A Neural Network-Based Navigation System for Mobile Robots

K.C. Koh*, H.R. Beom*, J.S. Kim** and H.S. Cho***

* Research Engineer, Research & Development Laboratory
Gold Star Industrial Systems, Co. Ltd. Anyang, Korea.
** Research Engineer, Production Technology Division
Samsung Electronics Co. Ltd. Suwon, Korea.
*** Professor, Department of Precision Engineering and Mechatronics,
KAIST, 373-1, Kusong-dong, Yusong-gu, Taejon,305-701, Korea.

Abstract

For mobile robots to be autonomous, they should have essential functional capabilities such as determination of their current location and heading angle, path control in order to follow the desired path and local path planning for uncertain environments. This paper deals with the above three issues and illustrates how the artificial neural network can be utilized to solve such problems. This neural network-based navigation system offers a method of determining the mobile robot's position; a 3-D landmark sensing system with neural estimator. It also offers a neural net-based feedforward controller designed to accurately track a desired path and a sensor-based local path planning capability to adapt to complex and changing environments. System software/hardware architecture to implement the above functional capabilities are discussed and some experimental and simulation results are illustrated to show the effectiveness of the proposed navigation system.

1. Introduction

Due to extensive research efforts, mobile robots are new becoming autonomous, intelligent and self-adaptive to environmental changes. Environment perception and modelling, path planning, localization and path tracking control are the essential capabilities required for intelligent mobile robots. In the situations where mobile robots navigate through complex and dynamically changing environments where unknown obstacles are located on a priori planned path, it is essential that the robots possess the following capabilities; decision-making based upon knowledge acquisition and past experiences, cognition of environment changes and learning of obstacle avoidance behavior. In recent years, there have been many research efforts to develop such capabilities. One of such efforts is the use of artificial neural network in the navigation task domains mentioned in the above [1 - 3].

This paper addresses the neural network applications in mobile robot navigation. In particular, local path planning, localization and path tracking control are treated for application tasks. Fig.1 shows a typical navigation space through which a mobile robot moves along a desired path from the start position to the goal position. The workspace consists of the desired path planned by global path planner, unexpected obstacles in the space and landmarks fixed at the known locations. The obstacles are assumed to be unexpected and unknown and, thus, the mobile robot should navigate in uncertain environment towards the goal position, avoiding the obstacles on the path. To this end we present a neural network-based navigation system suitable for navigation in uncertain environments. The proposed system is designed to follow a desired path, to control mobile robot's motion by accurately estimating its position when deviated from the desired path and to avoid obstacles which may be located on the preplanned path. To demonstrate the effectiveness of navigation performance the navigation system is implemented in an autonomous mobile robot, named as LCAR.

2. System Description

A proposed navigation system is used for a wheeled mobile robot, LCAR which has been

0-7803-1901-X/94 $4.00 ©1994 IEEE

developed in the Laboratory for Control Systems and Automation, KAIST. As shown in Fig.2, the robot has four wheels; two driven wheels are fixed at both sides of the mobile robot and two castors are attached at the front and rear sides of the robot, respectively. The robot moves and changes its heading angle by the rotation of two driven wheels. Navigation system consists of three major parts; a main control computer, a wheel servo system, and sensors with their processing systems. The major function of the main control computer is to supervise sub-systems during the navigation and perform the path planning, path tracking control algorithm and position estimation. We adopt an IBM-PC/386 as the main control computer in LCAR control system. Four DC servo motors are used to drive two wheels and the camera mounting device. The mobile robot is equipped with internal sensors (encoders and tachometers), external sensors (a camera, a laser structured light and ultrasonic sensors) and DSP board. The monocular camera is used for recognizing landmarks which are located at the specified positions and 26 ultrasonic sensors attached at front face of mobile robot are used for detecting the obstacles around the mobile robot.

3. Design of the Navigation System

3.1. Position estimation system

In constructing the control system for mobile robots, accurate determination of it location is very important for autonomous navigation. For an absolute position sensor of LCAR, a visual sensor is used for recognizing landmarks with a special pattern pre-located at some known positions around the path. Assuming that the robot moves on the planar surface, the robot position and heading angle are required to be represented in a world coordinate frame.

The location determination problem is to find the relationship between the image deformation and camera location relative to the landmark. Although many researchers have studied on the design of landmark patterns providing an effective solution to the location determination problem[4], most of them are based on analytic methods in deriving estimation algorithms and thus involve constraints in designing the landmark pattern. Furthermore, uncertainties existing in the intrinsic and extrinsic parameters of the camera system degrade the performance of the analytic estimation algorithms. To rectify this problem, we have presented an alternative approach based on a neural network model for camera calibration in the recent work[5]. This method involves a multi-layered neural network used for constructing the uncertain camera optics as well as interpolating the nonlinear functional relationship between the image distortion and the camera location. Fig. 3 shows a 3-D rectangular pattern used for the landmark in this study. Through a binary image processing, four vertex points of front rectangular pattern and a center point of the rear bar pattern are extracted as the image features and then enter the neural network through the ten input nodes. The network has three output nodes corresponding to the camera location(2-D) and orientation relative to the landmark.

For training of the neural network, a training data set are collected from total 160 images measurements, which are performed at various locations and orientations around the landmark. The image feature vector measured at a measuring point and the real posture of the robot are concatenated to compose a set of training patterns. The training is performed off-line by using a well-known error back-propagation method[6]. During the training process, the relative weights of the connections between nodes are adjusted to reduce differences between teaching signals and network's output.

3.2. Neural net-based dynamic control

To improve the tracking performance in controlling two wheel velocities, we have proposed an intelligent control approach in which a feedforward controller is designed based on the neural network dynamic model trained from past experiences[7]. Theoretically, a feedforward controller based on inverse model of the plant can improve the control performance in tracking of servo problem. In practice, it is extremely difficult to accurately obtain the inverse dynamic model. On the contrary, the neural network-based approach has merits in that it does not require the complex dynamic modelling of the plant and has the self-learning ability adaptable to uncertain environment. Fig. 4 shows the block diagram of the proposed control structure. In which we adopt a three layered parallel structure. The application of the neural network to control problem requires the on-line training technique because of the importance of prompt adaptability to environmental changes. We adopted the feedback error learning method introduced by Kawato, et al.[8], which uses the output of the feedback controller for the training signal.

3.3. Local path planning using fuzzy logic and reinforcement learning

Whenever the mobile robot navigates in uncertain environment towards the goal position, two behaviors such as avoidance behavior and goal-seeking behaviors always conflict. The avoidance behavior is used to avoid the obstacles irrespective of the goal position, while the goal-seeking behavior is used to seek the goal position irrespective of obstacle location. Two behaviors are independently designed by fuzzy logic and reinforcement learning and then combined by the behavior selector according to situations around the mobile robot[9]. The fuzzy logic are used to represent the mapping between sensor input space and mobile robot action space consisting of steering angle and linear velocity of mobile robot. Correct mapping from input state space to output action space is gradually accomplished by the reinforcement learning system.

When the control action signals are applied to mobile robot at time step, t, the mobile robot may collide with obstacles or move away from goal position. To avoid such failures in case when the mobile robot navigates again through the environment similar to the previously experienced one, the rules influencing the mobile robot face with these failures must be changed into the correct rules and this task is accomplished by two adaptive neuronlike elements consisting of associative search element(ASE) and associative critic element(ACE) shown in Fig.5[9][10]. The ACE receives the reinforcement signal, r_m^k externally feedbacked from failure detector and generates the internal reinforcement signal, $\hat{r}_m^k$. The weights of the ACE, υ_{mn}^k are learned through the trace of the fired rules and its output, $\hat{r}_m^k$. The subscript m denotes the control action types, that is, linear velocity(m=1) and incremental steering angle(m=2). The subscript n denotes the nth weight strength of the ACE and superscript k denotes the behavior, that is, avoidance(k=1) and goal-seeking(k=2) behaviors. The weight strengths of the ASE, ω_{mn}^k are learned through internal reinforcement signal and eligibility trace of the fuzzy rules. Eventually, if the rules for each behavior are sufficiently learned in the specific uncertain environments, the weights of ASE will be converged into the fixed values.

The membership functions of the input linguistic values are completely known before learning. On the other hand, the membership functions of the output linguistic values are partially known only in their shape, but their center values are unknown. Thus, their center value, f_{mn}^k is determined at each time step by the nth weight of the ASE associated with mth control action type.

4. Results of Simulations and Experiments

4.1. Position estimation

To investigate estimation accuracy of the proposed method utilizing the 3-D landmark, we conducted a series of experiments with the real mobile robot system. Fig.6 shows the result of the visual tracking experiments performed to demonstrate the feasibility of the proposed estimation system via real-time navigation, combined with a path tracking controller. The path tracking controller is based on the time-optimal bang-bang controller with a landing curve design developed for the smooth tracking motion[11]. The tracking experiments were conducted involving lane-changing motions with speed of 0.1m/sec. Starting from the position laterally distanced by 1.2m from the mark axis, the robot makes the lane-changing motion to a straight-line path planned along the mark axis by the tracking control algorithm. In the figure, the solid straight line indicates the final desired path, while the dotted curve indicates the intermediate trajectory of the robot position estimated by the dead reckoning(DR) method. To show effectiveness of the absolute position estimation, a bumper causing slippage is placed on the way of the robot. As seen from the figure, the robot tracked the desired path with negligible error using the absolute position estimation method, whereas the dead reckoning method shows a large estimation error in such uncertain situation.

4.2. Wheel control simulation

To investigate the control performance of the neural network-based wheel controller, a series of tracking simulations were performed for the dynamic model of the mobile robot. In the simulation studies, the neural network was so trained that the mobile robot follows the specific circular path with the given velocity profiles. After learning, the robot is expected a similar tracking performance for a strange path never experienced before. Fig.7 shows a good example in which the path's curvature abruptly changes in negative direction at the midpoint of the path. Therefore, it remains an important further work to find out a method of accumulating past experiences and deriving the generalities from similar ones.

4.3. Local path planning

A series of simulation were made for an arbitrary constructed environment composed of sixteen obstacles which was shown in Fig.1. In order to acquire the environmental information, sensor simulator which outputs the range data of ultrasonic sensors was used. In simulation, the rule bases for avoidance and goal- seeking behaviors consist of 486 and 56 rules, respectively. In the beginning, all the center values of the membership functions of the output fuzzy sets representing the linear velocity and the incremental steering angle are set to constant values. Let us consider the case of learning the avoidance behavior using reinforcement learning method. As shown in the upper part of Fig.8, the mobile robot frequently collides with obstacles at the position marked as C. But, the number of collision begins to decrease as the number of learning steps increases as shown in the lower part of the figure. The rule base for goal-seeking behavior is learned by the same procedure as rule base for avoidance behavior is learned.

Once the rule bases for two behaviors are completely built through reinforcement learning, combination of these behaviors is required to make the mobile robot accomplish its goal, that is, arriving at the given goal position without colliding with obstacles. When the mobile robot is located in a situation, one of these behavior is selected by behavior selector[9][12] acting based on repulsive and attractive potentials. As can be seen from Fig.9, even if the mobile robot faces with situation where the local minimum occurs, the proposed navigation method shows an ability to escape from the local minimum.

5. Conclusions

We have presented a neural network-based navigation system for mobile robots for autonomous navigation in indoor environments. The environments considered herein were locally known where obstacles were located on a planned path. For obstacle detection and avoidance we have designed a sensor-based local path planner that utilizes fuzzy logic and reinforcement learning and ultrasonic sensor signal. In addition, for determination of the position and heading angle of a mobile robots and path tracking control we have used a vision system and perceptron neural networks. The integration of these designed subsystems illustrates applicability of this system and demonstrates the feasibility of autonomous navigation.

References

[1] J.H. Lee and Z.N. Bien, "Collision-free trajectory control for multiple robots based on neural optimization network", Robotica, Vol.8, pp.185-194, 1990.

[2] M. Taniguchi and M. Lang, "Comparison of different neural approximation approaches in the path tracking problem", Proc. Int. Joint Conf. Neural Network, pp.1743-1746, 1993.

[3] P.Deplanques, C. Novales, R. Zapata and B. Jouvencel, "Sensor-based control versus neural network technology for the control of fast mobile robots behaviours in unstructured environments," Proc. Int. Conf. Industrial Electronics, Control, Instrumentation and Automation, U.S.A., pp.694-699, 1992.

[4] J. H. Kim and H. S. Cho, "Real-time determination of a mobile robot's position by linear scanning of a landmark", Robotica, Vol. 10, pp.309-319, 1992.

[5] J. S. Kim, K. C. Koh and H.S. Cho, " A neural net-based position estimation method for mobile robots in indoor environments", Proc. Int'l Symp. on Meas. and Con. in Robotics, ISMCR '92,

pp.899-906, 1992.
[6] D.E. Rumelhart, J. L. MoClelland and the Topresearch group, "Parallel distributed processing", Broadford Books, Vol.182 MIT Press, 1986.
[7] K.C. Koh and H. S. Cho, "A neural net-nased feedforward control scheme for mobile robots", Proc. IEEE IROS '91, pp.813-817, 1991.
[8] M. Kawato, Y. Uno, M. isobe and R. Suzuki, "Hierachical neural networks for voluntary movement with application to robotics", IEEE Control System Magazine, 8-15, April, 1988.
[9] H.R. Beom and H.S. Cho, "A sensor-based navigation for a mobile vobot using fuzzy logic and reinforcement learning", submitted to IEEE Trans. Syst. Man Cybern.

[10] A.G. Barto and R.S. sultton, "Nueronlike adaptive elements that can solve difficult learning control problems," IEEE Trans. Syst. Man Cybern., Vol.SMC-13, No.5, pp.834-846. Sept/oct. 1983.
[11] K. C. Koh and H. S. Cho. "A steering control method for wheel-driven mobile robot", KACC '91, pp.787-792, 1991.
[12] H.R. Beom and H.S. Cho, "Behavioral control in mobile robot navigation using fuzzy decision making approach", submitted to IROS '94.

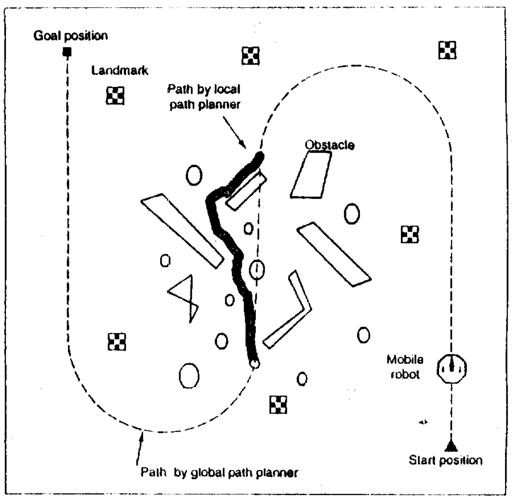

Fig.1 A typical navigation space.

Fig.2 The wheeled mobile robot, LCAR.

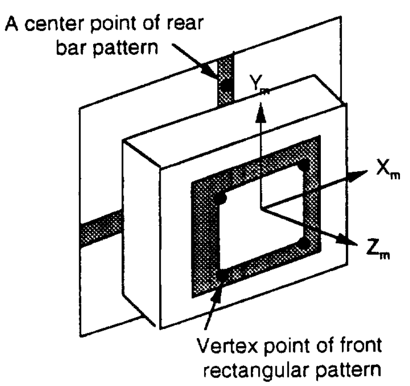

Fig.3 The proposed 3-D landmark.

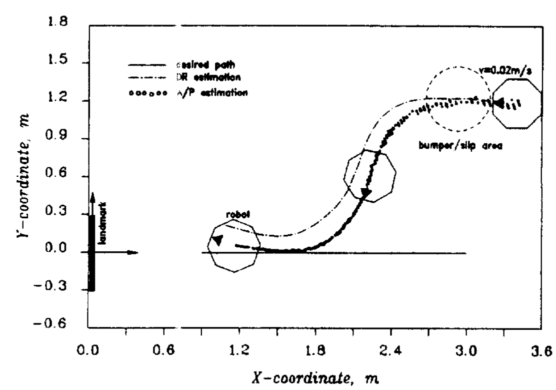

Fig.6 The result of tracking control using A/P estimation.

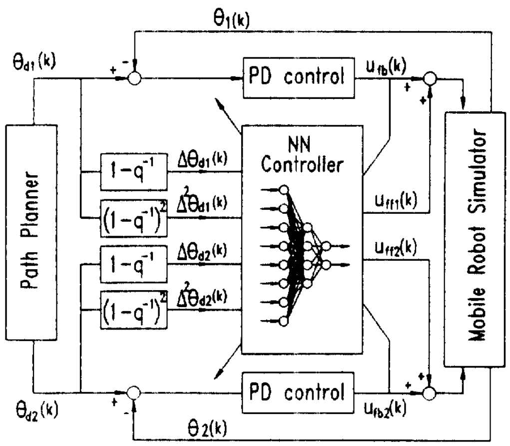

Fig.4 The NN-based control scheme.

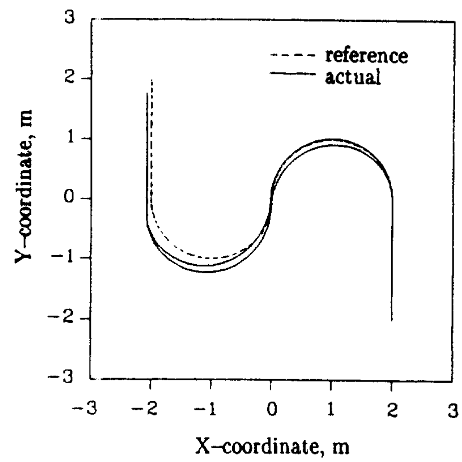

Fig.7 The tracking results for a S-curve path.

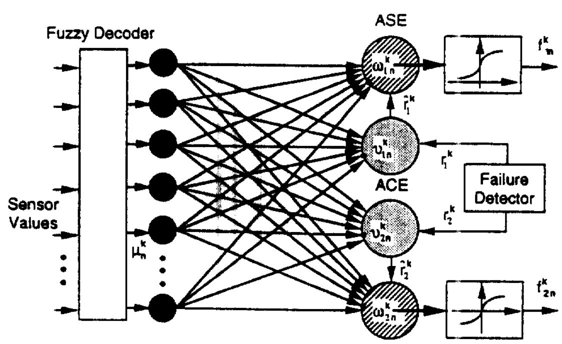

Fig.5 The structure of the neural network for learning the k-th behavior.

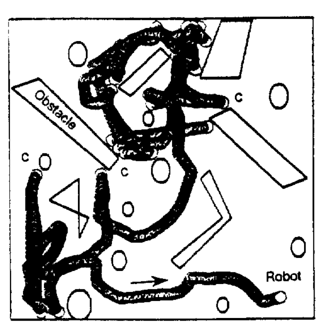

Fig.8 The trajectory of the mobile robot in learning process.

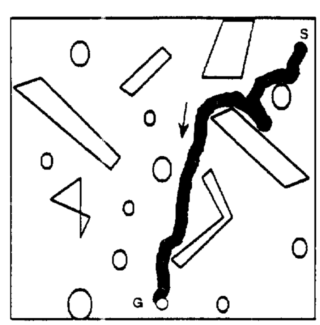

Fig.9 The trajectory by local path planner.