Basic Design of Visual Saliency Based Autopilot System Used for Omnidirectional Mobile Electric Wheelchair

Hirokazu Madokoro*, Keigo Shirai, Kazuhito Sato, Nobuhiro Shimoi

Faculty of Systems Science and Technology, Akita Prefectural University
84-4, Tsuchiya Aza Ebinokuchi, Yurihonjo City, Akita, 015-0055, Japan

Copyright ©2015 by authors, all rights reserved. Authors agree that this article remains permanently open access under the terms of the Creative Commons Attribution License 4.0 International License

Abstract This paper presents a fundamental design of an autopilot system to actualize automatic locomotion for an electric wheelchair with emphasis on simplicity and functionality. For this study, we designed a novel electric wheelchair with advanced mobility using Mecanum wheels that actualize omnidirectional movements without turning, and developed a prototype with consideration devoted to the exterior design. Moreover, our developed prototype is considered with devotion of the exterior design. Our design concept is Electric Personal Assistive Mobility Device (EPAMD), which has ease of integration with a person's daily life. To prevent collisions, ranging sensors and depth sensors are used for environmental recognition. This paper presents global locomotion and local locomotion as frameworks for an autopilot. For global locomotion, we address algorithms of visual landmark detection based on visual saliency and the creation of category maps based on adaptive and unsupervised machine learning. Our method visualizes time-series features and their relations on visual landmarks in a low-dimensional space. We examine a novel design and its possible adaptation to an electric wheelchair as an EPAMD, especially intended to improve the independence of elderly people in their daily life.

Keywords Electric wheelchair, autopilot, ranging sensor, Mecanum wheel, Electric Personal Assistive Mobility Device (EPAMD)

motors, are used by elderly people because they require only minimum physical effort for movement. Existing electric wheelchairs require a change of the vehicle body to a direction controlled by a user. Subsequently, the wheelchair changes its direction to release a passenger easily. However, it is difficult to move in crosswise directions without turning, particularly in narrow areas that are smaller than the diagonal dimension of its body size.

Actually, joysticks are used commonly as control devices for electric wheelchairs. However, their use requires training to gain sufficient skill of operation, especially to move in narrow or crowded areas. Accidents caused by operational errors occur especially for elderly people, not only because of decreased perceptual capability with aging, but also because of unfamiliar control systems. According to a report issued by the Tokyo Metropolitan Police Agency in Japan, up to 200 accidents involving cars and electric wheelchairs occur each year [2]. Of those, 25 % were attributable to operational mistakes [3]. However, electric wheelchairs are not included among the data because the users are categorized as pedestrians according to the Road Traffic Law in Japan.

The numbers of electric-wheelchair-using inpatients and outpatients have increased year by year, as have accidents and problems in hospitals. These tendencies have been confirmed by numerous comments from medical doctors and nurses. Particularly, crashes and runaway incidents attributable to unfamiliar control of wheelchairs occur often among elderly people, especially in places such as elevators, narrow aisles, and bed-side areas. To move safely, an autopilot system for electric wheelchairs should be used [4] because of the greatly restricted control when using a joystick.

This study was undertaken to provide basic consideration of an autopilot system for an omnidirectional autonomous mobile electric wheelchair. Fig. 1 depicts our electric wheelchair prototype [5]. For the actualization of an autopilot, we intend to prevent accidents by reducing the opportunity for a user's erroneous control. For actualizing omnidirectional movements, we

1 Introduction

Low birthrate and longevity are progressing rapidly, especially in economically developed countries [1]. For this society, the demand for new systems is increasing not only for supporting the physical capabilities of elderly people, but also for reducing the burdens of caregivers. Electric wheelchairs, which are driven by electric

Figure 1. Exterior of our electric wheelchair prototype.

used Mecanum wheels, which are driven independently using four motors controlled with a microcomputer. Our design principles emphasized cost and system simplicity, using a monocular depth camera and ranging sensors to measure the distance to obstacles.

2 Related Studies

In the field of robotics, various mechanisms have been developed to achieve omnidirectional autonomous locomotion [6, 7]. Using these mechanisms, several methods have been applied to electric wheelchairs [8, 9, 10, 11]. Wada et al. developed a parallel mobile electric wheelchair using a rotating chair and omnidirectional wheels for the front tires [12]. For their method to actualize parallel motion, they turn the vehicle body to 45 deg after 45 deg rotation of the chair. However, the diagonal width of the wheelbase must be used when the chair body turns 45 deg. Therefore, it is impossible to move in a confined space, as in the case of parallel movements in an elevator. The mechanism and programs are complicated for the holonomical control of the both the drive wheels and the turning of the chair. This complex mechanism engenders cost problems for its commercialization and practical use.

Kitagawa et al. actualized omnidirectional movements using four omni-wheels [13]. However, the travelling performance is low because of the omni-wheels, which are spherical. Moreover, the efficiency of the use of electric motors drops dramatically for the basic movement forward, which is high efficiency of use because of two-wheel drive, except of movements related to turning. For the quality of riding, to limit the frequency of discomfort, they removed characteristic vibrations of the system using a hybrid shaping method proposed by Yano et al. [14].

For this study, we specifically examined Mecanum

wheels for use with an electric wheelchair. Electric wheelchairs using Mecanum wheels are available commercially, such as MoonWalk by Mec Design Corp. in Japan and FJ-UEC-600 by Fujian Fortune Jet Mechanical & Electrical Technology Corp. in China. Both manufacturers use four motors to control each wheel using a joystick. They have no consideration for an intelligent system. They merely use a relay circuit for a motor controller as manual operation. For example, MoonWalk requires switching of a relay circuit to move both sides. From the perspective of handling and user interface, intelligent approaches require a robotic electric wheelchair [4].

For autonomous locomotion and automatic driving of electric wheelchairs, several prototypes have been released, such as a Wheelchair Robot by Shimizu Corp., an Autonomous Wheelchair by the National Institute of Advanced Industrial Science and Technology, and the Intelligent Wheelchair Robot by Fujitsu Co. Ltd. In addition, the Robotics Wheelchair designed by MIT has been studied as a large-scale project [15]. For these studies, omnidirectional stereo cameras and laser range finders (LRFs) were used for sensing of a wide range for collision avoidance of objects and people. Simultaneous Localization and Mapping (SLAM) is the mainstream approach used to estimate the position while creating an environmental map together.

The approach using SLAM is the most realistic solution for an electric wheelchair as a boarding-type autonomous mobile robot [16]. For creating and updating a map in real time, the processing load is high using SLAM, although FastSLAM was proposed by Montemerlo et al. [24] to reduce computational costs. Moreover, SLAM requires high-precision and wide-range sensors. Suzuki et al. developed a robot wheelchair prototype to follow a person such as a helper or a caregiver [17]. However, they used an omnidirectional camera and three LRFs in the sensing system. Because of the rapid progress of longevity and the aging of society, cost reduction using simple sensors is an important task that must be accomplished to popularize the autopilot of electric wheelchairs with safe movements. Furthermore, it requires construction of a system using a low-power computer while reducing the computational costs because electric wheelchairs are driven by a battery.

As a robot navigation system based on visual saliency, Chang et al. proposed a method that combined Saliency Maps (SM) with Gist, a global scene descriptor [18]. They actualized indoor and outdoor navigation using a monocular camera mounted on a small mobile robot called Beobot2.0. However, their method continuously detected landmarks based on saliency through simple template matching that is weighted in advance. Ho et al. proposed a navigation framework to combine SLAM and visual landmarks detected using a monocular camera [19]. Nevertheless, the evaluation experiment was done in an environment that included narrow crossroads or a T-junction at a corridor, although they proposed a method of fusing with SLAM. The evaluation of real environments was insufficient.

Figure 2. Our concept of an autopilot system used for an electric wheelchair in an indoor environment.

Figure 3. Design of our electric wheelchair prototype: (a) front view and (b) rear view.

3 Electric Wheelchair Prototype

In the current market, electric wheelchairs are expensive, with prices roughly equal to those of motorbikes. To support their wider use, cost reduction is important. Nevertheless, the cost must be increased somewhat to equip sensors and its control system for an autopilot. The aim of this study is to actualize such a system using reasonable sensors such as ultrasonic sensors and a monocular camera. Generally, an autopilot is used for ships or airplanes, but this autopilot system is used for the control of an electric wheelchair. Fig. 2 depicts our concept model of an autopilot system used for an electric wheelchair.

3.1 Glimmer

Our design concept is based on Electric Personal Assistive Mobility Device (EPAMD), which provides high quality of comfortable riding and ease of integration with

Figure 4. Interior of our electric wheelchair prototype.

a person's daily life. With consideration of this concept, we designed a novel electric wheelchair as shown in Fig. 3 using Three-Dimensional Computer Aided Design (3D CAD) software. Fig. 1 depicts our developed omnidirectional electric wheelchair prototype based on this 3D design. We named it Glimmer from the phrase "a glimmer of hope." We therefore designated this prototype as Glimmer.

The body of the prototype was built using aluminum pipes to ensure its strength. Cypress boards were used for the seat, backrest, and exterior of the body. We took into consideration the soft and comfortable characteristics for the exterior using wood. With consideration of the overall design, we installed sensors where they could not be viewed by a user. Furthermore, light emitting diode (LED) tapes were installed for illumination to improve the design and to ensure safety for surrounding people.

Fig. 4 depicts the inside of the vehicle body. The two batteries of the yellow top occupy the vast majority of the space. These are deep-cycle lead batteries (Optima; Johnson Controls Inc.) We selected them with consideration of their cost and safety, although lithium-ion polymer batteries might produce a smaller and lighter system overall. Herein, there is no consideration of a folding mechanism resembling that of a manual wheelchair.

3.2 Interface

As a constraint of Mecanum wheels, it is hard for Glimmer to move on a rough road or steps. Therefore, we assume that the outdoor usage is unsupported. The main usage is only indoor areas such as hospitals, nursing-care facilities, and senior residences.

To prevent accidents caused by erroneous control movements, we assigned weights for movements using the automatic mode. We prepared the manual mode to switch for a situation that it is difficult to move for the automatic mode. A touch panel is used for the interface. Fig. 5(a) depicts a screenshot for the main control panel showing three functions: automatic mode for the autopilot, manual mode, and emergency stop.

For the automatic mode, the selected destination is displayed on the touch panel. After selecting this mode, Glimmer estimates its own position to detect visual landmarks from the background and objects present in a moving environment. Furthermore, Glimmer gener-

Figure 5. Touch panel interface for autopilot: (a) main control panel, (b) manual operation mode I, and (c) manual operation mode II.

ates a path toward the destination automatically from the current position. The current position in the path is displayed on the touch panel in real time. Persons and obstacles are present without the existing map with actual paths. For collision prevention, we used depth sensors and ranging sensors.

There are two manual operation modes using a tablet computer. Fig. 5(b) depicts operation mode I to control the directions of movement with the icon on the right side and to control the speed with the bar on the left side. Fig. 5(c) depicts operation mode II to use the tablet as a 3D handle. For this operation, the touch-panel display is not used. Glimmer is controlled by a user with a gyro sensor and an acceleration sensor in the tablet. Thereafter, we describe details of operation mode II.

Images obtained using the camera are displayed on the display of the tablet in real time. For the front direction of the image as the reference surface, orthogonal coordinates of X, Y, and Z are defined respectively as the roll, yaw, and pitch angles. These angles respectively represent the rotational angles of vertical, depth, and horizontal of the tablet. The gyro sensor detects the respective rotations of three axes. Glimmer moves forward if the tablet is leaned to the front side. Similarly, Glimmer moves back, left, and right directions if the tablet is leaned to respective directions. Moreover, Glimmer turns to the clockwise and counterclockwise directions if the tablet is turned to the respective directions. We set limitations of acceleration and deceleration only for the movements of forward and backward. The degrees of acceleration and deceleration are linearly proportional to the degree of the gradient. Glimmer stops to maintain the tablet horizontally. Emergency stop is executed by the trigger of pulling the tablet using an acceleration sensor.

Glimmer stops immediately to maintain safety if the emergency stop button is selected. For the manual oper-

Figure 6. Exterior of Mecanum wheels (TDAM-0083).

Table 1. Specifications of TDAM-0083.

Material of frame	Aluminum
Material of barrel tire	SBR rubber
Load capacity (one wheel)	40.0 kg
Thickness	65 mm
Outside diameter	203.0 mm
Inner diameter	Φ 28.6 mm
Weight (one wheel)	1.15 kg

ation mode, Glimmer maintains the waiting status unless the user selects it again. For the automatic operation mode, Glimmer returns to autopilot operation if the user selects it again.

3.3 Mecanum Wheel

Omnidirectional platforms have numerous advantages for autonomous mobile robots [21]. We used Mecanum wheels for the mobile mechanism of Glimmer. Fig. 6 portrays the exterior of a Mecanum wheel (TDAM-0083; AndyMark, Inc.). The major specifications are shown in Table 1. In each aluminum wheel, 12 barrel tires are installed with 45 deg around the axle. The barrel tires are made of styrene butadiene rubber (SBR).

The Mecanum wheel movement resembles standard wheels for transmission from motors. However, the barrel tires over the wheels move in the direction of 45 deg with free rotation. The omnidirectional autonomous locomotion is actualized with sliding tires and the combination of the rotational directions of each wheel that drives independently. Using Mecanum wheels, the steering mechanism is obviated.

3.4 Motor Control

For driving motors, we used 24 V direct current (DC) motors (BLH450K-15; Oriental Motor Co. Ltd.). We used four motors and drivers controlled by a microcomputer. There are 10 channels for the control signal input and output ports. START/STOP, RUN/BRAKE, and CW/CCW pins are assigned to the input ports. The control of the ON/OFF pin is controlled by the register to valid input. MTU and PWM are used for speed control. The rotational direction is switched by the CW/CCW pin. The STOP/BRAKE pin is used, respectively, for a normal stop and an emergency stop. The SPEED pin outputs 30 pulses per rotation from the motor. We used this pulse signal to correct the distance of movements after calculating the number of rotations of the motor.

Figure 7. Control patterns of wheels for movement: (a) forward, (b) right-forward, (c) right, and (d) clockwise turning.

Using the pulse-wave width τ and the cycle T , D is represented as

$$D = \frac{\tau}{T}. \quad (1)$$

The rotational speed becomes fast or low if the applied voltage is high or low. However, it is impossible to control the applied voltage directly from the microcomputer as continuous values. The signal output of a microcomputer is merely a binary value: high or low. Therefore, the rotational speed can be controlled to change the pulse width in a fixed interval using PWM.

For omnidirectional movements without turning, it is necessary to control the respective wheels of the rotational direction and their speeds independently. We implemented 12 functions: forward, back, right, left, right-forward, left-forward, right-back, and left-back for basic motions; clockwise turning and counter-clockwise turning; and normal stop after deceleration and emergency stop. These functions are combined to actualize the autonomous locomotion. Herein, we present four motions in detail: right-forward, right, clockwise turning, and emergency stop, as shown in Fig. 7.

The positions of respective motors are described as Front Left (FL), Front Right (FR), Rear Left (RL), and Rear Right (RR). Fig. 7(a) depicts the forward to rotate FL and RL for counter-clockwise and FR and RR for clockwise. The driving force to the horizontal side is neglected between FL/RL and FR/RR. Fig. 7(b) depicts the right-forward motion. The two motors of FL and RR are rotated. The other two motors are slipped without torque. Fig. 7(c) depicts movement to the right. The FL and FR wheels rotate clockwise. The driving force to the vertical side is neglected between FL/RL and FR/RR. Fig. 7(d) depicts the turning movement. Glimmer turns clockwise by rotating all wheels counter-clockwise.

3.5 Output Torque

Let m be the Glimmer's weight. The total weight including a passenger is 120 kg. We set the rolling resistance coefficient μ to 0.021, which is multiplied by the safety rate. This rate is 40% higher than the rolling resistance coefficient between a concrete road and the

Figure 8. Sensors used for this study: (a) ranging sensor and (b) depth sensor.

Figure 9. Distance and output voltage of the ranging sensor from the datasheet.

normal tire of a bicycle. The wheel radius is 101.5 mm. Therefore, the load torque T_L is 2.509 N·m. Let ω represent the angular acceleration. T_a is defined as

$$T_a = J \frac{d\omega}{dt}. \quad (2)$$

The moment of inertia J is defined as

$$J = \frac{1}{2}mr^2. \quad (3)$$

Therefore, the acceleration torque T_a is 1.72 Nm if the time of acceleration is 1.0 s. The required torque T for movement of Glimmer is the summation of T_L and T_a as

$$T = \frac{T_L + T_a}{2}, \quad (4)$$

where the denominator is a coefficient corresponding to the minimum number of motors. T is 2.11 Nm that is valid for the allowable torque of BLH450K-15.

4 Sensing System

We used ranging sensors and depth sensors for environmental measurements. This section presents a description of details related to each sensor and their respective assignments.

4.1 Ranging Sensor

Glimmer uses a ranging sensor (Sharp Corp.) as shown in Fig. 8(a). The range of this sensor is 100–800 mm on the datasheet. Comparison of LRF, this sensor obtains merely 1D point information. We use this sensor for the detection of obstacles, collision avoidance, wall following, and approaching a bed side.

Figure 10. Sample images obtained using Xtion: (a) RGB image and (b) depth image.

This sensor has three ports for the power, ground, and sensor. The power port is supplied to 4.5–5.5 V. We shared the ground port to that of the microcomputer. The sensor port outputs voltages concomitantly with distances. For calculating the distance from the voltage, we connected this port to an input port including an A/D converter in the microcomputer.

Fig. 9(a) depicts the relation between the distance L and the output voltage V_o from the datasheet. The output voltage V_{10bit} is converted from V_o using a 10-bit A/D converter on a microcomputer. Let V_{near} be an approximate output voltage. The supply power is 5.0 V, so V_{10bit} gives the following.

$$V_{10bit} = \frac{1024}{5} V_{near} \quad (5)$$

The measured distance L' , which is depicted in Fig. 9(b), is calculated as the following.

$$L' = \frac{6787}{V_{10bit} - 3} - 4 \quad (6)$$

Actually, V_o increases from 0 mm, and peaks out at 3.1 voltage near 70 mm. Subsequently, V_o decreases nonlinearly over 70 mm. It was flat over 800 mm. For this property, we considered that the valid range of this sensor is 100–800 mm.

The internal circuit of the ranging sensor requires 38.3 ± 9.6 ms for charging after supplying the electric power. Subsequently, an interval for 5 ms is required as the maximum to update the output value. Therefore, the waiting time is set to 53 ms at the beginning of A/D conversion. Moreover, the updating output value is set to 5 ms intervals after using A/D converters. As a mechanism to engender this interval, we used a compare match timer (CMT) as a timer of a cyclic interrupt.

4.2 Depth Sensor

Glimmer used a depth sensor (Xtion PRO; ASUS Corp.) as shown in Fig. 8(b). This sensor is a compatible model to a widely used controller (Kinect; Microsoft Corp.). The measurement range is 0.8–3.5 m on

the datasheet. Glimmer uses this sensor for sensing objects that located a middle-range distance. In comparison with a ranging sensor, 3D information is obtained using this sensor. Glimmer uses this sensor not only for the detection of pedestrians, obstacles, and visual landmarks, but also for measurement of the whole structure of the environment to extract valid moving areas.

The ranging sensor measures a straight-line distance to a target. In contrast, the depth sensor measures a distance from the surface of the sensor to a target. This is a sine distance from the sensor to a target, not a straight-line distance as the minimum distance. Therefore, the actual distance is short compared with its apparent distance, especially for terminal regions of an image.

The Xtion body is 35 mm high, 180 mm wide, and 50 mm deep. It includes an infrared projector, an infrared camera, and an RGB camera. The projector shows infrared random dot patterns. The image resolutions of the infrared camera and the RGB camera are, respectively, 320×240 pixels and 640×480 pixels. The frame rates of both cameras are 30 fps.

We used OpenNI as the driver software for Xtion. The representation range of OpenNI is 0.5–10 m. However, the infrared camera is fundamentally restricted to the recognition range of 0.8–3.5 m because of the limitation of its hardware performance. The vertical and horizontal view angles are, respectively, 29 deg and 45 deg. Regarding the view angles, the valid view ranges are 0.6–2.6 m width and 0.8–3.5 m height.

Fig. 10 depicts an image obtained using Xtion. An RGB image and a depth image are obtained simultaneously. Respective distances in the image are represented to the brightness in each pixel. The brightness from high to low corresponds to the distance from near to far. The regions of shadow parts and the outside of the range are shown as low brightness.

4.3 Sensor Assignment

Fig. 11 depicts two examples of the assignment of the both sensors. We installed 12 ranging sensors in each corner of the vehicle body. The installation was done at the upper part of the wheels. Fig. 11(a) depicts an example of depth sensor installation to the top of the chair backrest. This general approach is useful for existing studies conducted with a passenger. For this approach, sensors can measure the environment wide range from a high position. Sometimes, a pole is used for an extension to install sensors higher than this approach, although that design entails many sacrifices. We are aiming to provide a novel design for invisible sensor systems not only for a user, but also for surrounding people. Fig. 11(b) depicts our current approach. Two depth sensors are installed in each front pillar.

5 Autonomous Locomotion

Humans can obtain visual information for the view range even if no precision map for the environment is available. In the human brain, memory can be used to navigate in the actual world. This memory mechanism is referred from World Image (WI) [35] as a conceptual model. A model is created for position estimation and

Figure 11. Examples of assignment of depth sensors and estimated sensing range. Two depth sensors are installed (a) on the top part of the backrest or (b) in front of pillars.

recognition used for computers and robots, much as humans might, if we can actualize WI as an engineering model.

5.1 Global Locomotion

SALM-based approaches necessitate a high processing cost with a complex system implementation for matching between the existing map and a created map obtained using high-dimensional distance sensors such as LRF. For this study, we assess a novel navigation system based on visual landmarks without using SLAM. Glimmer detects visual landmarks using features of visual saliency, as shown in Fig. 12. As a novel approach for an autopilot, Glimmer uses visual landmarks for both localization and navigation.

The destination is selected by a user through a touch panel interface on a tablet computer as an operational device. For this interaction, Glimmer begins to run automatically. Glimmer searches for the shortest path from the current location to the destination. We use Potential Field Methods (PFMs) [36] for path planning and obstacle avoidance. For the path planning of PFMs, pop and push forces are presented, respectively, from the destination and obstacles. This relation is defined as potential functions.

Generally, potential functions are solved using the steepest descent method. According to potential functions, Glimmer moves to the destination while avoiding obstacles. However, it is a challenging task to set a suitable parameter to control the degree of pop force because of its tradeoff relation [37]. Therefore, we consider an approach using Multi-Objective Genetic Algorithms (MOGA) [38] to resolve this tradeoff problem.

Figure 13. Example of visual landmark detection based on visual saliency: (a) original image, (b) SM, (c) return inhibition, and (d) saliency regions as visual landmarks.

We specifically examine visual saliency for the actualization of global locomotion to use objects and partial regions of the background in the scene as visual landmarks without using existing landmarks that are installed in advance. Actually, SMs were used by Itti et al. [22] to detect visual landmarks from a scene image. Humans process vast amounts of information obtained from sensory organs, especially from vision. However, humans do not use all of that obtained information because they have a system that unconsciously assigns attention to remarkable objects.

Based on physiological knowledge, Itti et al. [22] created a model of the process from the retina to the primary visual cortex via the lateral geniculate for SMs as visual search models. The SMs extract visual features that differ from the surroundings, as high saliency regions using a bottom-up approach. Using SMs that obtain saliency spatial distribution in our view range, we can extract necessary information in real time from vast amounts of visual information in the world. We meet a problem to repeat gazing movements at particular regions when we extract several high-saliency regions from there sequentially. To avoid this problem, SMs equip a mechanism of return inhibition to prevent gazing at a certain subject repeatedly. For this mechanism, previously detected points and neighboring regions are left out of consideration as target candidate regions for a while.

Standard SMs are used for processing for a single image. Our processing target is time-series images. Our method actualizes steady detection of visual landmarks without using return inhibition among image frames. Fig. 13 depicts an image example of detection of visual landmarks. Return inhibition is executed after normalization using the local maximum method.

5.2 Visual Saliency

Fig. 14 portrays the procedure of SMs, comprising six steps. The first step is to apply Gaussian filters to create pyramid images from an original image. The second

Figure 12. Our conceptual design to detect visual landmarks based on visual saliency.

Figure 14. Whole procedure of SMs.

step is to create intensity, color, and orientation channel images. The third step is to create feature maps (FMs) that represent the visual features of respective components using Center-Surround images. The fourth step is to create conspicuity maps (CMs) from a linear summation of FMs. The fifth step is to create SMs from linear summation of CMs. The final step is to detect the highest saliency points using Winner-Take-All (WTA) competition. We describe the details related to each step as the following.

5.2.1 Gaussian Pyramids

For the creation of nine pyramid images, the scale is changed from $1/2$ to $1/256$ steps by $1/2$. Subsequently, Gaussian filters are applied to these pyramid images that are obtained using Gaussian filters, which work as low-pass filters that cut a high-frequency band because of

amplification of a low-frequency band. Low-frequency bands are emphasized with the width of the Fourier transform according to the expanding filter size. The effect of blurring is apparent by Gaussian filters. For superimposing all images, Gaussian pyramid images are created.

Intensity, color, and orientation channels are extracted from the Gaussian pyramid images. Let I be an intensity channel defined as

$$I = \frac{r + g + b}{3}, \quad (7)$$

where r , g , and b respectively represent red, green, and blue channels. The hue channel comprises RGB and the yellow channel Y , which is calculated as

$$Y = \frac{r + g}{2} - \frac{|r - g|}{2} - b. \quad (8)$$

Orientation channels are created on the edge of four directions: $\theta = 0, 45, 90$, and 135 deg. Gabor filter G is defined as the product of the sine wave and the Gaussian function [39]. $G(x, y)$ is defined as shown below.

$$G(x, y) = \exp\left\{-\frac{1}{2}\left(\frac{R_x^2}{\sigma_x^2} + \frac{R_y^2}{\sigma_y^2}\right)\right\} \exp\left(i\frac{2\pi R_x}{\lambda}\right) \quad (9)$$

$$R_x = x \cos \theta + y \sin \theta \quad (10)$$

$$R_y = -x \sin \theta + y \cos \theta \quad (11)$$

Therein, λ , θ , σ_x , and σ_y respectively represent the wavelength of the cosine component, the direction component of the Gabor function, the filter size in the vertical axis direction, and the filter size in the horizontal axis direction. The integral value to the vertical side is the maximum if G is applied to lines with gradients in the image. We extract the gradient and frequency components using this property. We defined the filter size as $M \times N$ pixels. The filter output $Z(x, y)$ at the sample point $P(x, y)$ is defined as

$$Z(x, y) = \sum_{j=-\frac{N}{2}}^{\frac{N}{2}} \sum_{i=-\frac{M}{2}}^{\frac{M}{2}} G(x+i, y+j) \cdot P(x+i, y+j). \quad (12)$$

The formula above includes the complex term. The final output Z is defined as

$$Z = \sqrt{Rm^2 + Im^2}. \quad (13)$$

5.2.2 Feature Maps

The attention positions are identified by superimposing the differences among different scale pairs obtained using Gaussian pyramid images. These are designated as center-surround operations, which are represented by the operator $\ominus$. For the difference operation, small images are extended to large images. When defining scales as $c, s(c < s)$, a larger scale is represented as $c=2,3,4$; a smaller one is represented as $s = \{c + \delta | \delta \in 3, 4\}$. For the intensity component, the difference $I(c, s)$ is calculated as shown below.

$$I(c, s) = |I(c) \ominus I(s)| \quad (14)$$

Let $RG(c, s)$ and $BY(c, s)$ respectively represent the differences between the red and green component and the blue and yellow component.

$$RG(c, s) = |(R(c) - G(c)) \ominus (G(s) - R(s))| \quad (15)$$

$$BY(c, s) = |(B(c) - Y(c)) \ominus (Y(s) - B(s))| \quad (16)$$

Orientation features are obtained from the difference in each direction.

$$O(c, s, \theta) = |O(c, \theta) \ominus O(s, \theta)| \quad (17)$$

Finally, 42 FMs are obtained from 6 intensity channels, 12 color channels, and 24 orientation channels.

5.2.3 Saliency Maps

We normalize each FM according to the following procedures.

1. The maximum value M on the map is searched.
2. Normalizing all pixels for the fixed range $[0, \dots, M]$.
3. The mean value m is calculated using all local maximum values that are extracted without M .
4. All values are multiplied by $(M - m)^2$.

After normalizing, FMs are superimposed in each channel. Herein, small maps are zoomed for summation in each pixel.

Let N be a normalization function. Linear summations of intensity channel $\bar{I}$, color channel $\bar{C}$, and orientation channel $\bar{O}$ are defined as the following.

$$\bar{I} = \bigoplus_{c=2}^4 \bigoplus_{s=c+3}^{c+3} N(I(c, s)) \quad (18)$$

$$\bar{C} = \bigoplus_{c=2}^4 \bigoplus_{s=4}^{c+3} [N(RG(c, s)) + N(BY(c, s))] \quad (19)$$

$$\bar{O} = \sum_{\theta} N \left(\bigoplus_{c=2}^4 \bigoplus_{s=c+3}^{c+3} N(O(c, s, \theta)) \right) \quad (20)$$

The obtained maps are referred for CMs. Normalizing respective channels of FMs and linear summation, SMs are obtained as

$$S = \frac{N(\bar{I}) + N(\bar{C}) + N(\bar{O})}{3}. \quad (21)$$

Finally, high saliency regions are extracted using WTA.

5.3 Category Map

Our method visualizes time-series features and their relations on visual landmarks to a low-dimensional space. For this visualization, we use a category map [27] to categorize similar features for improvement of robustness against occlusion and corruption.

The initial category map is created in the first running. For multiple runs in the same environment, category maps are updated while maintaining a balance between stability and plasticity together. This balance is maintained adaptively using a mechanism to control the range of updating units. We use U-Matrix by Ultsch et al. [34] to visualize category boundaries to measure the similarity of weights between neighborhood units. Our method actualizes relearning without using previous datasets. For relearning, we use weights that are compressed learning results. The following particularly describes our proposed method to create a category map [33].

Fig. 15 portrays the network architecture of Adaptive Category Mapping Networks (ACMNs) as a learning method for visualizing time-series features on a category map. Actually, ACMNs comprise three modules: a codebook module for vector quantization of input data, a labeling module for creating labels as candidates of categories, and a mapping module for visualizing spatial relations of categories on a category map. These modules comprise Self-Organizing Maps (SOMs) [25], Adaptive Resonance Theory (ART) networks [26], and Counter Propagation Networks (CPNs) [27]. Herein, SOMs and ART are unsupervised neural networks; CPNs are supervised neural networks. In fact, ACMNs actualize both learning modes for an original mechanism to create labels as candidates of categories. The following presents detailed explanations of the respective algorithms after presenting the overview in each module.

Input data are presented directly to the codebook module. This module is used if dimensions of input features differ among datasets. For example, dimensions are various according to the number of feature points on Scale Invariant Feature Transform (SIFT) [28], which is used widely in generic visual object recognition as a part-based local feature. Using this module, input features are quantified to a specific dimension to represent the distributions of histograms. Moreover, this module conducts vector quantization if the dimensions of input features are high. For this process, the data topology is preserved while changing to a low-dimensional space. Herein, this module is not mandatory for use. This module can be passed if the dimensions of input features are fixed for all datasets. We use this mechanism to reduce the load that is attributable to learning for creating codebooks and updating them incrementally.

$\alpha(0) < 1.0$) is the initial value of $\alpha(t)$. $\alpha(t)$ is defined at time t as

$$\alpha(t) = \alpha(0) \cdot \left(1 - \frac{t}{O}\right). \quad (25)$$

In the initial stage, the learning speed is higher when this rate is high. In the final stage, the learning converges while the range decreases.

For this module, the input features of I dimension are quantized into the J dimension, which is a similar dimension to the number of units on the mapping layer. The module output $y_j(t)$ is calculated as

$$y_j(t) = \sqrt{\sum_{i=1}^I (x_i(t) \cdot w_{i,j}(t))^2}. \quad (26)$$

This module is connected to the labeling module at the training phase. For the testing phase, this module is switched to the mapping module. Moreover, this module is passed when input features are used without creating codebooks directly.

5.3.2 Labeling Module

The role of this module is to create labels used for category candidates. For this study, we created this module using ART, which is a theoretical model of unsupervised neural networks to create labels adaptively and incrementally with preservation of plasticity and stability together for time-series data.

In ART of various types [32], we use ART-2 [26], into which it enables input continuous values. The network of ART-2 comprises two fields: Field 1 (F1) for feature representation and Field 2 (F2) for category representation. Here, F1 comprises six sub-layers: p_i , q_i , u_i , v_i , w_i , and x_i . The sub-layers actualize Short-Term Memory (STM), which enhances features of input data and removes noise for a filter. Here, F2 actualizes Long-Term Memory (LTM) based on finer or coarser recognition categories. LTM is created in each unit assigned to independent labels. j -th unit of F2 and the sub-layer p_i are connected. Top-down weights Z_{ji} and bottom-up weights Z_{ij} are included. The weights are initialized as

$$Z_{ji}(0) = 0, \quad (27)$$

$$Z_{ij}(0) = \frac{1}{(1-d)\sqrt{J}}. \quad (28)$$

Therein, J is the number of units of F2. Subsequently, input data x_i are presented to F1; the sublayers are propagated as

$$w_i(t) = x_i(t) + au_i(t-1), \quad (29)$$

$$h_i(t) = \frac{w_i(t)}{e + \|w\|}, \quad (30)$$

$$v_i(t) = f(h_i(t)) + bf(q_i(t-1)), \quad (31)$$

$$u_i(t) = \frac{v_i(t)}{e + \|v\|}, \quad (32)$$

$$q_i(t) = \frac{p_i(t)}{e + \|p\|}, \quad (33)$$

$$p_i(t) = \begin{cases} u_i(t) & (\text{inactive}), \\ u_i(t) + dZ_{ji}(t) & (\text{active}), \end{cases} \quad (34)$$

$$f(x) = \begin{cases} 0 & \text{if } 0 \leq x < \theta, \\ x & \text{if } x \geq \theta. \end{cases} \quad (35)$$

Therein, a and b respectively denote coefficients of feedback loops from u_i to w_i and from q_i to v_i . θ is a parameter to control a noise detection level in v_i . Also, θ is a parameter to control the noise detection level in v_i . In addition, e is a coefficient to prevent zero from occurring in the denominator. The most active unit of its index c is sought as

$$c = \operatorname{argmax}_{1 \leq j \leq J} \left(\sum_i p_i(t) Z_{ij}(t) \right), \quad (36)$$

For c , weights are updated as

$$Z_{ci}(t+1) = Z_{ci}(t) + \alpha \cdot (p_i(t) - Z_{ci}(t)), \quad (37)$$

$$Z_{ic}(t+1) = Z_{ic}(t+1) + \alpha \cdot (p_i(t) - Z_{ic}(t)). \quad (38)$$

The vigilance threshold ρ is used to ascertain whether input data belong correctly to a category, as

$$\rho < e + \|r\|, \quad (39)$$

$$r_i(t) = \frac{u_i(t) + s \cdot p_i(t)}{e + \|u\| + \|s \cdot p\|}, \quad (40)$$

where s is a coefficient for propagation from p_i to r_i , and d is a learning rate coefficient. Furthermore, $s \cdot \alpha / (1 - \alpha) \leq 1$ is the constraint between them. When (39) is false, the active unit is reset and is searched to a next active unit. Repeat until the range of change of F1 is sufficiently small if (39) is true. Herein, teaching signals are used for labels if ACMNs are used for supervised learning. The index c is stored as a label if ACMNs are used for unsupervised learning.

5.3.3 Mapping Module

For this module, category maps are created as a learning result. We built this module using CPNs, which are supervised neural networks, to classify patterns into particular categories with the functions of competitive and neighborhood learning.

The network architecture of CPNs comprises three layers: an input layer, a mapping layer, and a Grossberg layer. The input layer and mapping layer resemble those of SOMs in MM. Teaching signals are presented to the Grossberg layer. For our method, labels that are assigned for F2 on ART-2 of the labeling module are used for teaching signals. Our method actualizes automatic labeling to combine CPNs with ART.

The order of units on F2 is assigned as labels used for teaching signals in the supervised learning mode. For the semi-supervised learning mode, mixed labels that include teaching signals and without teaching signals created from ART are mapped on the category map. For the unsupervised learning mode, labels obtained using ART are used for learning CPNs. The usage of labels differs in each learning mode. Using the intermediate representation as labels, this module performs similar learning behaviors in respective modes.

Learning results are represented as a category map on the mapping layer. Spatial relations among datasets based on similarity are visualized on a category map.

Actually, ACMNs create it automatically without setting the number of categories. Moreover, redundant labels are removed through the process of competitive and neighborhood learning.

The learning algorithm of CPNs is the following. Herein, for visualization characteristics of category maps, we set the mapping layer to a two-dimensional structure $X \times Y$ unit. For this study, we set one dimension of the input and Grossberg layers, although they can take any structure. The numbers of units are, respectively, I and K . $u_{i,j(x,y)}(t)$ are weights from an input layer unit i to a mapping layer unit $j(x,y)$ at time t . $v_{j(x,y),k}(t)$ are weights from an Grossberg layer unit k to a mapping layer unit $j(x,y)$ at time t . These weights are initialized randomly before learning. $x_i(t)$ are training data presented to the input layer unit i at time t . The unit for which the Euclidean distance between $x_i(t)$ and $u_{i,j(x,y)}(t)$ is the smallest, sought as the winner unit. $c(x,y)$ is the index of the unit.

$$c(x,y) = \underset{(1,1) \leq j(x,y) \leq (X,Y)}{\operatorname{argmin}} d_{j(x,y)}, \quad (41)$$

$$d_{j(x,y)} = \sqrt{\sum_{i=1}^I (x_i(t) - u_{i,j(x,y)}(t))^2}. \quad (42)$$

The neighborhood region $N_{c(x,y)}(t)$ around $c(x,y)$ is defined as

$$N_{c(x,y)}(t) = \lfloor \mu \cdot (X,Y) \cdot \left(1 - \frac{t}{O}\right) + 0.5 \rfloor, \quad (43)$$

where $\mu(0 < \mu < 1.0)$ is the initial size of the neighborhood region, and O is the maximum iteration for training. $u_{n,m}^i(t)$ of $N_{c(x,y)}(t)$ are updated to close input feature patterns using Kohonen's learning algorithm as

$$u_{i,j(x,y)}(t+1) = u_{i,j(x,y)}(t) + \Delta u, \quad (44)$$

$$\Delta u = \alpha(t)(x_i(t) - u_{i,j(x,y)}(t)). \quad (45)$$

Subsequently, $v_{j(x,y),k}(t)$ of $N_{c(x,y)}(t)$ are updated to close teaching signal patterns using Grossberg's learning algorithm.

$$v_{j(x,y),k}(t+1) = v_{j(x,y),k}(t) + \Delta v, \quad (46)$$

$$\Delta v = \beta(t)(T_k(t) - v_{j(x,y),k}(t)). \quad (47)$$

Therein, T_k are training signals obtained using ART-2. $\alpha(t)$ and $\beta(t)$ are learning coefficients that have decreasing values with the progress of learning. $\alpha(0)$ and $\beta(0)$ respectively denote the initial values of $\alpha(t)$ and $\beta(t)$. The learning coefficients are given as

$$\begin{bmatrix} \alpha(t) \\ \beta(t) \end{bmatrix} = \begin{bmatrix} \alpha(0) \\ \beta(0) \end{bmatrix} \cdot \left(1 - \frac{t}{O}\right). \quad (48)$$

In the initial stage, the learning is done rapidly when the efficiencies are high. In the final stage, the learning converges, although the efficiencies decrease. As the maximum number of $v_{j(x,y),k}(t)$ for the k -th Grossberg unit, category $L_k(t)$ is searched as

$$L_k(t) = \underset{(1,1) \leq j(x,y) \leq (X,Y)}{\operatorname{argmax}} v_{j(x,y),k}(t). \quad (49)$$

A category map is created after determining categories for all units. Test datasets are presented to the network that is created through learning. The mapping layer unit, which is the minimum of the Euclidean distance as the similarity of test data and feature patterns, is burst. Categories for these units are recognition results for CPNs.

5.4 Local Locomotion

We assume that our autopilot system is used only in indoor environments structured mainly with artificial materials. For indoor sensing, we use ranging sensors and depth sensors as shown in Fig. 8. To extract valid moving areas, we used depth sensors that can obtain 3D information. The basic locomotion is wall-following, as shown in Fig. 2 to measure the distance between Glimmer and walls using ranging sensors. Glimmer continues to move straight in the same direction if it has a sufficient distance to a wall.

5.4.1 Obstacle Sensing

Glimmer used depth sensors for obstacle recognition. Moreover, Glimmer determines motional states in each obstacle using distance changes obtained from depth sensors. Glimmer uses relative velocity between the speed of Glimmer and that of an object. Subsequently, Glimmer avoids an obstacle after checking other obstacles if it is a static object. After this locomotion, Glimmer returns to the original path planning to continue the former locomotion. Glimmer stops the locomotion if the obstacle is a moving object and getting closer. We consider that it is better to stop without taking active avoidance. Glimmer continues locomotion to maintain a distance to an obstacle if the obstacle is moving to the opposite direction.

Glimmer used a ranging sensor for distances up to 800 mm. This range is beyond the measurement target for the depth sensors. We used ranging sensors to stop Glimmer. The normal stop and emergency stop are switched using a threshold and measurement values obtained using ranging sensors. We set the threshold to 465 mm based on our experiment. The normal stop is selected if a measured value is greater than this threshold value. The emergency stop is selected if a measured value is smaller than this threshold. However, it is only the moving direction to stop. It is impossible for ranging sensors to ascertain details of the shape of an obstacle.

After stopping, Glimmer obtains object shapes and its moving status using depth sensors. Glimmer determines a subsequent behavior according to the moving status. No active behavior is conducted if objects are detected by sensors except for the direction of movement. Thereby, crashes from the rear are avoided. The probability of crashing into a moving object is increased if it is approaching from the rear. Our autopilot system is unsupported because it will be complex for sensing and generation of behavior patterns, although we consider that it is better to be distant from the object.

5.4.2 Path Planning and Obstacle Avoidance

We used PFMs to ascertain the direction of movement for autonomous locomotion. Actually, PFMs represent pop and push forces based on a magnet to pull between different poles and to push between same poles [23]. Let $U_a(r)$ and $U_b(r)$ respectively denote a pop force at a relay point s of the front and a push force from walls or obstacles. Presumably, there is a reflecting object at the position r . The potential energy $U(r)$ of this object is defined as

$$U(r) = U_a(r) + U_b(r). \quad (50)$$

Herein, $U(r)$ is defined as

$$U(r) = -\frac{A}{r^n} + \frac{B}{r^m}. \quad (51)$$

Let $F(r)$ be a force to be received of the wheelchair at r . $F(r)$ is obtained as differentiated $U(r)$ as

$$F(r) = -\frac{dU}{dr} = -\frac{nA}{r^{n+1}} + \frac{B}{r^{m+1}}. \quad (52)$$

Therein, A , B , n , and m are positive constants. We replace nA , mB , $n+1$, and $m+1$ to A' , B' , n' , and m' .

$$F'(r) = -\frac{A'}{r^{n'}} + \frac{B'}{r^{m'}}. \quad (53)$$

The first and second terms respectively address a vector $U_a(r)$ and a vector $U_b(r)$. The direction of the wheelchair is ascertained using a resultant vector of the two vectors. Let (x_b, y_b) be push vector components. Each component is represented as

$$x_b = x_0 + x_1 + \cdots + x_N, \quad (54)$$

$$y_b = y_0 + y_1 + \cdots + y_M, \quad (55)$$

where N and M are the numbers of respective components. Let (x_a, y_a) be pop vector components. Let (x_c, y_c) be resultant vector components. The orientation of movement F is obtained as

$$F = \sqrt{x_c^2 + y_c^2}, \quad (56)$$

where

$$x_c = x_a + x_b, \quad (57)$$

$$y_c = y_a + y_b. \quad (58)$$

Paths are updated using moving functions that are approximated to F . For this study, we created behavior patterns combined with moving functions of 12 types. Moreover, F is approximated to the direction of movement without turning of eight types.

6 Evaluation Experiment

6.1 Preliminary Experiment

The aim of this preliminary experiment is to predict errors between theoretical values and actual measurement values. Fig. 16 depicts the assignment of sensors.

Figure 16. Setting of the experiment.

Figure 17. Results of measurements for respective distances.

For a comparison of L' and the Ground Truth (GT) value L , we measured a distance between the sensors and the white panel among 100–800 mm with steps by 100 mm. The five sensors AN0–AN4 are used from the bottom to the top. The polling interval was 10 ms. We measured L' for five times at each distance. Glimmer obtained respective means distances to convert 10 bit values obtained from five sensors five times. Therein, the sensors are used independently.

Fig. 17 portrays the mean measurement values and standard errors in each distance. Detailed values can be referred from Table 2. The horizontal and vertical axes respectively show L and L' . Using a measuring tape (Shimadzu Corp.), L is obtained. The vertical bars attached to the mean values show standard errors. The slope of the graph approaches 1.0 as the ideal straight line if GT and measurement values are equal, which means high-precision sensing. The error between the mean measurement value and GT is 10.2% at the range of 100–300 mm. Moreover, the errors are ± 12.0 – 18.9% and the range of 400–700 mm and $\pm 32.5\%$ at 800 mm.

As a global tendency, the standard errors were increased according to the extension of the distance, es-

Table 2. Measurement accuracy of ranging sensor: A, GT distance obtained using a tape measure; B, mean measurement values; C, standard errors.

A [mm]	B [mm]	B-A [mm(%)]	C [±%]
100	90	-10 (-9.9%)	57
200	213	13 (6.7%)	25
300	331	31 (10.2%)	40
400	465	65 (16.2%)	30
500	560	60 (12.0%)	125
600	714	114 (19.0%)	110
700	760	60 (8.6%)	313
800	1060	260 (32.5%)	572

Figure 18. Assignment of Glimmer and the object as an obstacle.

pecially for 500–800 mm. For example, the measurement value was 714 mm at $L=600$ mm. However, the standard error is 5.65 % for the range of 100–400 mm without overlapping data. In addition, L' is longer than L for this range except for 100 mm. We consider that the range of 100–400 mm is suitable for use which shows small standard errors. Therefore, the valid range is used for $L'=90$ –465 mm for a valid range.

6.2 Experimental Setup

For all vehicles, the most important factor is to stop immediately and correctly. As a fundamental factor to maintain safety, this is the essential approach to stop movement for an autopilot. For this experiment, we tested the emergency stop capability of Glimmer in an actual environment.

Fig. 18 depicts the assignment of Glimmer and an object. The object has 390 mm height, 465 mm depth, and 480 mm width as its dimensions. Initially, the object is located 1,000 mm front from Glimmer. Glimmer moves straight in the direction of this object. After the forward movement, Glimmer stops if the measured distance to the object is less than 465 mm, which corresponds to the actual distance 400 mm. The distance is measured by the measuring tape used in the preliminary experiment. After stopping, Glimmer measures the distance to the object again. This trial was conducted for 10 iterations.

6.3 Theoretical Distance

Let t_a , t_b , and t_c respectively denote the void running time for stop control through the mean value detection, the polling time including A/D converting time, and the pin control time with a operational function. t_a is calculated as the summation of t_b and t_c as

$$t_a = t_b + t_c. \quad (59)$$

Using the Glimmer weight W and the wheel diameter D , the inertia moment J of the motor is calculated as

$$J = \frac{1}{8}d^2W. \quad (60)$$

Table 3. Values of respective parameters for calculating theoretical distance.

Parameter	Value	Unit
t_a	0.15	s
t_b	0.06	s
t_c	0.09	s
t_d	0.24	s
W	40	kg
d	0.20	m
J	0.21	kg·m ²
T_L	0.0203	N·m
γ	15	–
N	33.4	rpm
v_{max}	6.39	km/h
v_0	0.355	m/s
T_m	0.2	N·m
T_b	3.0	N·m
t	0.39	s
X	84.0	mm
L	400.0	mm
D	316.0	mm

The load torque T_L is calculated using the friction coefficient μ as

$$T_L = \frac{1}{4}\mu dW. \quad (61)$$

Using the rating torque T_m and the torque ratio γ , the brake torque T_b is calculated as

$$T_b = \gamma T_m. \quad (62)$$

We set the Glimmer maximum velocity v_m and the duty ratio r of PWM. Therefore, the Glimmer initial velocity v_0 is obtained as

$$v_0 = dv_{max}. \quad (63)$$

Using the torque ratio γ , the revolutions per minute N are

$$N = \frac{v_0}{\pi d}, \quad (64)$$

The total braking time t_d , which is the time from the port control to the stop, is the following.

$$t_d = \frac{JN}{9.55(T_L + T_b)} \quad (65)$$

The total braking time t is calculated as

$$t = t_a + t_d. \quad (66)$$

Let α be an acceleration for the constant acceleration straight movement from v_0 to the terminal speed $v = 0$ in 0.50 s. The total running distance x is calculated as

$$X = v_0t - \frac{1}{2}\alpha t^2. \quad (67)$$

Therefore, the theoretical distance D is the following.

$$D = L - X, \quad (68)$$

Table 3 portrays the values of respective parameters. Finally D is calculated as 316.0 mm for this experiment.

Table 4. Experimental results: A, distance between the object from the sensor; B, standard error from the theoretical value.

Trial	A [cm]	B [cm]
1	22.95	8.65
2	23.15	8.45
3	29.98	1.62
4	23.06	8.54
5	20.40	11.20
6	21.31	10.29
7	26.95	4.65
8	21.98	9.62
9	21.39	10.21
10	23.37	8.23
Ave.	23.45	8.15

6.4 Experimental Results

Table 4 shows the experimentally obtained results. For the measured distance with the sensor 400 mm, the mean distance between Glimmer and the object is 234.5 mm, which corresponds to 96.5 mm for the mean error to d . We infer that this error occurred from the slip between the barrel tires of the Mecanum wheels and the floor surface. The Mecanum wheel diameter is 203 mm. The circumference including the barrel tires is 637.4 mm. The gap separating the barrel tires is 53.1 mm because 12 barrel tires are attached at fixed intervals.

The barrel tires can be stopped stably if at least two tires are grounded. We consider that the gap distance between the tires increases the mean error for the theoretical distance. The rotation of the Mecanum wheels is suppressed because static torque is applied from the motor. The barrel tires over the Mecanum wheel rotate freely, although the static torque from the motor shaft is applied. We infer that the increased distance necessary to stop is related to the allowance of 28.58 mm of the diameter.

The error of the third measurement was the minimum 16.2 mm. In contrast, the error of the fifth measurement was the maximum 112.0 mm. The difference between the minimum and maximum values was 95.8 mm. To estimate the maximum of the variation, the average distance between the object and the sensors was 65.3 mm. We consider that the variation in the slip of the wheel was 36.0 mm because the standard measurement error of the sensor was 14.0 mm.

The mean error was 81.5 mm. We infer that minimum safety is guaranteed because the mean distance was 234.5 mm to the object before the collision, although extension of the braking distance results from this locomotion mechanism. In addition, Glimmer stopped within the range of 100–400 mm, which is the valid range of the ranging sensors. It is possible to correct this error in re-measuring the distance to the obstacle by inserting controls for reverse movement with avoidance of control after stopping.

7 Conclusion

This paper presented a fundamental design of an autopilot system to actualize autonomous locomotion of

electric wheelchairs with emphasis on simplicity and functionality. We designed a novel electric wheelchair with high mobility using Mecanum wheels that can move all directions without turning. Moreover, we developed a prototype with consideration of the exterior design as the concept of EPAMD, which incorporates ease of integration with a person's daily life. Our system used range sensors and depth sensors for environmental recognition to prevent collisions. We addressed global locomotion based on visual landmarks extracted using SMs and local locomotion using PFMs. For the evaluation experiment of emergency stop using ranging sensors, Glimmer stopped for mean errors of 81.5 mm while preserving the minimum of the error of braking distance of Mecanum wheels.

For future work, we shall strive to develop software to actualize fully autonomous locomotion as an autopilot to reduce the load for driving, especially for elderly people. Moreover, we must strive to improve safety using various sensors such as depth sensors and long-distance sensors. Furthermore, these developed systems will be combined with Micro Air Vehicles (MAVs) to extend sensing of the environment.

Acknowledgements

This research was supported by the Suzuki Foundation, Japan.

REFERENCES

- [1] White Paper on Aging Society 2012, Cabinet Office, Government of Japan, 2013.
- [2] Traffic Accidents of Electric Wheelchairs, Tokyo Metropolitan Police Department, 2013.
- [3] Safety Examination Report of Handle-Type Electric Wheelchairs, National Institute of Technology and Evaluation in Japan, 2007.
- [4] Y. Matsumoto, "Vision of Mobility Assistance," *Journal of the Robotics Society of Japan*, vol.27, no.6, pp.608–612, 2009.
- [5] K. Shirai, H. Madokoro, and K. Sato, "Proximity Sensing and Movements of Omnidirectional Mobile Electric Wheelchair for Autopilot," *Proc. Society of Instrument and Control Engineers (SICE) Annual Conference*, pp.1296–1301, Sep. 2014.
- [6] M. West and H. Asada, "Design of a Holonomic Omnidirectional Vehicle," *Proc. IEEE International Conference on Robotics Automation*, pp.97–103, 1992.
- [7] F.G. Pin and S.M. Killough, "A New Family of Omnidirectional and Holonomic Wheeled Platforms for Mobile Robots," *Trans. Robotics and Automation*, vol.10, no.4, pp.480–489, 1994.
- [8] U. Borgolte, H. Hoyer, C. Buhler, H. Heck, and R. Hoelper, "Architectural Concepts of a Semiautonomous Wheelchair," *Journal of Intelligent and Robotic Systems*, vol.22, no.3–4, pp.233–253, 1998.

- [9] M. Wada and H. Asada, "Design and Control of a Variable Footprint Mechanism for Holonomic Omnidirectional Vehicles and its Application to Wheelchairs," *Trans. Robotics and Automation*, vol.15, no.6, pp.978–989, 1999.
- [10] H. Kitagawa, T. Kobayashi, T. Beppu and K. Terashima, "Semi-Autonomous Obstacle avoidance of Omnidirectional Wheelchair by Joystick Impedance control," *Proc. IEEE/RSJ International Symposium on Intelligent Robots and Systems*, pp.2148–2153, 2001.
- [11] H. Kitagawa, T. Beppu, T. Kobayashi and K. Terashima, "Motion Control of Omnidirectional Wheelchair Considering Patient Comfort," *Proc. World Congress International Federation of Automatic Control*, 2002.
- [12] M. Wada, "Omnidirectional Motion Control and Prototyping of a Four-Wheel-Drive Electric Wheelchair," *Journal of Robotics Society of Japan*, vol.27, no.3, pp.314–324, 2009.
- [13] H. Kitagawa, T. Beppu, Y. Ohno, T. Miyoshi, and K. Terashima, "Motion Control of Omnidirectional Mobile Wheelchair Considering User's Comfort," *Journal of the Robotics Society of Japan*, vol.22, no.7, pp.933–939, 2004.
- [14] K. Yano, N. Oguro, and K. Terashima, "Starting Control with Vibration Damping by Hybrid Shaped Approach Considering Time and Frequency Specifications," *SICE Journal of Control, Measurement, and System Integration*, vol.37, no.5, pp.403–410, 2001.
- [15] S. Hemachandra, T. Kollar, N. Roy, and S. Teller, "Following and Interpreting Narrated Guided Tours," *Proc. IEE International Conference on Robotics and Automation*, pp.2574–2579, 2011.
- [16] T. Bailey and H. Whyte, "Simultaneous Localization and Mapping: Part-II," *IEEE Robotics & Automation Magazine*, vol.13, no.3, pp.108–117, 2006.
- [17] R. Suzuki, E. Takano, Y. Kobayashi, Y. Kuno, K. Yamazaki, and A. Yamazaki, "Robotic Wheelchair Easy to Move and Communicate with Companions," *Proc. Extended Abstracts on Human Factors in Computing Systems*, pp.3079–3082, 2013.
- [18] C.K. Chang, C. Siangian, and L. Itti, "Mobile Robot Vision Navigation & Localization using Gist and Saliency," *Proc. IEEE /RSJ International Conference on Intelligent Robots and Systems*, pp.4147–4154, 2012.
- [19] K. Ho and P. Newman, "Using Visual Saliency and Geometric Sensing for Mobile Robot Navigation," *Dphil Transfer Report, Robotics Research Group Dept. of Engineering Science University of Oxford*, 2004.
- [20] B.E. Ilon, "Wheels for a course stable selfpropelling vehicle movable in any desired direction on the ground or some other base," US Patent 3876255, Apr. 1975.
- [21] O. Diegel, A. Badve, G. Bright, J. Potgieter, and S. Tlale, "Improved Mecanum Wheel Design for Omnidirectional Robots," *Proc. Australasian Conference on Robotics and Automation*, pp.117–121, Nov. 2002.
- [22] Itti, L. and Koch, C. and Niebur, E. and others, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.20, no.11, pp.1254–1259, 1998.
- [23] J.H. Chuang, C. Ahuja, and N. Ahuja, "An Analytically Tractable Potential Field Model of Free Space and Its Application in Obstacle Avoidance," *Trans. System, Man, and Cybernetics Part B*, vol.28, no.5, pp.729–736, 1998.
- [24] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem," *Proc. AAAI National Conference on Artificial Intelligence*, pp.593–598, 2002.
- [25] T. Kohonen, *Self-Organizing Maps*, Springer Series in Information Sciences, 1995.
- [26] G.A. Carpenter and S. Grossberg: "ART 2: Stable Self-Organization of Pattern Recognition Codes for Analog Input Patterns", *Applied Optics*, vol. 26, pp. 4919–4930, 1987.
- [27] R.H. Nielsen, "Counterpropagation networks," *Applied Optics* vol.26, pp.4979–4983, 1987.
- [28] D.G Lowe, "Object recognition from local scale-invariant features," *Proc. International Conference on Computer Vision*, vol.2, pp.1150–1157, 1999.
- [29] J. McQueen, "Some Methods for Classification and Analysis of Multivariate Observations," *Proc. Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pp.281–297, 1967.
- [30] J. Vesanto and E. Alhoniemi, "Clustering of the Self-Organizing Map," *IEEE Trans. Neural Networks*, vol.11, no.3, pp.586–600, 2000.
- [31] M. Terashima, F. Shiratani, and K. Yamamoto, "Unsupervised Cluster Segmentation Method Using Data Density Histogram on Self-Organizing Feature Map," *Trans. Institute of Electronics, Information and Communication Engineers*, vol.J79-D-2, no.7, pp.1280–1290, 1996.
- [32] G.A. Carpenter and S. Grossberg: "Pattern Recognition by Self-Organizing Neural Networks", The MIT Press, 1991.
- [33] H. Madokoro and K. Sato, "Adaptive Mapping Networks for Visualizing Topological Structure of Data on Category Maps," *Intelligence and Informatics (Journal of Japan Society for Fuzzy Theory and Intelligence Informatics)*, vol.26, no.6, pp.903–912, 2014.
- [34] A. Ultsch and H.P. Siemon, "Kohonen's Self Organizing Feature Maps for Exploratory Data Analysis," *Proc. International Neural Network Conference*, 1990.
- [35] K. Nakano, "How to make the Brain – Consideration the life through building a robot," Kyoritsu Publishing, 1995.
- [36] Y. Koren and J. Borenstein, "Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation," *Proc. of IEEE International Conference on Robotics and Automation*, pp.1398–1404, 1991.
- [37] D.H. Kim and S. Shin, "Local Path Planning Using a New Artificial Potential Function Configuration and Its Analytical Design Guidelines," *Advanced Robotics*, vol.20, pp.115–135, 2006.
- [38] T. Murata and H. Ishibuchi, "MOGA: Multi-objective genetic algorithms," *Proc. IEEE International Conference on Evolutionary Computation*, vol.1, 1995.

- [39] T. Lee, "Image Representation Using 2D Gabor Wavelets," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.18, no.10, pp.959–971, 1996.