vehicle when moving to a target position, as well as to attain a desired velocity of motion. Omnidirectional motion (holonomic case) or constrained motion (nonholonomic case) can be performed according to the assigned task. For instance, vision-based tracking of a dynamic object requires the object to be kept in the view field of the on-board CCD camera and is achieved by the coordinated control of the translational and rotational coordinates of the robot.

The motion controller of the robot is shown in Fig. 4, where: `SPL` denotes spline-interpolation, `SC` - servo-controllers, `D/A` - digital/analog converters, `M` denotes motors and $\mathbf{F}(\theta)$ is an orthogonal matrix of the coordinate transformation from $\mathbf{\Sigma}^r$ to $\mathbf{\Sigma}^w$:

$$\mathbf{x} = \mathbf{F}(\theta)\mathbf{x}^r. \tag{7}$$

The desired position $\mathbf{x}_d = (x_d, y_d, \theta_d)^T$ and motion velocity $\dot{\mathbf{x}}_d = (\dot{x}_d, \dot{y}_d, \dot{\theta}_d)^T$ are set by the operator or evaluated from the sensor data about the environment, e.g. during vision-based tracking of an object. The spline-interpolation is performed to achieve the

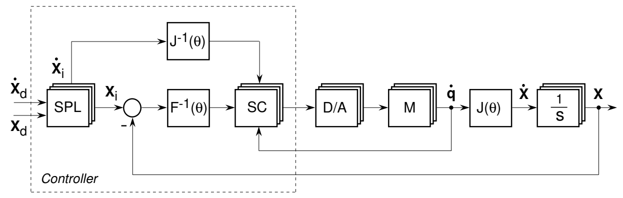

Figure 4: A motion controller of the robot

coordination of the drives and to generate the reference functions to control the position and orientation $\mathbf{x}_i = (x_i, y_i, \theta_i)^T$ and the velocity $\dot{\mathbf{x}}_i = (\dot{x}_i, \dot{y}_i, \dot{\theta}_i)^T$ of the robot. The servo-controllers ensure the accurate execution of the commanded reference functions.

The performance monitoring aims to increase the reliability and safety of the robot operation. The monitoring of measurable signals allows for fault detection and diagnosis in a closed-loop during operation, e.g. measuring the capacity level of the electric batteries of the robot and, if needed, requesting the control system to interrupt execution of the on-going task and direct the robot to a location where the electric batteries can be replaced.

4 Motion Generation

The motion generation approach makes use of a modified cubic spline-interpolation [7, 8]. In this section, the motion generation is discussed for one coordinate of the robot. In the control system, the approach is extended for the three coordinates. Let us consider a cubic polynomial

$$x(t) = \sum_{i=0}^{3} a_i \left(\frac{t}{T}\right)^i, \qquad 0 \le t \le T, \tag{8}$$

where a_i are real coefficients and $T > 0$ is an interpolation period. Let the boundary conditions be

$$\left\{ \begin{array}{ll} x(0) = x_0, & x(T) = x_T, \\ \dot{x}(0) = \dot{x}_0, & \dot{x}(T) = \dot{x}_T \end{array} \right. \tag{9}$$

and $\Delta x_T = x_T - x_0$.

Let us study the case, when for $t \ge 0$ the coefficients a_i are recomputed with a sampling period $T^\star$ such that $T_s < T^\star \ll T$ while $T_s > 0$ and T_s, T, $\dot{x}_T$, Δx_T are constant. Let the appropriations $x_0 := x(kT^\star)$, $\dot{x}_0 := \dot{x}(kT^\star)$, $x_T := x(kT^\star) + \Delta x_T$, $k = 1, 2, \ldots$ ensure the continuity of $x(t)$ and $\dot{x}(t)$ at the boundary between subsequent $(k-1)$ and k computations.

The goal is to obtain a smooth function $x(t)$ by interpolating from $x(kT^\star)$, $k = 0, 1, 2, \ldots$ to a given x_d when

$$|\, x_d - x(kT^\star)\,| \gg |\, \Delta x_T \,|, \tag{10}$$

and ensure, that for a given $\dot{x}_d$ the following condition holds for $n > k$:

$$|\, \dot{x}_d - \dot{x}(nT^\star)\,| < \varepsilon, \tag{11}$$

where $\varepsilon > 0$ is a small constant, as well as

$$|\ddot{x}(t)| \le \ddot{x}_{max}, \tag{12}$$

where $\ddot{x}_{max} > 0$ is a given constant.

The recomputation of the coefficients a_i terms in the proximity of the given x_d:

$$|\, x_d - x(kT^\star)\,| \le |\, \Delta x_s \,|, \tag{13}$$

where $\Delta x_s = \Delta x_s(\dot{x}_d, \ddot{x}_{max})$ and $|\, \Delta x_s \,| > |\, \Delta x_T \,|$.

A name *virtual running point* can be used to indicate that each k-th interpolation step is performed to a virtual point situated at a distance Δx_T and this point is shifted with a sampling period $T^\star$ in the direction of x_d, as it is illustrated by Fig. 5.

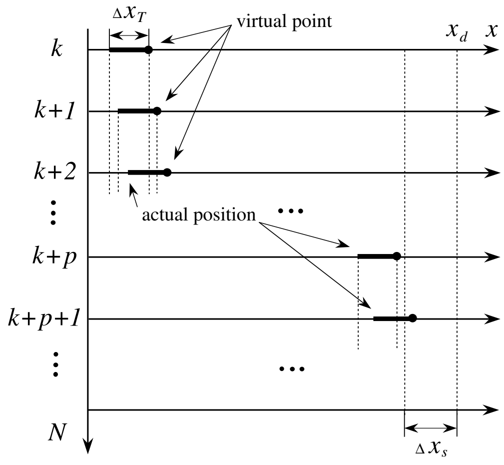

Figure 5: A virtual running point

When the coefficients a_i are recomputed with the sampling period $T^\star$, one can derive from Eq. (8) and Eq. (9) [8]:

$$\lim_{k\to\infty} \dot{x}(kT^\star) = \frac{6\,(1-\tau)\,v - 3\,(2-\tau)\,\dot{x}_T}{4 - 3\,\tau}, \tag{14}$$

where $\tau = \frac{T^\star}{T}$ and $v = \frac{\Delta x_T}{T}$. Because $\dot{x}_T$ is an auxiliary constant in such computations, one can choose $\dot{x}_T = 0$ and rewrite Eq. (14) as

$$\lim_{k\to\infty} \dot{x}(kT^\star) = \frac{6\,(1-\tau)}{4 - 3\,\tau}\, v. \tag{15}$$

Hence, if v is computed as

$$v = \frac{4 - 3\,\tau}{6\,(1 - \tau)}\,\dot{x}_d, \qquad (16)$$

the condition given by Eq. (11) is satisfied according to Eq. (15).

For an instant $t = kT^\star$, $\dot{x}_T = 0$ and recomputation of the coefficients a_i with the sampling period $T^\star$, one can obtain from Eqs. (8), (9), and (16):

$$\ddot{x}(kT^\star) = \frac{(4 - 3\,\tau)\,\dot{x}_d}{6\,(1 - \tau)\,\Delta x_T} \cdot \left(\frac{4 - 3\,\tau}{1 - \tau}\,\dot{x}_d - 4\,\dot{x}(kT^\star)\right). \qquad (17)$$

From Eq. (17) it follows, that a value of $\ddot{x}(t)$ depends on a deviation between $\dot{x}_d$ and $\dot{x}(t)$. Taking into account Eq. (11), one can conclude that this deviation has its extremal value at the moment when a new $\dot{x}_d$ is given, i.e. in the beginning of the interpolation step. According to Eq. (17), for given $\dot{x}_d$ and $\dot{x}(kT^\star)$, there can always be found such $\Delta x_T \neq 0$ that

$$|\ddot{x}(kT^\star)| \leq \ddot{x}_{max}, \qquad (18)$$

where $\ddot{x}_{max} > 0$ is a given constant.

The features of the developed motion generation are:

- $x(t)$ and $\dot{x}(t)$ are continuous.
- $\dot{x}(t)$ tends to a given value $\dot{x}_d$ and the speed of the convergence depends on the value of Δx_T.
- $\ddot{x}(t)$ is limited and tends to zero while $\dot{x}(t)$ tends to $\dot{x}_d$.

These features allow one to apply the method to various robotic tasks where trajectory generation in real time is needed to control the motion of the mobile robots in a dynamic environment (e.g. collision avoidance or tracking an object).

5 Experiments

The developed control system was implemented and tested on the omnidirectional mobile robot shown in Fig. 2. The software was developed in C language. A VxWorks real-time operation system was used. The compiled code is transmitted via a wireless Ethernet to the on-board control system based on a Pentium processor and ISA bus.

The operation of the control system is illustrated by an example shown in Fig. 6 where the initial coordinates of the vehicle are (0.7, 1.1). In this example, the desired velocity is 0.15 m/s, and the default rotational velocity is 0.28 rad/s. First, the robot moves in the direction of an intermediate position (0.2, 2.7). Then, after a period of 5 s used in this example, a new goal position (−1.6, 2.2) is given so that the robot has to change its motion direction. The robot's motion is depicted in Fig. 6. The corresponding velocity profiles are near-trapezoidal, as it is shown in Fig. 7.

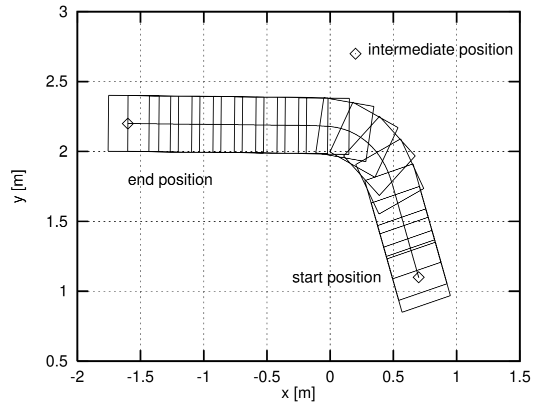

Figure 6: Motion on the (x, y)-plane

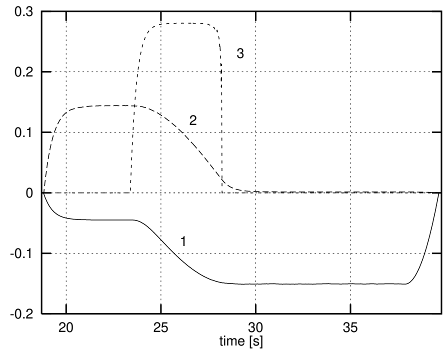

Figure 7: Velocity profiles: 1 - $v_x(t)$ in m/s, 2 - $v_y(t)$ in m/s and 3 - $v_\theta(t)$ in rad/s

The control system was tested on two visually-coupled omnidirectional mobile robots. Our experimental setup is shown in Fig. 8. One robot is assigned as the leader and the other as the follower. The leader has a spherical red cover that serves as a reference object for the image processing performed by the follower. The follower is equiped with a CCD camera mounted on the top of the robot. The acquired color image is quantified with a coarse grid to allow for on-line computation. The segmentation of the reference object from the background is performed by means of thresholding the RGB colors in the image [9, 10]. The distance to the leader is computed from the pixel width of the reference object. The lateral position of the leader relative to the follower is obtained from the horizontal pixel deviation of the center of the reference object relative to the center of the image.

A remote human operator uses a graphical user interface and sets the goal positions and the motion velocity of the leader within an environmental map shown on the display as it is illustrated in Fig. 9. During motion, the leader's position is depicted on the map and the operator can view the visual data gathered by the on-board CCD camera of the robot about the local environment. The follower is visually-coupled to the leader and moves as a visually-connected trailer. Our video illustrates the experiments performed. The visual tracking of the mobile robots will be extended to tracking a laser light projected onto the floor surface.

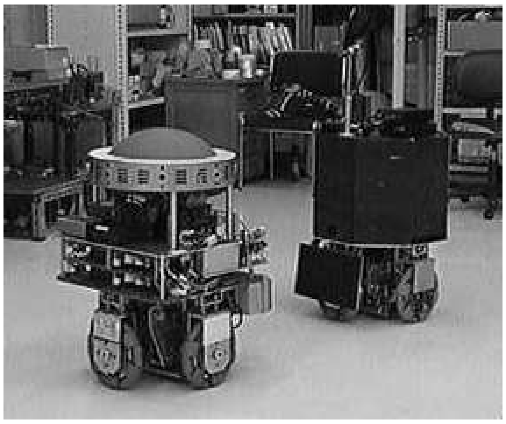

Figure 8: Visually-coupled mobile robots: the leader (left) and the follower (right)

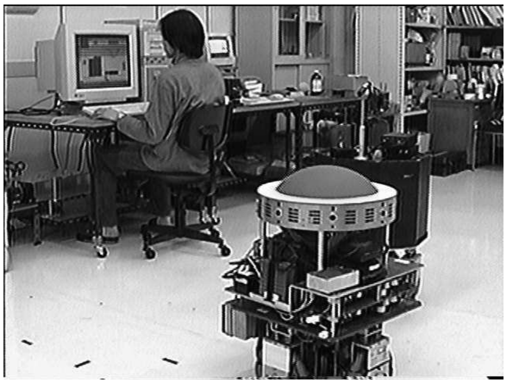

Figure 9: Remote control by an operator

6 Conclusion

The concept of an optical guidance system that makes use of a laser pointing device was introduced. The control architecture and motion generation approach for the vision-based guidance of the omnidirectional mobile robot were considered. The experimental results on the remote control and the visually-coupled motion of the two mobile robots have shown effective operation of the control system. The visual tracking of the mobile robots will be extended to tracking a laser light projected onto a floor surface.

References

[1] J. Borenstein, H. R. Everett, L. Feng, *Navigating Mobile Robots – Systems and Techniques*, A K Peters, Wellesley, MA, USA, 1996, 225 p.

[2] H. Asama *et al.*, Development of an Omni-Directional Mobile Robot with 3 DOF Decoupling Drive Mechanism, *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Nagoya, Japan, May 21-27, 1995, pp. 1925-1930.

[3] T. Suzuki *et al.* A Multi-Robot Teleoperation System Utilizing the Internet, *Advanced Robotics*, Vol. 11, No. 8, 1998, pp. 781-797.

[4] Y. Arai *et al.*, Collision Avoidance among Multiple Autonomous Mobile Robots using LOCISS (LOcally Communicable Infrared Sensory System), *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Minnesota, MN, USA, April 22-28, 1996, pp. 2091-2096.

[5] Y. Arai *et al.*, Adaptive Behaviour Acquisition of Collision Avoidance among Multiple Autonomous Mobile Robots, *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Grenoble, France, Sept. 7-11, 1997, pp. 1762-1767.

[6] I. E. Paromtchik, U. M. Nassal, Reactive Motion Control for an Omnidirectional Mobile Robot, *Proc. of the Third European Control Conference*, Roma, Italy, Sept. 5-8, 1995, pp. 3074-3079.

[7] H. Akima, A New Method of Interpolation and Smooth Curve Fitting Based on Local Procedures, *Journal of Assc. for Comp. Mach.*, Vol. 17, No. 4, 1970, pp. 589-602.

[8] I. E. Paromtchik, H. Asama, A Motion Generation Approach for an Omnidirectional Vehicle, *Proc. of the IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA, USA, April 24-28, 2000, pp. 1213-1218.

[9] J. R. Parker, *Algorithms for Image Processing and Computer Vision*, John Wiley, 1997.

[10] J. D. Crisman, Color Region Tracking for Vehicle Guidance, *Active Vision*, MIT Press, Cambridge, MA, 1991, pp. 107-220.