Collision Avoidance Algorithm Using Deep Learning Type Artificial Intelligence for a Mobile Robot

Takeaki Takiguchi, Jae Hoon Lee, *Member, IAENG*, and Shingo Okamoto, *Member, IAENG*

***Abstract*—This paper presents an intelligent navigation system to generate safe motion for a mobile robot. It consists of two main modules of recognizing obstacle and making decision with artificial intelligence, respectively. Firstly, a recognition algorithm using laser range finder (LRF) and robot odometry is developed to detect objects near the robot. The proposed recognition algorithm provides both position and velocity information of obstacles by using range data accumulated for a certain time period. Then, the result is used for computing a safe moving direction of the robot to avoid collision with objects. For that, an artificial intelligence algorithm of multi-layered neural network was designed and trained by deep learning method with many data sets of information including pairs of sensor data and its solution of motion command for various situations.**

***Index Terms*— Artificial Intelligence, Deep learning, Mobile robot, Collision avoidance**

I. Introduction

According to advancement of sensor technology, many researches related to algorithms for utilizing sensor data for autonomous navigation of mobile robot have been carried out during last several decades. The technologies have been developed actively and applied to essential algorithms such as map building, positioning, path planning, safe motion generation, and so on [1-3]. Meanwhile, artificial intelligence has been showing remarkable advances in image recognition area, especially technologies for autonomous driving of vehicle [4-6].

Mobile robots for service tasks in human daily environments are required to have a capability to recognize the situation near itself and avoid collision to objects. It is necessary to calculate reciprocal positions and velocities between obstacles and to decide a next action in order to avoid them [7-9]. Typical collision avoidance technologies basically assume that most situations near the robot including moving obstacles could be recognized sufficiently. Thus, multiple target tracking technique, for detecting and tracking each object individually, is employed in spite of its complex algorithm and heavy computation burden. Besides, decision making algorithm for computing motion command also requires computation power in general.

Therefore, an intelligent navigation method that does not relying on multiple target tracking algorithm but exploiting artificial intelligence for processing sensor data is investigated in this research. This paper proposes two algorithms. The first one is a method to estimate position and velocity of multiple objects with one LRF and robot odometry not by individual tracking but by manipulating a set of range scan data accumulated for a certain time period. The second one is an artificial intelligence to compute safe motion direction to avoid moving objects. Though human being has no capability to measure accurate position and velocity of objects near him, he can generate optimal decision for safe motion to a goal position with 'feel' like overall sense about his surroundings. An algorithm to realize motion planning of such human sense by optimizing parameters with many sets of sensor and human solution data for searching safe motion direction was proposed in this paper.

This paper is organized as follows. The proposed method to estimate objects' position and velocity by using LRF data sets and its experimental result are addressed in section 2. The design of artificial intelligence to search safe motion direction and its optimization process are explained in section 3. In section 4, experimental results of collision avoidance navigation with a mobile robot installed with the improved algorithm are explained. Finally, conclusions and further researches are addressed in section 5.

II. Estimation of Position and Velocity of Moving Objects Using LRF Data Sets

A. Process of Estimating Position and Velocity with LRF Point Cloud

Typical method for estimating velocities of multiple objects is to track each object individually that requires complex algorithm and computation power. However, the proposed method computes position and velocity information of all detected objects not by individual tracking but by manipulating scan data sets synthetically. Data sets obtained by accumulating multiple scan data for a certain time period are represented as point clouds in a three dimensional space where its time information is used as vertical coordinate. Where moving objects are represented as clusters of multiple points, called as point cloud. Geometry of a point cloud of an object is dependent to its shape as well as moving velocity. Particularly, its inclination is changed according to object's velocity. Therefore, velocity of an object can be estimated

Manuscript received December 11, 2017; revised February 11, 2018.
All authors are with Mechanical Engineering Course, Graduate School of Science and Engineering, Ehime University, 3 Bunkyo-cho, Matsuyama 790-8577, Japan.
Takeaki Takiguchi (e-mail: d840016b@mails.cc.ehime-u.ac.jp)
Jae Hoon Lee (e-mail: lee.jaehoon.mc@ehime-u.ac.jp)
Shingo Okamoto (e-mail: okamoto.shingo.mh@ehime-u.ac.jp)

from the slope of object's point cloud. It is the main idea of this research to estimate objects' velocity.

Specific processes of the proposed method are as follows. LRF data sets taken in a certain period are classified according to the distances between points as clusters of point cloud for each object. And the center coordinates of gravity of the lower half and the upper half parts are calculated. As a result, average velocity of each object is expressed by the inclination of the line connecting centers of both parts of each object's point cluster. Where LRF data taken from ten times of successive scanning is defined as one set for each computation in this research. And, x and y coordinates of each point are computed in meter unit. The sampling time (Δt) of scanning, 25 milliseconds in this research, is transformed as 0.025 meters as z coordinate in the three dimensional space. Point Cloud Library (PCL) was used for computation of clustering and finding center of each cluster [10-12].

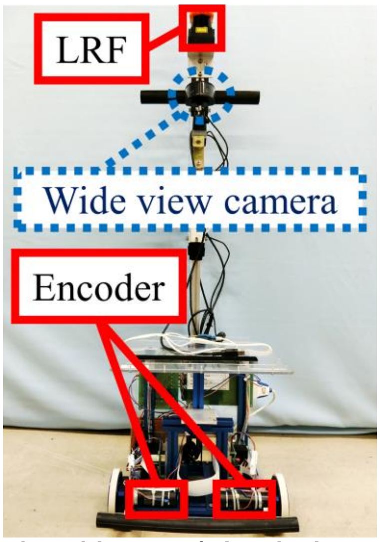


Fig.1 Mobile robot with laser range finder and wide view camera

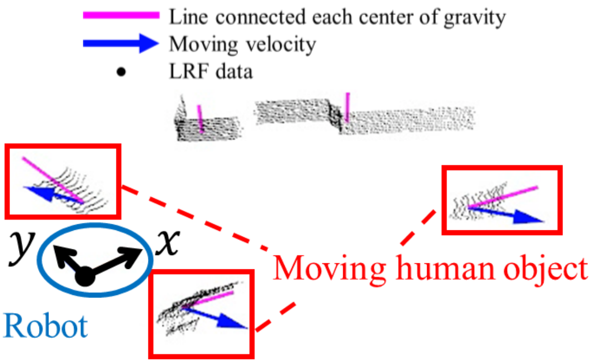


Fig. 2 Experimental results of estimating positions and velocities of three moving human objects

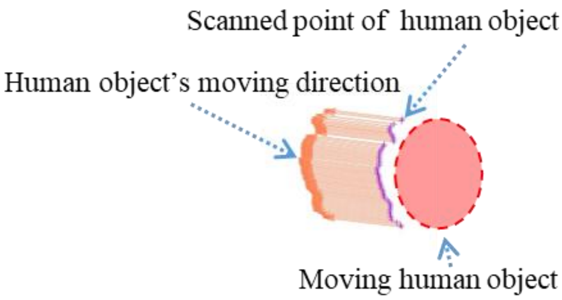


Fig. 3 An example of estimated velocity of an object transformed on two dimensional space from point cloud data

B. Experimental Works of Estimating Positions and Velocities of Moving Objects

The mobile robot used in this research is shown in Fig. 1. A LRF (UTM-X001s, HOKUYO Automatic Co., LTD.) was employed as a main sensor to detect objects near the robot. It was attached at the height of about 1 meter on the robot to detect human body near waist. A wide view camera was also attached under the LRF of the robot to capture experimental situations. While the mobile robot is moving in indoor environment according to manual motion command, estimation of positions and velocities of objects around the robot was conducted with the proposed algorithm.

An experimental result of estimating objects' position and velocity with multiple scan data taken from the moving mobile robot is shown in Fig. 2. It displays the situation of experiment where three human objects are walking near the robot. Where LRF data cluster of objects are represented with points, and their lines connecting both centers of lower and upper parts of clusters are also depicted as solid lines. The arrow lines representing moving velocity of detected objects are drawn by projecting the solid lines onto the ground. So the length of arrow line is dependent to the velocity of the object, i.e., long arrow line denotes fast motion. It has been observed that the position and velocity of multiple objects can be estimated with the proposed algorithm. Finally, an example of estimated velocity transformed to two dimensional data is shown as an enlarged picture in Fig. 3.

III. Artificial Intelligence Algorithm to Search Safe Moving Direction of Mobile Robot

A. Design of Artificial Intelligence Algorithm with Neural Network

An algorithm for searching safe moving direction was deep learned using many data sets of LRF data and solution. Training data sets including position as well as velocity of objects were prepared by computational simulation using RVO2 (Reciprocal Velocity Obstacle 2) library [13]. It is an open source which can compute a simulation of collision avoidance between multiple moving objects. After learning with data sets, the artificial intelligence algorithm could have the capability of computing a solution that is used as motion command to make mobile robot to head for a goal point safely.

For learning process of artificial intelligence algorithm, a computer equipped with CPU of Intel Core i7-7700 3.60 GHz and GPU of NVIDIA GeForce GTX 1080 was used in this research. The program of artificial intelligence algorithm was

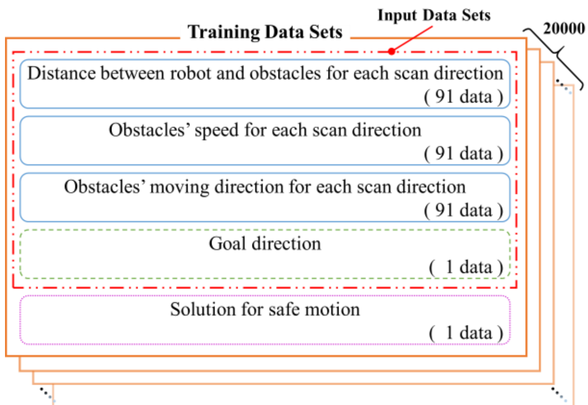


Fig. 4 Structure of training data sets used in deep learning considered velocity of object

developed with library of TensorFlow in Python programming environment [14].

The proposed algorithm was designed with a neural network to compute safe motion based on both position and velocity of objects near the robot. The structure of training data sets used in deep learning is shown in Fig. 4. Distance between the mobile robot and obstacles, and velocity of obstacles are given as values for each scan direction. The front area of 180 degrees of the robot is set to sensing area whose radius is set to 3 meters. Where, scan direction means one of 90 directions which divide the sensing area of 180 degrees. In addition, all measured data are transformed on the robot's moving coordinate system.

Fig. 5 shows the schematic of neural network having three layers designed for computing safe moving direction. The input data set of 274 data combined with 91 distance parameters between the mobile robot and objects, 91 speed and 91 direction parameters of objects' velocities, and 1 goal direction parameters, respectively. The first layers have 2048 nodes with hyperbolic tangent type activation, and the second and third layer have 2048 nodes where their outputs are transferred to the next layer through sigmoid type activation function. Finally, the output layer consists of 20 nodes where soft-max function was employed for each node. Resultantly,

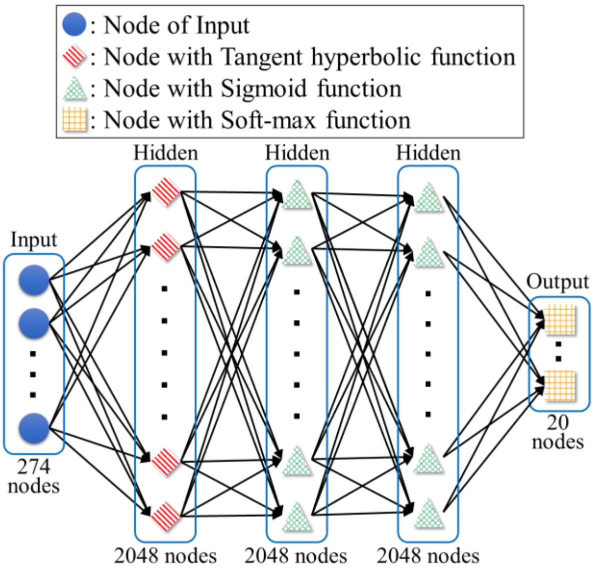


Fig. 5 Schematic of neural network designed for computing safe moving direction based on objects' velocity and position

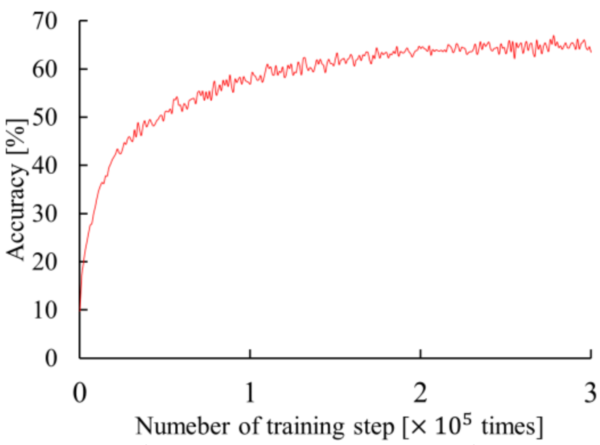
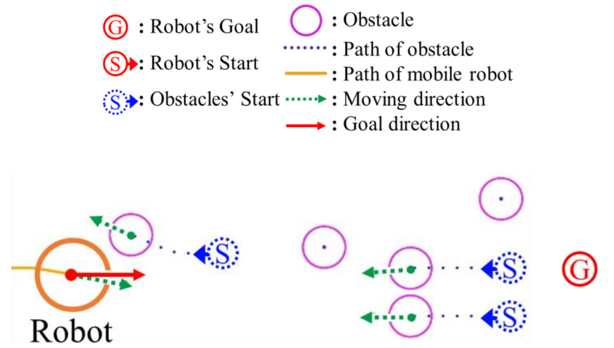


Fig. 6 Change of accuracy according to the number of training steps (The accuracy was calculated for every training step by comparison between the computed output with optimized parameters and the solution of training data)

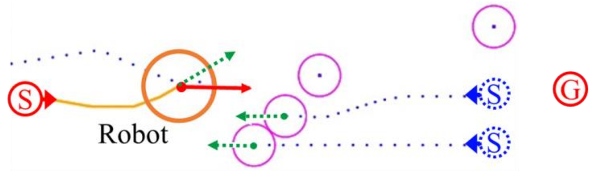
the output of the artificial intelligence algorithm is given as one of 19 directions dividing the sensing area of 180 degrees. In other words, the angular resolution of algorithm's output is set to 10 degrees. In the case of no safe direction, the output becomes 20 and the robot should stop.

B. *Optimization of Artificial Intelligence with Training Data*

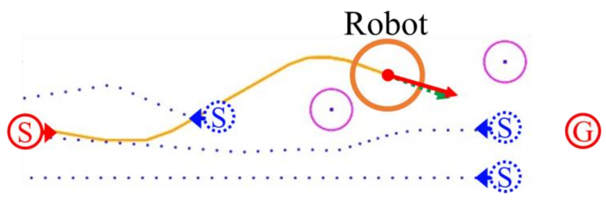
The artificial intelligence algorithm of neural network was learned by AdamOptimizer in TensorFlow with an index of minimizing error of cross-entropy loss function. The parameters of neural network were optimized with automatically adjusted learning rate by using 100 sets of batch data randomly chosen from learning data sets, where 20,000 sets of data was prepared for learning process.



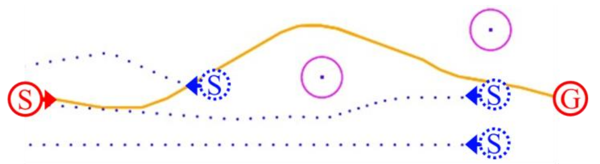
(a) When mobile robot avoids a moving objects



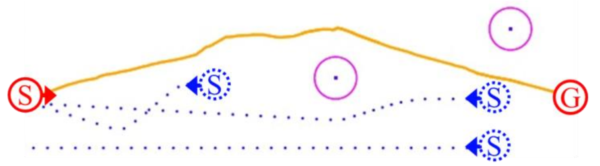
(b) When mobile robot avoids two moving objects



(c) When mobile robot avoids static objects



(d) Resultant paths of mobile robot and moving objects using NN with both position and velocity information of objects



(e) Resultant paths of mobile robot and moving objects using NN with only position information of objects for the same scenario with above simulation

Fig. 7 Computational results of the proposed algorithm computing safe motion command based on both position and velocity of objects

The change of optimization accuracy according to iterations of deep learning process is shown in Fig. 6. The accuracy is obtained by comparing the solution of data sets to the computed output of the algorithm with the optimized parameters at each training step. It was observed that the resultant accuracy of the proposed algorithm is about 65%.

C. Computation Results of the Proposed Algorithm

In order to confirm the proposed algorithm, computer simulation using RVO2 software was conducted. A simulation result with the algorithm considering velocity information is indicated in Fig. 7 (a) to (c). Where, the robot starts from a point, then moves to the right direction slightly to avoids a moving object in left front area as shown in Fig. 7 (a), and moves to the left direction to avoid moving two objects in right front and a static object in front as shown in Fig. 7 (b), then moves to the right to avoid a static object as shown in Fig. 7 (c). It is shown that the mobile robot installed with the proposed algorithm optimized by the training data sets has capability to avoid both static and moving objects. Meanwhile, Fig. 7 (d) shows another result with the proposed algorithm not considering velocity information. By comparing Fig. 7 (d) with Fig. 7 (e), it is denoted that the mobile robot using both position and velocity information of objects moves more smoothly than that using only position information. It is supposed that the computed artificial intelligence has function to predict next actions of objects by using their velocity information and also provide the safe moving direction to the mobile robot with predicted information. In addition, though the accuracy of learning is not so high as shown in Fig. 6, most incorrect computation outputs of the algorithm were also given as similar one to the correct solution. In other words, incorrect outputs are also near to the correct solution and its error is not so large.

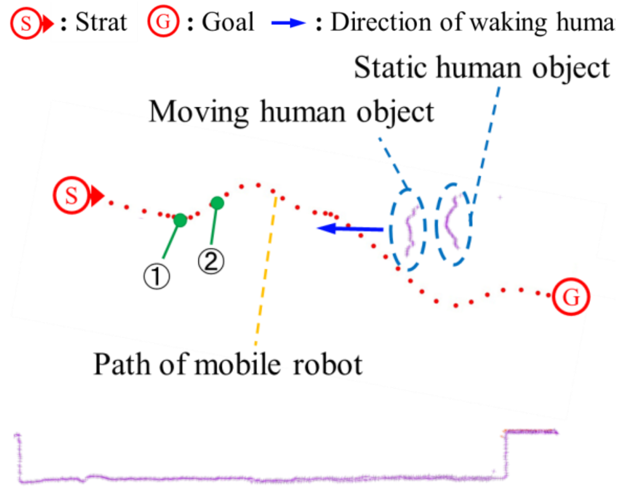
D. Experimental Result of Collision Avoidance with the Proposed Algorithm Used in Computational Simulation

An experimental work was carried out with a mobile robot in indoor environment. The experiment is that the mobile robot goes to a goal with avoiding collision to a walking human object and a standing human object. The result is shown in Fig. 8. It was observed that the proposed algorithm generates a safe direction. However, it was difficult to avoid a moving human object because of the slow response of the mobile robot and difference of conditions between simulation and experimental environments.

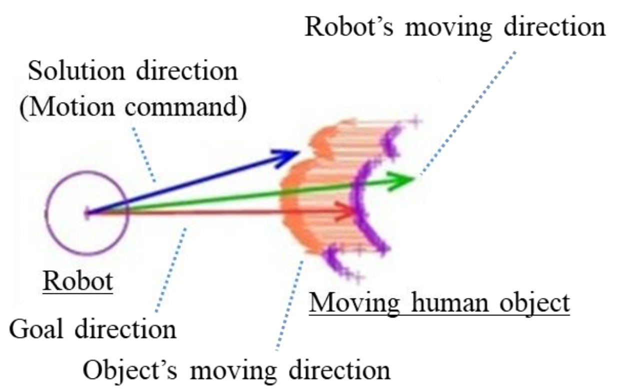
IV. Improvement of Collision Avoidance Algorithm with Data Set of Human Decision

A. Training Data Set for Experimental Works

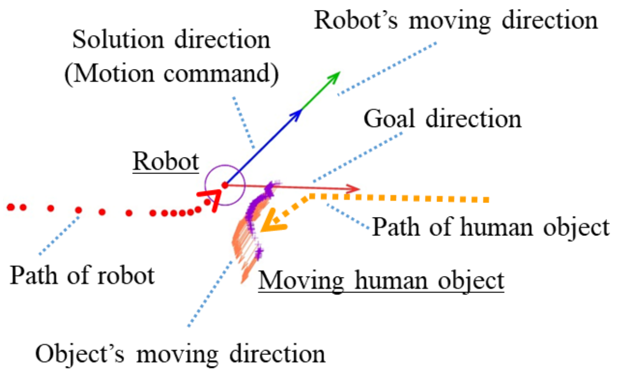
The data sets were manually prepared for various situations. They include many cases where objects exist near the robot, and the correct solution of safe moving direction for the mobile robot was made based on human intelligence, namely designer's decision. Especially, the predicted positions of moving objects were considered for making



(a) Resultant motion of the robot avoiding a moving human object and a static human object



(b) Enlarged graph of the computation result of the proposed algorithm for detecting objects and collision avoidance when the robot was moving near the position 1 in Fig. 8 (a)



(c) Enlarged graph of the computation result of the proposed algorithm for detecting objects and collision avoidance when the robot was moving near the position 2 in Fig. 8 (a)

Fig. 8 Results of experiments to avoid static and moving human objects

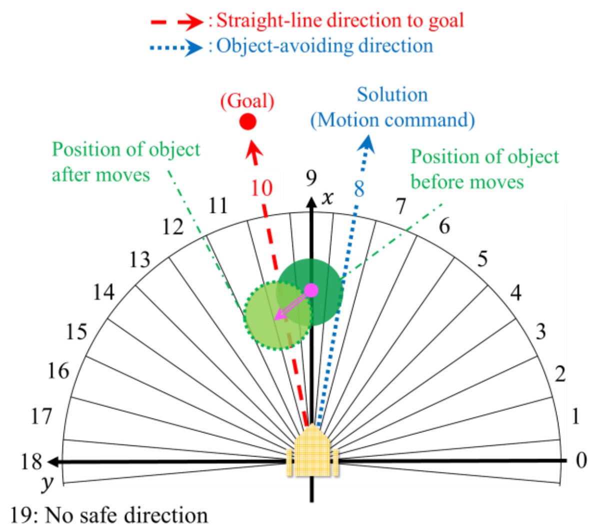
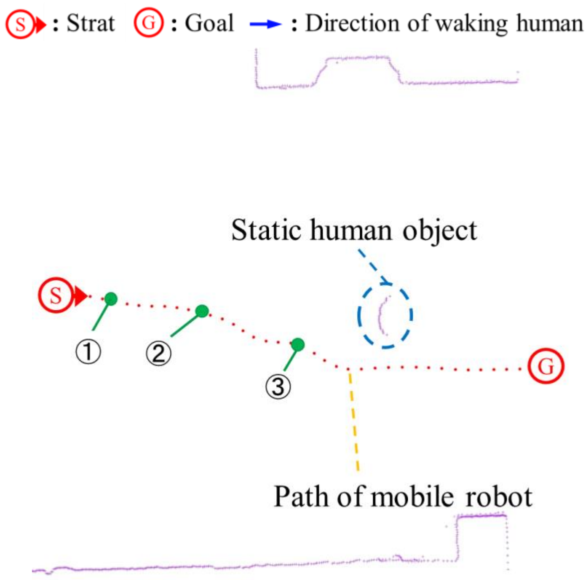
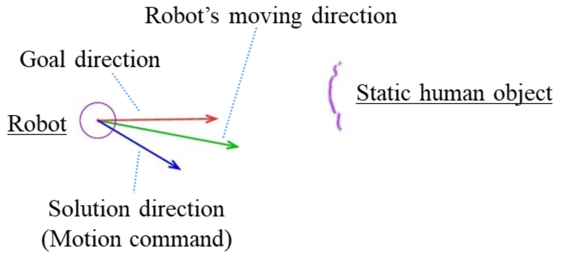


Fig. 9 An example of solution for the proposed algorithm in a situation when the robot avoids a moving human object

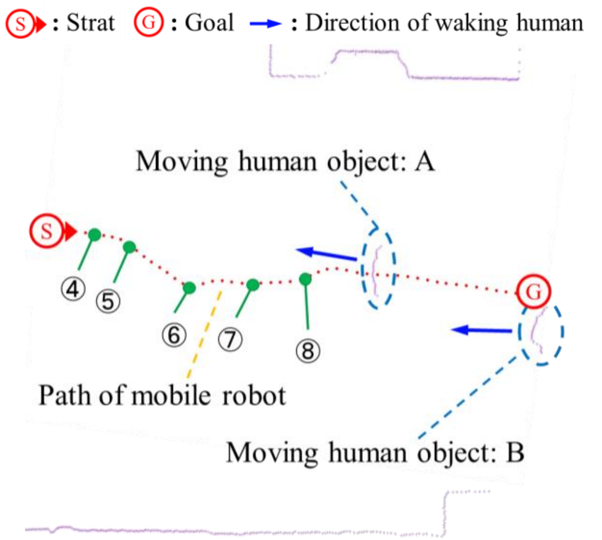
ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)



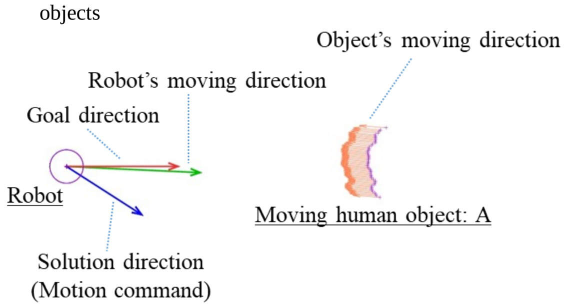
(a) Resultant motion of the robot avoiding a static human object



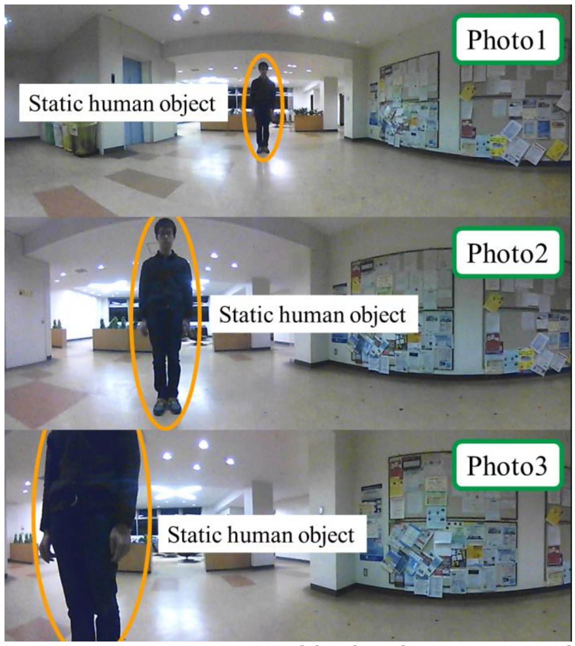
(b) Enlarged graph of the computation result of the proposed algorithm for detecting objects and collision avoidance when the robot was moving near the position ② in Fig. 10 (a)



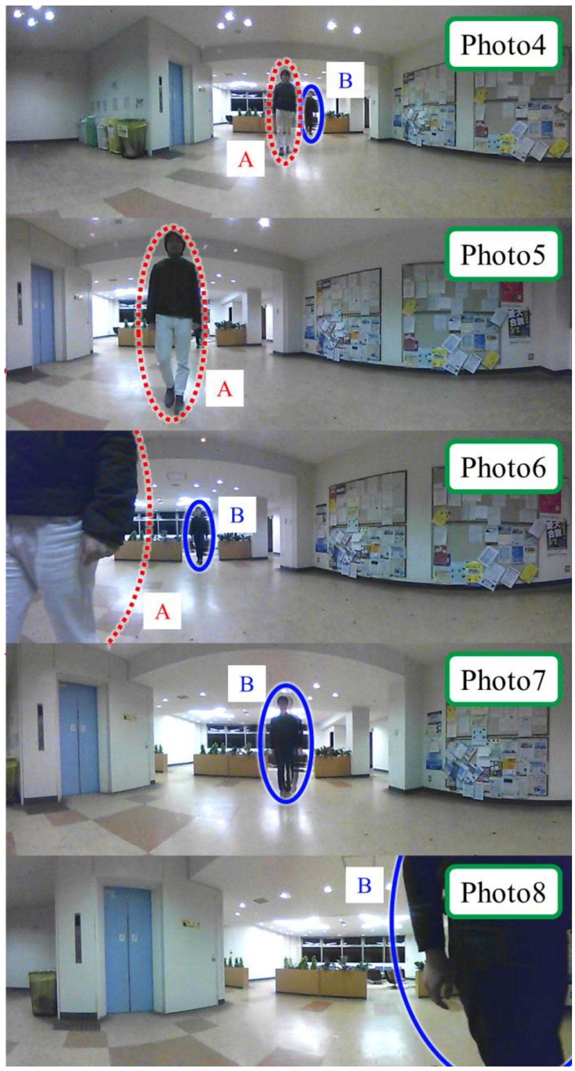
(c) Resultant motion of the robot avoiding two moving human objects



(d) Enlarged graph of the computation result of the proposed algorithm for detecting objects and collision avoidance when the robot was moving near the position ④ in Fig. 10 (c)



(e) Front view pictures captured by the robot passing near the position ① to ③ in Fig. 10 (a)



(f) Front view pictures captured by the robot passing near the position ④ to ⑧ in Fig. 10 (c)

Fig. 10 Results of experiments to avoid static and moving human objects

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

solutions for the training data sets. A representative example of solution in a situation including a moving object is displayed in Fig. 9. It shows a case when there is a moving object in front of the robot and the direction to goal is the direction of number '10' depicted with a red arrow line. If the robot just keeps moving forward, it will collide to the object. However, the next position of the moving object in next scan time can be predicted with its velocity information. So the correct solution given in training data set should be the direction of number '8' depicted with a dotted blue line in the figure. Where, the same NN of section III. *A* was used in the improved algorithm.

B. Collision Avoidance Experiment

In order to confirm the performance of the proposed algorithm, two experimental works were carried out with a mobile robot in indoor environment. The first experiment is that the mobile robot goes to a goal with avoiding collision to a standing human object. The second experiment is to avoid two walking human objects. Where, the computer used for the experiment of estimating positions and velocities of obstacles was utilized in the experimental works of collision avoidance. The resultant motions of the robot generated with the algorithm during the experiments are shown in Fig. 10. It was confirmed that the robot with the proposed algorithm can move to the goal position safely in the situations including static and moving human obstacles. Examples of the computation result by the proposed algorithm for a certain instance taken from the experiments are shown in Fig. 10 (b) and Fig. 10 (d). It is observed that the algorithm estimates both position and velocity of objects near the robot and generates a safe direction.

V. CONCLUSIONS

A synthetic algorithm to estimate position and velocity information of multiple moving objects by using multiple sets of LRF data cumulated for a short period was proposed in this research. In addition, artificial intelligence algorithm of neural network utilizing both position and velocity information of objects and being trained by deep learning method with big sensor data was also investigated for searching safe direction of a mobile robot. From the results of computer simulations and experiments, it was confirmed that the mobile robot with the proposed algorithm is capable of avoiding collision with both static objects and moving objects effectively.

As further study, research work is expected for an algorithm to search safe direction for a mobile robot moving in a narrow space like a passage.

REFERENCES

[1] S. Kita, J. H. Lee, S. Okamoto, "Experiments of NDT-Based Localization for a Mobile Robot Moving Near Buildings," *Proceedings of IMECS 2016*, Vol. 1, pp. 63-68, 2016.

[2] K. Okada, J. H. Lee, S. Okamoto, "People Tracking by Collaboration between a Mobile Robot and Static Laser Scanner," *Proceedings of IMECS 2014*, Vol. 2, pp. 646-651, 2014.

[3] M. Jaimez, J. Gonzalez-Jimenez, "Fast Visual Odometry for 3-D Range Sensors," *IEEE Transactions on Robotics*, Vol. 31, No. 4, pp. 809-822, 2015.

[4] C. Chen, A. Seff, A. Kornhauser, J. Xiao, "Deep Driving: Learning Affordance for Direct Perception in Autonomous Driving," *Proceedings of ICCV*, pp. 2722-2730, 2015.

[5] P. Long, W. Liu, J. Pan, "Deep-Learned Collision Avoidance Policy for Distributed Multiagent Navigation," *IEEE Robotics and Automation Letters*, Vol.2, No.2, pp. 656-663, 2017.

[6] L. Tai, S. Li, M. Liu, "A Deep-network Solution Towards Model-less Obstacle Avoidance," *Proceedings of IROS*, 2016.

[7] J. Snape, J. van den Berg, S. J. Guy, D. Manocha, "The Hybrid Reciprocal Velocity Obstacle," *IEEE Transactions on Robotics*, Vol.27, No. 4, pp. 696-706, 2011.

[8] J. Snape, J. van deg Berg, S. J. Guy, D Manocha, "Smooth and Collision-Free Navigation for Multiple Robots Under Differential-Drive Constraints," *Proceedings of IROS*, 2010.

[9] J. van den Berg, S. J. Guy, M. Lin, D. Manocha, "Reciprocal n-body collision avoidance," *Proceedings of ISRR*, 2009.

[10] Willow Garage, 2011, "PCL - Point Cloud Library (PCL)," < http://pointclouds.org/ >, access 2017.

[11] T Takiguchi, J. H Lee, S Okamoto, "Velocity estimation of moving obstacles with accumulated range data and safe motion generation of a mobile robot using artificial intelligence," *Proceedings of AROB*, pp. 940-945, 2018.

[12] T Takiguchi, J. H Lee, S Okamoto, "Velocity estimation method of moving objects with laser range scanner and computational algorithm to search safe moving direction using deep learning type artificial intelligence," *Proceedings of SICE4*, pp. 69-71, 2017.

[13] J. van den Berg, S. J. Guy, J. Snape, M. C. Lin, D. Manocha, 2008, "RVO2 Library: Reciprocal Collision Avoidance for Real-Time Multi-Agent Simulation," <http://gamma.cs.unc.edu/RVO2/>, access 2018.

[14] Google, 2015, "TensorFlow TM," < https://www.tensorflow.org/ >, access 2017.