End-to-End Deep Learning for Autonomous Navigation of Mobile Robot

Ye-Hoon Kim
Samsung Electronics
Seoul R&D Campus
Seongchon-gil, Seocho-gu
Seoul, Republic of Korea
Email: yehoon.kim@samsung.com

Jun-Ik Jang
Samsung Electronics
Seoul R&D Campus
Seongchon-gil, Seocho-gu
Seoul, Republic of Korea
Email: ji.jang@samsung.com

Sojung Yun
Samsung Electronics
Seoul R&D Campus
Seongchon-gil, Seocho-gu
Seoul, Republic of Korea
Email: sojung15.yun@samsung.com

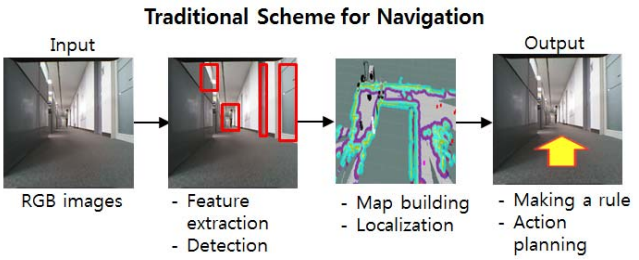
***Abstract*—This paper proposes an end-to-end method for training convolutional neural networks for autonomous navigation of a mobile robot. Traditional approach for robot navigation consists of three steps. The first step is extracting visual features from the scene using the camera input. The second step is to figure out the current position by using a classifier on the extracted visual features. The last step is making a rule for moving the direction manually or training a model to handle the direction. In contrast to the traditional multi-step method, the proposed visuo-motor navigation system can directly output the linear and angular velocities of the robot from an input image in a single step. The trained model gives wheel velocities for navigation as outputs in real-time making it possible to be implanted on mobile robots such as robotic vacuum cleaner. The experimental results show an average linear velocity error of 2.2 cm/s and average angular velocity error of 3.03 degree/s. The robot deployed with the proposed model can navigate in a real-world environment by only using the camera without relying on any other sensors such as LiDAR, Radar, IR, GPS, IMU.**

I. Introduction

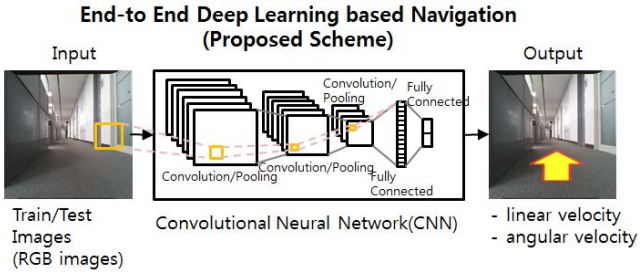
In the past years, tremendous progress has been made on autonomous navigation technologies for robot, drone and vehicle. However, it has been difficult to navigate using only vision-based approaches even though visual localization methods such as visual simultaneous localization and mapping (SLAM) have been developed. Most of the methods applied on commercial products utilizes diverse sensors such as IR, radar, ultrasonic along with the camera. While adding sensors have benefit of easy perception, they are accompanied by additional cost and power-consumption. Also, sensors sometimes fail to detect unexpected obstacles. For example, one dimensional IR sensors which are typically used in commercial mobile robots have problems with detecting complex structures. Besides, when applied on autonomous vehicles, these sensors are unreliable in bad weather conditions. For example, LiDAR scanner, which autonomous vehicles heavily rely on, will not work on rainy days due to the reflectance of rain drops. Systems based on vision may be more reliable even on unexpected situations. Even when other sensors are present, vision-based system can be used to improve the existing system and increase its robustness.

In most implementations of vision-based mobile robot, the process is multi-step as shown in Fig. 1a. First, the visual features are extracted from the camera input for detecting bounding boxes or for segmentation [1]-[3]. Then the current position or situation needs to be identified with respect to the environment. This is achieved mostly by using visual-SLAM technique [4]-[6]. Lastly, based on this situation, the final driving action is determined by using classifier or by incorporating a set of predetermined rules for possible scenarios [7]-[9]. When using such a multi-step approach, each step needs to be redesigned or retrained every time the situation or environment of the robot changes. Also, each step can have error and these errors are accumulated during the entire process. The performance of the last module such as action planning is dependent on the performance of earlier module such as perception module.

In this paper, we propose a single-step, end-to-end approach for mobile robot navigation, to directly infer the final velocity from the visual input as shown as in Fig. 1b. This reduces the



(a) Navigation method using traditional method.



(b) Navigation procedure using proposed method.

Fig. 1. Comparison between traditional perception and planning method, and proposed end-to-end approach.

978-1-5386-3025-9/18/$31.00 ©2018 IEEE

complexity of the implementation and utilizes only a camera thereby reducing the cost, power and computational time. We use the state-of-the-art framework of deep convolutional neural network (CNN), which takes image as an input and produces the linear and angular speeds of motion for the mobile robot. Training and testing the robot in an office environment under various situations, our vision-based system is more robust than robots which use other sensors for navigation. In the experimental results, our approach provides a novel, simple and effective vision-based method for local path planning. The proposed autonomous navigation system can be used in various applications including robotic vacuum cleaner, mobile social robot at home, service robot at building and autonomous vehicle.

II. Related Works

Traditionally, robot navigation and action planning had been controlled in a multi-step process. The robot would first detect obstacles, draw a map, localize the robot, and plan the robot's local path according to environmental structure. Nonetheless, vision-based end-to-end methods have been introduced showing comparable accuracy in robot navigation with the sensor-based multi-step methods. The first obstacle avoidance method based on end-to-end CNN was applied on a robot, DAVE [10]. It could directly learn the driver's steering angles from input images of the robot's point of view. The main difference between navigation system of DAVE and ours is type of output. The CNN trained for DAVE performed binary classification with the two outputs as 'turn left' and 'turn right' commands. Our system performed regression having linear and angular velocities of the robot as outputs. Continuous velocity outputs can control the robot more elaborately. The robot applied with our system can carry out simple obstacle avoidance and also perform complex tasks such as 'stopping in front of cliff' or 'pass through the door after it is opened'.

Recently, several end-to-end learning methods have been presented to solve efficient navigation problem in the unpredictable environment. Robot takes LiDAR information with target position as input to perform path planning by end-to-end learning algorithm [11]. However LiDAR sensors have limitations such as limited sensor range and impossibility to detect transparent materials. Also, robot navigates by taking camera input image at the current location and the image taken at the target area as inputs for end-to-end reinforcement learning [12]. However this method has practical limitation which can be used only in a trained environment.

Meanwhile, end-to-end approaches using neural networks on learning visuomotor policies has been presented on other applications including robotic arm control and imitation learning [13]-[15]. They presented methods for learning robot control policies directly from visual inputs. One system could control mechanical actions of robots by observing the hand of the robot itself [13], [14]. The other system could let the robot learn and imitate human actions obtained from the robot camera [15].

Most techniques on autonomous driving vehicles are multi-step processes, and involve multiple sensors along with the camera. Many implementations detect or track the moving cars [16], [17], or the lane markers on the road [18], [19]. However, there exists unnecessary detections that are not directly used in deciding the final driving direction. DeepDriving [20] proposes a direct perception approach in autonomous driving. However, perception and action planning are still separated. Recently, an end-to-end CNN navigation system on autonomous vehicle has been proposed and has been tested on real roads [21].

III. Proposed End-to-end Deep Learning System for Navigation

A. Navigation System using Deep CNN

The proposed system architecture is shown in Fig. 1b. The input of the proposed architecture is an RGB image and the outputs are linear and angular velocities. The system does not require detection, localization or planning module for navigation separately. The CNN architecture used in this paper is AlexNet [22]. Even though other well-known architectures such as VGGNet [23], GoogleNet [24] or ResNet [25] can be used, these networks are not applicable for real-time robot navigation due to slow inference speed. Our network performs multi-label regression giving outputs as two real-values. The ground truth velocities are in the range of 0 to 0.5m/s for linear velocity, and in the range of -1.5 to 1.5 radians for angular velocity. Two velocities were normalized to the values between 0 and 1 for CNN input. Since the output value are oscillated, using raw output values make the robot movement to be unstable. For acquiring consistent outputs, post-processing for noise reduction was conducted to make the movement of the robot stable.

Raw RGB frames collected from a camera at a speed of 30 fps were used as the input of the network, and no further image pre-processing was performed. A total of 7,202 images were used for training and 2,332 images were used for testing. The train data and the test data were recorded navigating the same area, but the obstacles such as human or objects were different between the two sets. Data was collected by manually operating the robot through joystick control. Both image and velocity data were collected simultaneously. We performed experiments on pre-recorded videos with ground-truth velocity values for accuracy measurement. Moreover, the system was applied on a mobile robot for real-time navigation.

B. System Design Implementation

For implementation and experiment, personal robot kit with open-source software ROS similar to robotic vacuum cleaner is used. Fig. 2 illustrates various modules of our implementation, and their functions and interactions. A laptop which acts as the robot server communicates with the robot via the wireless network. This robot server acts as the master node of ROS. The robot server is connected to the deep learning (DL) server which is equipped with the GPU to run the CNN based on Caffe framework. The image is captured by the camera on the robot and then sent to the robot server. The input image size is

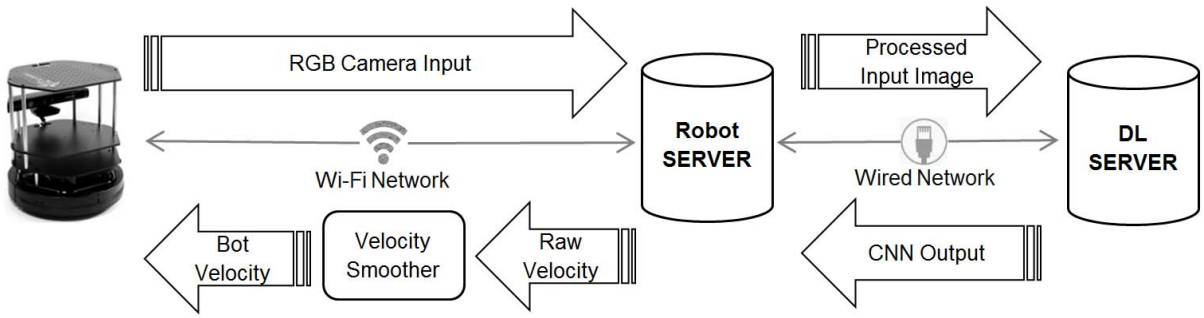


Fig. 2. Visuo-motor system configuration of modules and functions for mobile robot

scaled down to 224x224 to act as the input for the CNN. This image is sent to the DL server using a cURL-based script. The output of the CNN is again fetched from the DL server and the outputs are scaled to produce the final linear and angular velocities for the robot.

At a regular interval of 0.5s, the robot server publishes the most recent velocity to the robot. To prevent rapid changes in velocities due to delays between these intervals, a velocity smoother module is applied.

IV. Experimental Environment and Scenarios

In this section, scenarios of local path planning for mobile robot navigation in indoor environment are presented. These scenarios are designed to verify that the robot can avoid structures including walls, static obstacles such as objects, dynamic obstacles such as human in normal situations. Moreover, experiments avoiding extreme situations such as falling down the stairs or confronting a closed glass door were accomplished.

A. Train and Test Scenarios for Normal Situation

First of all, the following scenarios aim to verify normal obstacle avoidance capability by distinguishing the situations for stopping or detouring.

- Go straight when no obstacles are present at the corridor.
- Stop when dynamic obstacles appear.
- Detour when static obstacle appears.
- Detour when dynamic and static obstacles appear simultaneously.
- Turn around the corner.

The robot should stop when it encounters a human because the human can move quickly at any time. However, in cases when static objects such as chair appear the robot can detour. In a complex situation when human and chair appear at the same time, the robot should decide the action based on scene understanding. Researches on scene parsing or understanding techniques based on end-to-end deep learning have been presented [26], [27]. As a car should detour its direction on a road construction situation in the real world, the robot can understand the scene and detour if the robot interprets that the human in the front is doing some work with the object. Moreover, both trained and untrained obstacles were tested to verify the robustness of proposed visuo-motor system. For example, although only certain types of chairs with armrest and headrest were trained, the robot could avoid different types of chairs on the testing phase.

B. Train and Test Scenarios for Extreme Situation

On normal scenarios, the proposed visuo-motor system can perform as similar as the robot navigation system using sensory fusion. However, sensor-fusion based system occasionally fails navigating due to limitations of sensors in extreme cases. For instance, laser or IR sensor cannot detect a wall which is very far, glassy objects or a cliff. The following scenarios are designed to navigate on extreme situations using the proposed system.

- Escape stuck situation by rotating and checking for wide spaces - Laser or IR sensors cannot detect the objects located in a far distance due to short sensor ranges.
- Stop before going down the stairs to avoid falling - One dimensional laser or IR sensors cannot detect cliffs.
- Stop in front of the closed glass door and pass through the opened glass door - Laser or sensors cannot detect transparent objects.

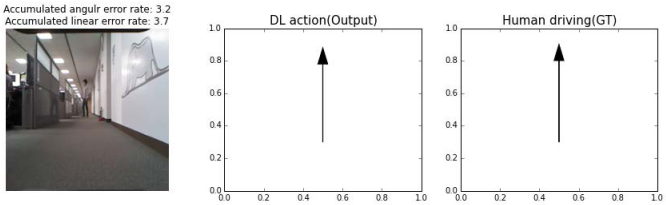
V. Experimental Results

The trained visuo-motor system based on deep learning was tested on pre-recorded video for measuring the accuracy. Also, the proposed system was applied to mobile robot in a real environment to verify if it is possible to automatically control itself in real-time. A demo video clip of the experimental results was uploaded[1].

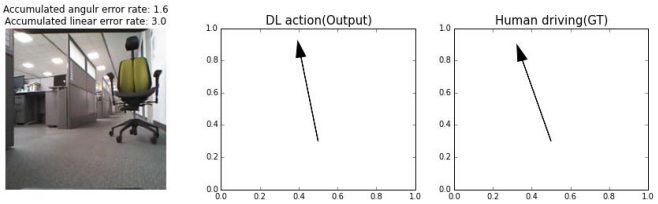
A. Results on pre-recorded Video

Test results on pre-captured videos and the heading angles converted from linear and angular velocities are expressed by an arrow in Fig. 3-7. The length of the arrow indicates magnitude of the velocity. Examples of the test cases in normal situations are shown in Fig. 3, 4. In Fig. 3, results show that the robot can avoid trained obstacles. Moreover, in Fig. 4 experimental results show that the robot can also efficiently avoid untrained obstacles. For instance, the green chair in Fig. 3b was included in the train set, but the black chair

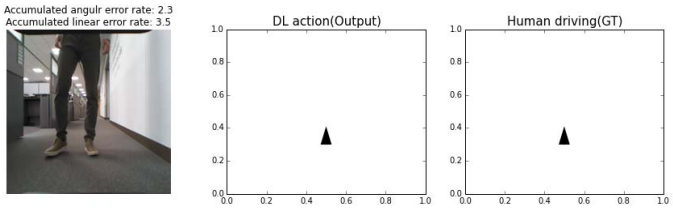
[1]Demo video: https://youtu.be/xukJGeHwKf4



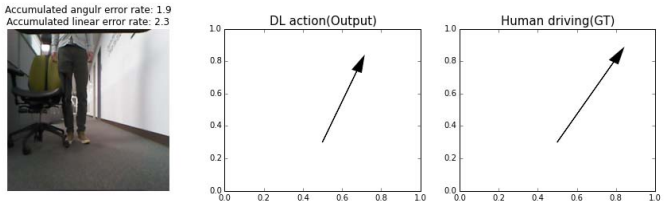
(a) Going straight.



(b) Avoiding trained and static obstacle.



(c) Stopping in front of the trained and dynamic human.



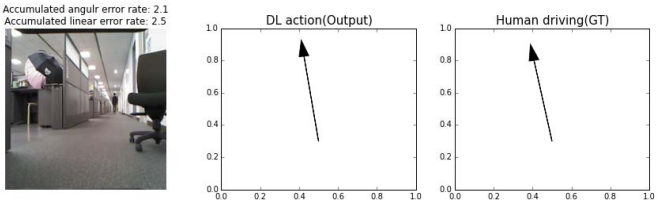
(d) Avoiding trained human and chair (complex scene).

Fig. 3. Test cases on trained obstacles.

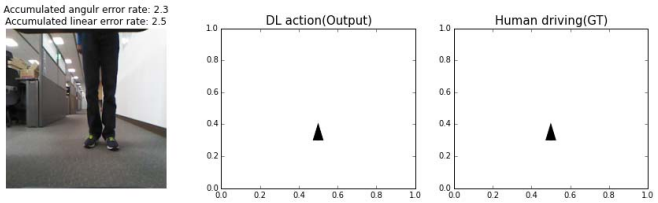
in Fig. 4a was not trained. Still, the mobile robot was able to avoid both the two chairs. Similarly, the human in Fig. 4b, the human and the chair in Fig. 4c and phone booth in Fig. 4d were not trained but the resulting heading angle was appropriate for driving compared to ground truth (GT). Results in the untrained cases (Fig. 4) show the proposed visuo-motor system's robustness. The average error of linear and angular velocity outputs compared to the GT velocities obtained from human by joystick control for the scenario of 'navigation in office' were 1.4 cm/s and 4.12 degree/s, respectively.

One of the feature map of second convolution layer is described in Fig. 5. The trajectories calculated by velocity and head angle about relative location of robot are described in 6. The trajectory tendency (red line) of deep learning system itself was similar to human driving trajectory (green line).

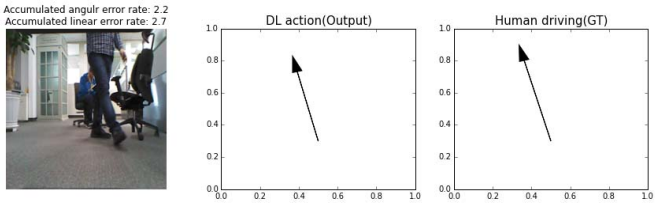
Examples of results on extreme scenarios are shown in Fig. 7. First, in Fig. 7a there is no visual object in an wide open space around the robot. In this case, the robot following the traditional methods using short-range LiDAR or IR sensors



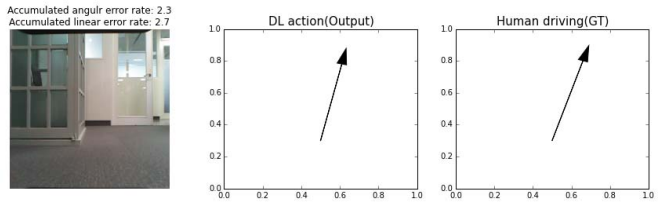
(a) Avoiding untrained and static obstacle.



(b) Stopping in front of untrained and dynamic human.



(c) Avoiding untrained and multiple humans and chairs.



(d) Avoiding untrained and new type of obstacle.

Fig. 4. Test cases on untrained obstacles.

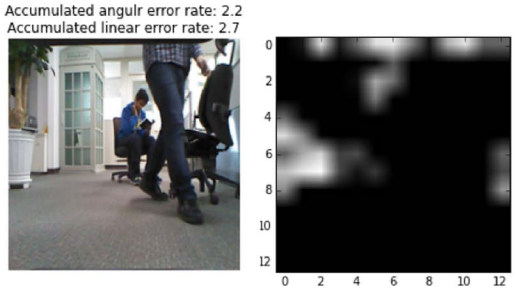


Fig. 5. An example of feature map (right) of second convolutional layer.

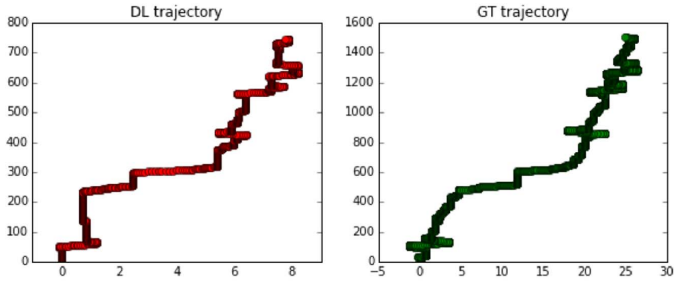


Fig. 6. Trajectories of proposed system and human manual driving.

is expected to fail to localize. Using traditional methods, if the robot fails finding nearby obstacles it cannot localize and move. However the result in Fig. 7a shows that the robot applied with the proposed system can deal with this situation. Second, the case in Fig. 7b shows the robot safely stopped in front of the stairs without falling down. The last cases in Fig. 7c and 7d show that the robot could stop in front of the closed glass door and pass through the glass door only if the door is opened. Sensors often fail to detect transparent objects. In extreme scenarios, the average error of linear and angular velocities were 3.1 cm/s and 1.89 degree/s, respectively. As a result, the total error on both normal and extreme scenarios of linear and angular velocities were 2.2 cm/s and 3.03 degree/s, respectively.

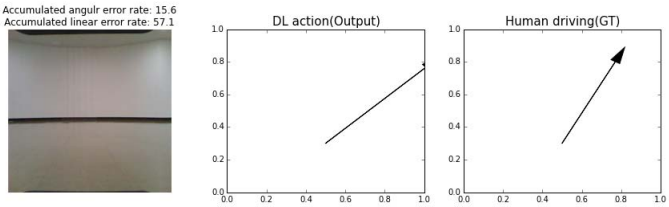
B. Results on Navigation of Real Mobile Robot

The system was configured as Fig. 2 for the verification of visuo-motor system in a real-world environment. The input image of robot is sent to an external deep learning server to infer robot velocity outputs. Since input image from robot camera is affected by robot action such as heading angle or wheel velocities, wrong output velocities can lead to a series of unexpected input images causing wrong movements. Therefore, large accumulated errors can be fatal to robot navigation. Thus the movement of the robot with the proposed visuo-motor system should be robust to small errors of output velocities on individual frames. As shown in Fig. 8, local path planning for obstacle avoidance of the mobile robot in real-time test lead the robot drive without collision.

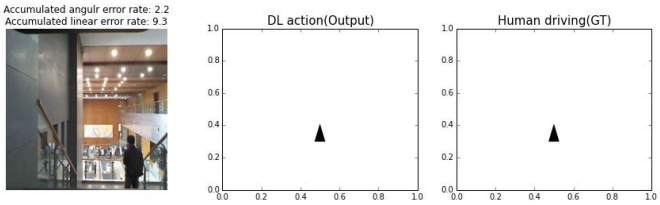
VI. Conclusions

Traditional methods for robot navigation or path planning requires multiple and complex algorithms for localization, navigation and action planning. The proposed approach using end-to-end deep learning could make it possible to control the robot motor directly from the visual input as the human did. The human can decide the path seeing only a local scene without any information of the global map. This result verified the potential of the proposed system as a local path planner.

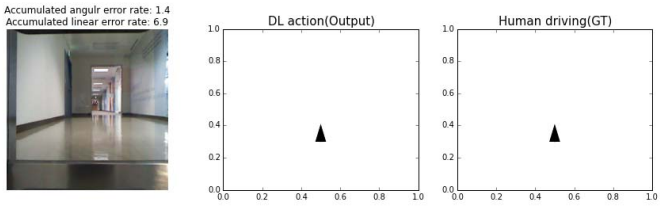
For future work, the visuo-motor system as a global path planner can be developed. Moreover, the model can be compressed for direct deployment of the visuo-motor system on the embedded board without server.



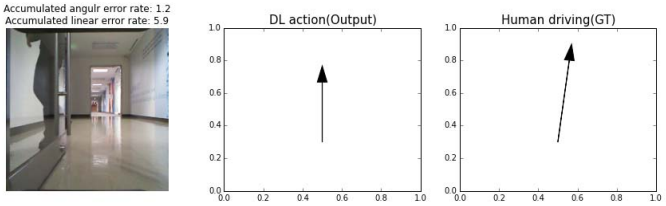
(a) Escape wide open space.



(b) Stop to avoiding fall down.



(c) Stop in front of the closed glass door.



(d) Pass through the opend glass door.

Fig. 7. Extreme test cases.

References

[1] A. Vale, J. M. Lucas and M. I. Ribeiro, "Feature extraction and selection for mobile robot navigation in unstructured environments," in IFAC Proceedings, vol. 37, issue 8, pp. 102-107, Jul. 2004.

[2] S. Zhang, L. Xie and M. D. Adams "Feature extraction for outdoor mobile robot navigation based on a modified Gauss Newton optimization approach," in Robots and Autonomous Systems, Elsevier, vol. 54, issue 4, pp. 277-287, Apr. 2006.

[3] M. A H Ali, M. Mailah, W. A. B. Yussof, Z. B. Hamedon, Z. B Yussof and A. P P Majeed, "Sensors Fusion based Online Mapping and Features Extraction of Mobile Robot in the Road Following and Roundabout," in IOP Conference Series: Materials Science and Engineering, vol. 114, conference 1, 2016.

[4] W.-Y. Jeong and K.-M. Lee, "Visual SLAM with Line and Corner Features," in IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2570-2575, Beijing, Oct. 2006.

[5] C. Roussillon, A. Gonzalez, J. Sola, J.-M. Codol, N. Mansard, S. Lacroix, and M. Devy, "RT-SLAM: a generic and real-time visual SLAM implementation," in Computer Vision Systems of the series Lecture Notes in Computer Science, vol. 6962, pp. 31-40, 2012.

[6] Y. J. Shih, C. C. Hsu, W. Y. Wang and Y. T. Wang, "Feature extracted algorithm for simultaneous localization and mapping (SLAM)," ICCE, 2015.

[7] T. Belke and D. Schulz, "Local action planning for mobile robot collision

Fig. 8. The mobile robot avoids obstacle by communicating between deep learning server and motor controller in real-time.

avoidance," in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002.

[8] O. Hachour, "Path Planning of Autonomous Mobile Robot," in International Journal of Systems Applications, Engineering and Development, Issue 4, Volume 2, 2008.

[9] M. Sridharan and P. Stone, "Comparing Two Action Planning Approaches for Color Learning on a Mobile Robot," In VISAPP International Workshop on Robotic Perception (VISAPP-RoboPerc08), Funchal, Portugal, Jan. 2008.

[10] U. Muller, J. Ben, E. Cosatto, B. Flepp, and Y. L. Cun "Offroad obstacle avoidance through end-to-end learning," in Advances in Neural Information Processing Systems, pp. 739-746, 2005.

[11] M. Pfeiffer, M. Schaeuble, J. Nieto, R. Siegwart and C. Cadena, "From perception to decision: A data-driven approach to end-to-end motion planning for autonomous ground robots," in IEEE International Conference on Robotics and Automation (ICRA), 2017.

[12] Y. Zhu, R. Mottaghi, E. Kolve, J. Lim, A. Gupta, L. Fei-Fei and A. Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning," in IEEE International Conference on Robotics and Automation (ICRA), 2017.

[13] S. Levine, C. Finn, T. Darrell, and P. Abbeel "End-to-End Training of Deep Visuomotor Policies," in Journal of Machine Learning Research 17, 2016.

[14] S. Levine, P. Pastor, A. Krizhevsky and D. Quillen "Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection," In The 33rd International Conference on Machine Learning (ICML), 2016.

[15] P. Sermanet, C. Lynch, J. Hsu and S. Levine, "Time-Contrastive Networks: Self-Supervised Learning from Multi-View Observation," in arXiv:1704.06888, Apr. 2017.

[16] Y.-K. Lai, Y.-H. Huang and C.-M. Hwang, "Front moving object detection for car collision avoidance applications," in IEEE International Conference on Consumer Electronics (ICCE), 2016.

[17] S.-G. Kim, J.-E. Kim, K. Yi and K.-H. Jung, "Detection and tracking of overtaking vehicle in blind spot area at night time," in IEEE International Conference on Consumer Electronics (ICCE), 2017.

[18] D. Ding, J. Yoo, J. Jung, S. Jin and S. Kwon, "Various lane marking detection and classification for vision-based navigation system," in IEEE International Conference on Consumer Electronics (ICCE), 2015.

[19] B. Huval, T. Wang, S. Tandon, J. Kiske, W. Song, J. Pazhayampallil, M. Andriluka, P. Rajpurkar, T. Migimatsu, R. C.-Yue, F. Mujica, A. Coates and A. Y. Ng, "An Empirical Evaluation of Deep Learning on Highway Driving," in Computing Research Repository (CoRR), 2015.

[20] C. Chen, A. Seff, A. Kornhauser, and J. Xiao "DeepDriving: Learning Affordance for Direct Perception in Autonomous Driving," In Proceedings of 15th IEEE International Conference on Computer Vision (ICCV), 2015.

[21] M. Bojarski, D. D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao and K. Zieba, "End to End Learning for Self-Driving Cars," in arXiv:1604.07316, 2016.

[22] A. Krizhevsky, S. Ilya and G. E. Hinton. "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in International Conference on Learning Representations (ICLR), 2015.

[24] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, and A. Rabinovich "Going deeper with convolutions," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[25] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," in IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[26] W. Huang, X. Gong, "Fusion Based Holistic Road Scene Understanding," arXiv:1406.7525, 2014.

[27] C.-A. Brust, S. Sickert, M. Simon, E. Rodner and J. Denzler, "Convolutional Patch Networks with Spatial Prior for Road Detection and Urban Scene Understanding," VISAPP, 2015.