Received November 27, 2019, accepted January 10, 2020, date of publication January 17, 2020, date of current version January 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2967509

Unexpected Collision Avoidance Driving Strategy Using Deep Reinforcement Learning

MYOUNGHOE KIM[1], SEONGWON LEE[1], JAEHYUN LIM[1], JONGEUN CHOI[1], (Member, IEEE), AND SEONG GU KANG[2]

[1]School of Mechanical Engineering, Yonsei University, Seoul 03722, South Korea
[2]Institute of Science and Technology, Korea University, Sejong 30019, South Korea

Corresponding author: Jongeun Choi (jongeunchoi@yonsei.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government, (Ministry of Science and Information Technology) (2018R1A2B6008063).

ABSTRACT In this paper, we generated intelligent self-driving policies that minimize the injury severity in unexpected traffic signal violation scenarios at an intersection using the deep reinforcement learning. We provided guidance on reward engineering in terms of the multiplicity of objective function. We used a deep deterministic policy gradient method in the simulated environment to train self-driving agents. We designed two agents, one with a single-objective reward function of collision avoidance and the other with a multi-objective reward function of both collision avoidance and goal-approaching. We evaluated their performances by comparing the percentages of collision avoidance and the average injury severity against those of human drivers and an autonomous emergency braking (AEB) system. The percentage of collision avoidance of our agents were 78.89% higher than human drivers and 84.70% higher than the AEB system. The average injury severity score of our agents were only 8.92% of human drivers and 6.25% of the AEB system.

INDEX TERMS Autonomous vehicles, collision avoidance, intelligent vehicles, injury severity, multi-objective optimization, reinforcement learning.

I. INTRODUCTION

A. MOTIVATION

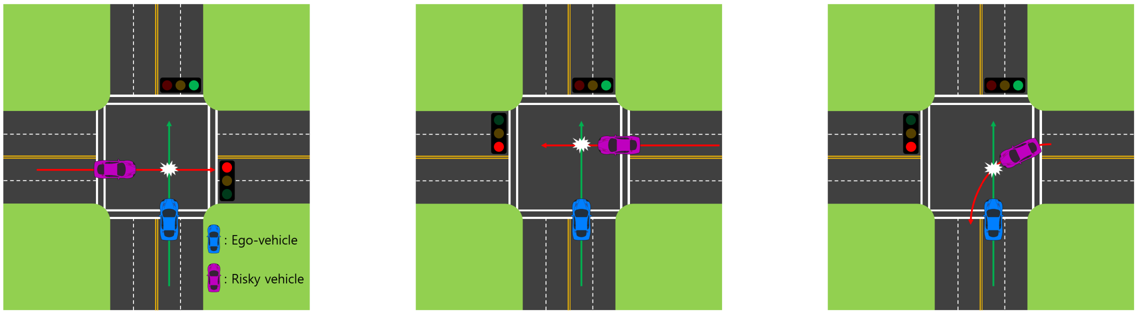
According to the National Highway Traffic Safety Administration (NHTSA), the main cause of 94 percent of the critical pre-crash event is attributed to drivers [1]. Among the driver-related reasons, recognition error and decision error accounts for 41 and 33 percent respectively. These statistics implies that the ability of drivers to recognize the driving situation and to decide the optimal driving control is imperfect. Drivers cannot fully recognize risky situations, since they are not able to consider the frontal, lateral, and rear situations simultaneously due to their physical limitation. Moreover, in risky situations caused by unexpected behaviors of arbitrary vehicle, it is extremely difficult for the driver to precisely recognize the situation and immediately decide to act in a way of avoiding the collision or minimizing the damage on his or her body. These unexpected collision scenarios include, for example, a case where a risky vehicle runs a red light dashing from the lateral direction while the ego-vehicle is straightly crossing the intersection [2].

Motivated by aforementioned needs, we focus on making our vehicle self-drive to avoid the collision or minimize the injury severity in 3 unexpected traffic signal violation scenarios at an intersection, described in Fig. 1. Our suggested scenarios occur about 254,000 times economically costing about 6,627 million dollars annually [2], which can be greatly reduced with a viable solution to avoid them.

B. RELATED WORKS

Many technologies have been investigated for the risk avoidance in driving situations. An Autonomous Emergency Braking (AEB) system [3] is a system that recognizes the driving situation using cameras and a laser scanner and autonomously brakes the vehicle when another vehicle is detected within the range of risk. However, an AEB system cannot take suitable actions other than just braking the vehicle in situations not predefined in advance. In [4], they introduced a cooperative collision avoidance (CCA) scheme for intelligent transport systems presenting a cluster- based organization

The associate editor coordinating the review of this manuscript and approving it for publication was Qiuhua Huang.



(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

FIGURE 1. Visualizations of 3 scenarios considered in this paper. In (a) and (b), the risky vehicle runs a red light from two lateral directions when the ego-vehicle start to go straight up at a green light. In (c), the risky vehicle runs a red light to make a left turn from lateral directions when the ego-vehicle start to go straight up at a green light.

of the target vehicles. In [5], they employed a cooperative autonomous driving system where a vehicle overtakes the one in front based on collective perception. A path planning and tracking framework is presented to maintain a collision-free path for autonomous vehicles in [6]. Various techniques handling the intersection collision avoidance are listed in [7]. Besides, there are still many Advanced Driver Assistance Systems (ADAS) like the pedestrian detection [8], autonomous parking systems and Tesla's Autopilot. In [9], a driving assistant companion system that provides drivers with useful information using an LSTM network is proposed. In [10], they propose a mixed-integer linear program-based urban traffic management scheme for an all connected vehicle environment at an intersection scenario. These studies usually assume a predefined or simplified situation and are not practically applicable in high dimensional and changeable state space like our problem. Therefore, investigations on the collision avoidance strategy considering the injury severity are scant. Our study uses a deep reinforcement learning method to solve unexpected traffic signal violation scenarios, which have never been attempted to be solved with deep reinforcement learning methods.

Reinforcement learning methods have been used in various tasks regarding autonomous driving [11]–[16]. In [11], a deep deterministic policy gradient method [12] was used to make a mapless motion planner taking 10-dimensional range findings and the goal position as the state. In [13], the asynchronous actor-critic method [14] was used to make an end-to-end driving agent using only the RGB image from a forward facing camera. In [15], a deep Q-network [17] was used to make an efficient strategy to navigate safely through unsignaled intersections. In [16], [18], [19], and [20], a deep Q-network and a deep deterministic policy gradient method were used for discrete and continuous actions respectively for the autonomous maneuvering in an open source car simulator.

C. CONTRIBUTIONS

The main contributions of this paper are as follows. First, using the deep reinforcement learning with only Light Detection and Ranging (LIDAR) observations, we generate intelligent self-driving policies that can avoid the collision or minimize the injury severity in unexpected traffic signal violation scenarios at an intersection. Next, we provide guidance on reward engineering in terms of the multiplicity of objective function, i.e., collision avoidance with or without goal-approaching. Finally, we consider the injury severity score to minimize the injury on the driver in case the agent cannot avoid the collision.

As a result our agents show 78.89% higher percentage of collision avoidance than human drivers and 84.70% higher than the AEB system. The average injury severity score of our agents are only 8.92% of human drivers and 6.25% of the AEB.

Only single-channel Light Detection and Ranging (LIDAR) observations were used as sensory inputs to recognize the surrounding situation. This constraint of sensor usage makes our work challenging comparing to the fact that most of the ADAS technologies require multiple types of high-cost sensors (e.g., multi-channel LIDAR, camera, and radar). Experiments in this paper shows that our agents perform much better than human drivers and the AEB system. We design two agents, one with a single-objective reward function and the other with a multi-objective reward function, and compared their performance and driving behavior. For minimizing the injury on the driver, we refer to [21] which studies a model that estimates the injury severity score in the two-vehicle crash using the Newtonian mechanics and generalized linear regression.

The organization of this paper is as follows. In Section II, we introduce backgrounds of reinforcement learning and a deep deterministic policy gradients method. Our problem in this paper is defined and the method to solve it is described. Experimental and comparison studies against human drivers and an AEB system are illustrated in Section III. Finally, results and conclusions are presented in Sections IV and V.

II. APPROACH

A. REINFORCEMENT LEARNING

In the reinforcement learning, an agent observes its state s_t and takes an action a_t decided by the policy π. The agent moves to the next state s_{t+1} getting a reward r_t. A reinforcement learning problem is normally set as a Markov Decision

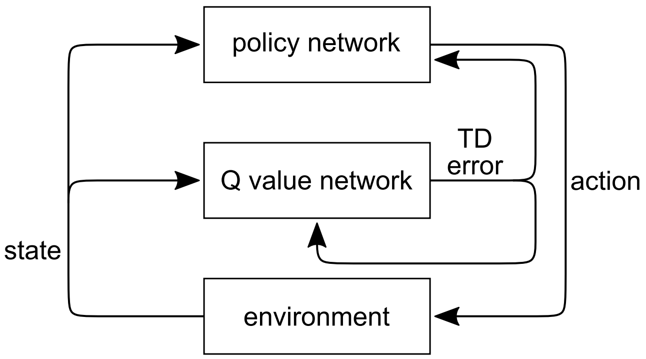


FIGURE 2. The actor-critic architecture. The policy function structure is known as the actor, and the value function structure is referred to as the critic. The actor produces an action given the current state of the environment, and the critic produces a temporal difference (TD) error signal given the state and reward.

Process (MDP) $\langle S, A, P, R, \gamma \rangle$, where S is a finite set of states, A is a finite set of actions, P is a state transition probability matrix, R is a reward function, and γ is a discount factor. MDP assumes the Markov property that the probability of moving to a new state is independent of all states and actions except for the current state and the previous action. The state transition probability defines the transition probability from all states to all successor states, the reward function yields a reward for a given time step, and the discount factor discounts future rewards preventing the total reward from going to infinity. The reinforcement learning agent learns to decide actions that maximize the expected return $\mathbb{E}[R_t]$, defined in the following equation.

$$\mathbb{E}\left[R_t\right] = \mathbb{E}\left[\sum_{k=0}^{T} \gamma^k r_{t+k}\right] \tag{1}$$

We use a deep deterministic policy gradients method for this optimization problem, which will be explained in the following section.

B. DEEP DETERMINISTIC POLICY GRADIENTS METHOD

We chose the deep reinforcement learning method to solve our problem. Recently deep reinforcement learning has been steadily improved. In [17], a deep neural network is used for function estimation of value-based reinforcement learning. This method is applicable only to tasks with discrete action space. To solve a continuous control task such as driving, [12] proposed a deep deterministic policy gradient method, which uses an actor-critic method shown in Fig. 2 to represent policy $\mu(s|\theta^\mu)$ and value $Q(s, a|\theta^Q)$ using deep neural networks. A replay buffer and target networks are introduced to solve the instability of learning caused by using deep neural networks. A replay buffer makes the training sample independently and identically distributed, which makes the algorithm much more data-efficient. The target network makes the parameters change more slowly. The critic network is trained using the following Bellman equation,

$$Q^\mu(s_t, a_t) = \mathbb{E}_{r_t, s_{t+1} \sim E}[r(s_t, a_t) + \gamma Q^\mu(s_{t+1}, \mu(s_{t+1}))], \tag{2}$$

where in the algorithm a random minibatch of N transitions (s_i, a_i, r_i, s_{i+1}) is sampled from a replay buffer and set as (3).

$$y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'}) \tag{3}$$

The following policy gradient is used for updating the actor.

$$\nabla_{\theta^\mu} J = \mathbb{E}_{s_t \sim \rho^\beta}[\nabla_{\theta^\mu} Q(s, a|\theta^Q)|_{s=s_t, a=\mu(s_t|\theta^\mu)}] \tag{4}$$

In the algorithm the sampled policy gradient is calculated from the sampled minibatch as

$$\nabla_{\theta^\mu} J \approx \frac{1}{N} \sum_i \nabla_a Q(s, a|\theta^Q)|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu} \mu(s|\theta^\mu)|_{s_i} \tag{5}$$

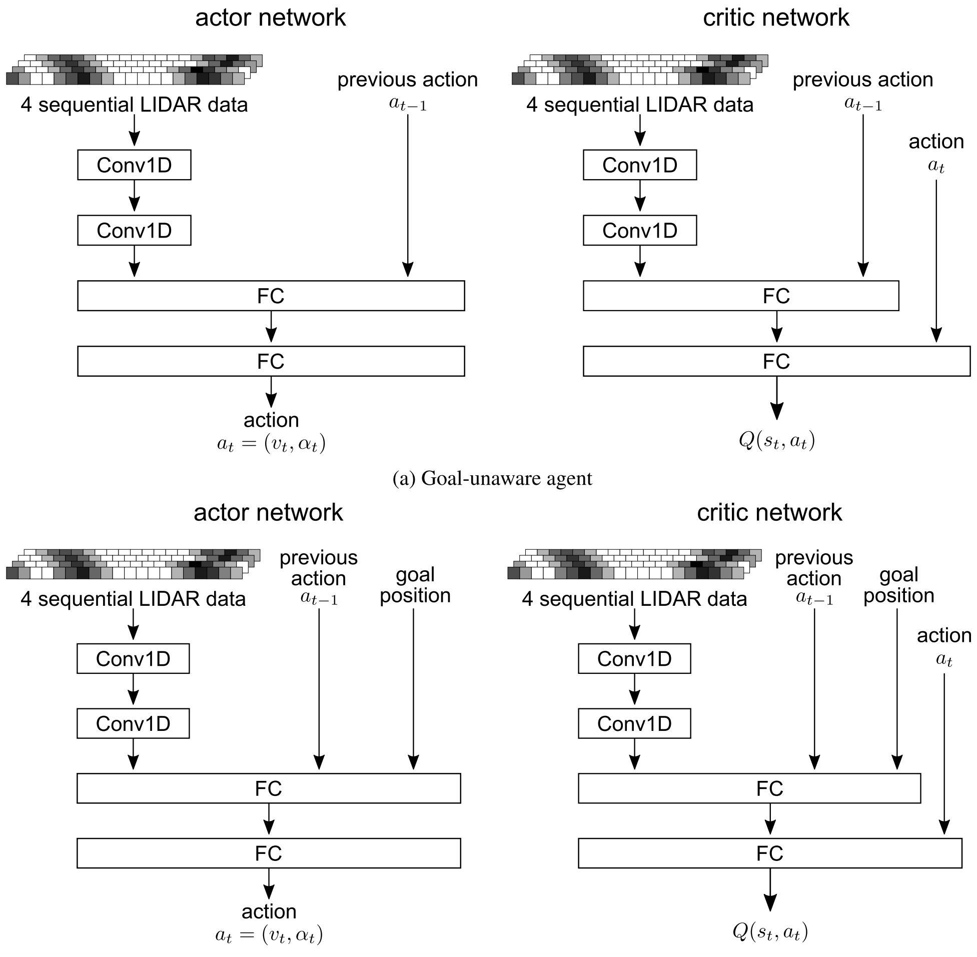
Since a deep deterministic policy gradient method is a proper deep reinforcement learning method for tasks requiring continuous action spaces, we applied this method to our problem to output continuous control action for the vehicle; desired speed and steering angle. The actor-critic network model is designed as described in Fig. 3. The actor network takes the state as input and returns the control action, and the critic network takes the state and the corresponding control action in that state as input and returns the value of the chosen action.

C. PROBLEM STATEMENT

We assume 3 scenarios of traffic signal violation accidents where the ego-vehicle is supposed to go straight through an intersection and a risky vehicle unexpectedly runs a red light in 3 different directions as visualized in Fig. 1. In the first and second scenario, the risky vehicle runs a red light from two lateral directions when the ego-vehicle start to go straight at a green light. In the third scenario, the risky vehicle runs a red light to make a left turn from lateral direction when the ego-vehicle start to go straight at a green light. These scenarios are adopted from the most frequent light-vehicle pre-crash scenarios reported by the NHTSA and occur about 254,000 times, economically costing about 6,627 million dollars annually [2]. Our problem is to make our vehicle self-drive to avoid the collision or minimize the injury severity in 3 described scenarios using a deep reinforcement learning method. The overall flowchart of solving our problem in this paper is shown in Fig. 4.

D. ENVIRONMENT FOR REINFORCEMENT LEARNING

We built up the reinforcement learning environment using a virtual robot experimentation platform (V-REP, Coppelia Robotics, Zurich, Switzerland) [22]. In this simulator, the ego-vehicle is equipped with LIDAR sensor which can simulate 36 LIDAR data around 360 degree range to detect surrounding vehicles. The desired vehicle speed v_t and steering angle α_t at timestep t are calculated by the policy network as the action of the agent. The schematic for the controller of the vehicle motion is described in Fig. 5. From the action (v_t, α_t), the desired angular velocity of the wheel, ω_{ref}^{wheel}, and the desired angle of left and right wheel from straight ahead,



(a) Goal-unaware agent

(b) Goal-aware agent

FIGURE 3. Actor-Critic model used in (a) goal-unaware agent and (b) goal-aware agent. The goal-unaware agent takes 4 sequences of 36 simulated LIDAR data and the action at the previous time step as the state to understand the surrounding situation. The goal-aware agent additionally takes the goal position as the state.

$\left(\alpha_{ref}^{left}, \alpha_{ref}^{right}\right)$, are calculated from the vehicle kinematics of following equations, which is known as the Ackermann steering model in Fig. 6.

$$\omega_{ref}^{wheel} = \frac{v_t}{r_{wheel}} \tag{6}$$

$$\alpha_{ref}^{left} = \arctan\left[\frac{L}{-D + L/\tan\alpha_t}\right] \tag{7}$$

$$\alpha_{ref}^{right} = \arctan\left[\frac{L}{D + L/\tan\alpha_t}\right], \tag{8}$$

where L denotes the wheel base of the vehicle, D denotes the distance between the center line and the wheel, and r_{wheel} denotes the radius of the wheel. These reference values are sent to the ego-vehicle and controlled by internal closed-loop controllers of the simulator. We set the ego-vehicle's goal on the position across the intersection, described in Fig. 11(b). The position of this goal will be chosen to be used or not used as state inputs for the performance comparison. In the following section, the reinforcement learning environments for the agent considering goal and the agent not considering goal will be described.

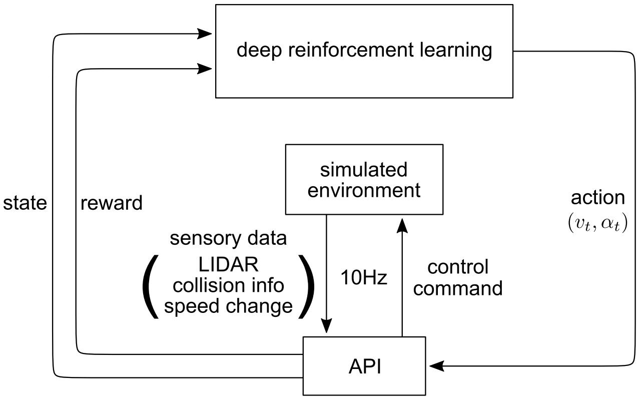


FIGURE 4. The overall flowchart of solving our problem. The API communicates with the simulated environment to observe sensory data and send control commands. Deep reinforcement learning algorithm receive this observation and compute the action while training the actor and critic networks.

1) GOAL-UNAWARE AGENT

The goal-unaware agent takes 4 sequences of 36 simulated LIDAR data and the action chosen at the previous time step as state inputs. It is trained to output an action that minimizes the

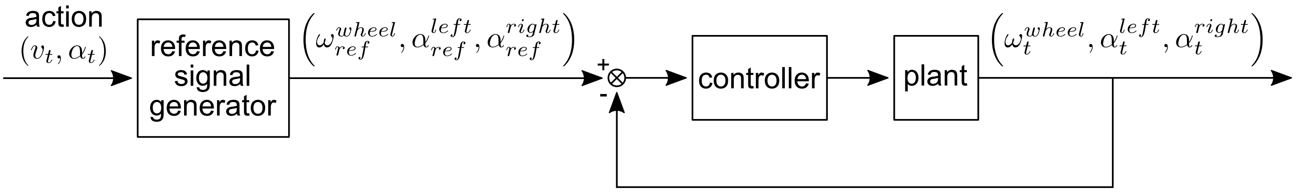

action
(v_t, α_t)
reference signal generator
$\left(\omega_{ref}^{wheel}, \alpha_{ref}^{left}, \alpha_{ref}^{right}\right)$
+
-
controller
plant
$\left(\omega_t^{wheel}, \alpha_t^{left}, \alpha_t^{right}\right)$

FIGURE 5. The schematic for the controller of the vehicle motion. Reference values calculated from the vehicle kinematics are sent to the ego-vehicle and controlled by internal closed-loop controllers of the simulator.

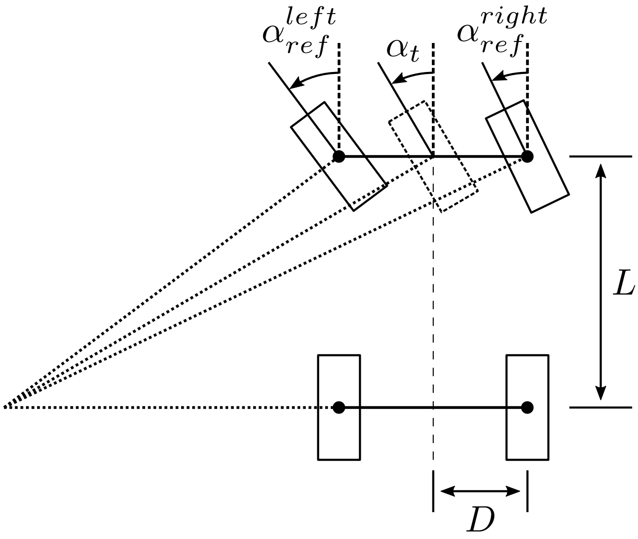


FIGURE 6. A graphical description of the Ackermann steering model. ***L*** denotes the wheel base of the vehicle and ***D*** denotes the distance between the center line and the wheel. From the geometry we can calculate the desired angle of left and right wheel from straight ahead, $\left(\alpha_{ref}^{left}, \alpha_{ref}^{right}\right)$, given the desired steering angle, α_t.

injury severity score of the crash. In each scenario one risky vehicle and one ego-vehicle are involved. At the beginning of each episode, the agent takes decent amount of simulation steps of pre-learning action to generate an impending collision situation. Each episode starts as the risky vehicle runs a red light with a speed of 60km/h. The ego-vehicle then accelerates forward watching the green light ahead. From the moment when it is 1 second before collision, the agent starts making decision. This procedure is visualized in Fig. 7. The type of scenario, the lane of ego-vehicle and risky vehicle, and the existence of neutral vehicles are randomized in every episode for the robustness against various unseen situations, as described in Fig. 8.

Now we define the reward function for minimizing the injury severity of the crash. To this end we need to estimate the injury severity score of the crash in our simulated environment. Sobhani *et al.* [21] studied a model that estimates the injury severity score in the two-vehicle crash using the Newtonian mechanics and generalized linear regression. Using this model, we can approximate the injury severity score in our simulated environment by observing the speed change between before and after the crash and the area of most significant damage. The reward function is defined in (9),

$$r = \begin{cases} 0 & \text{if not crash} \\ -f(\Delta V, A) & \text{if crash} \end{cases} \tag{9}$$

where f is the linear regression model in [21] fitted to estimate the injury severity score (ISS) in the two-vehicle crash given the speed change before and after the collision, ΔV, and the area of most significant damage, A. We will explain this ISS estimation model in more detail in section 11.

2) GOAL-AWARE AGENT

In addition to the goal-unaware agent in 1), we train another agent which is aware of the location of the goal across the intersection. By taking additionally the distance and angle to the goal with respect to the ego-vehicle as state inputs, the agent is expected to learn how to drive into the goal while minimizing the injury severity of the crash.

The reward function for this goal-aware agent is defined in (10),

$$r = \begin{cases} d_{t-1} - d_t & \text{if not crash} \\ -f(\Delta V, A) & \text{if crash} \end{cases} \tag{10}$$

where d_t denotes the distance to the goal at the time step t. Plus to the reward defined in the goal-unaware agent, the goal-aware agent obtains additional reward as much as the distance it approached to the goal.

E. ESTIMATION OF INJURY SEVERITY SCORE

As mentioned in section II-D.1, we use the model from [21] that estimates the injury severity score in the two-vehicle crash using the Newtonian mechanics and a generalized linear regression model. They first identify factors contributing to the speed change ΔV_s of a subject vehicle using the law of conservation of momentum. A Log-Gamma regression model is fitted to estimate ΔV_s of the subject vehicle based on the identified crash characteristics. Then another Log-Gamma regression model is fitted to estimate the Injury Severity Score (ISS) of the crash based on the estimated ΔV_s, the area of most significant damage A, gender and age of the occupant, and the presence of airbag and seat belt. Since we can directly read the speed change ΔV_s from our simulated environment, we used only the ISS model. For the simplicity, we fixed all factors other than the speed change ΔV_s and the area of most significant damage A. The fitted ISS model is

$$ISS = \exp[2.011 \times 10^{-7} \times (0.5 M \Delta V^2) + \alpha] \tag{11}$$

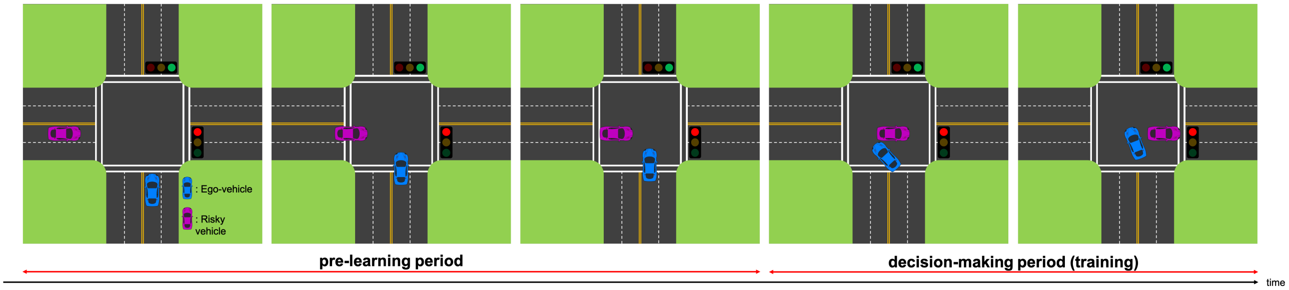


FIGURE 7. Visualization of procedure of every episode. At the beginning of each episode, the agent takes decent amount of simulation steps of pre-learning action to generate an impending collision situation. From the moment when it is 1 second before collision, the agent starts making decision.

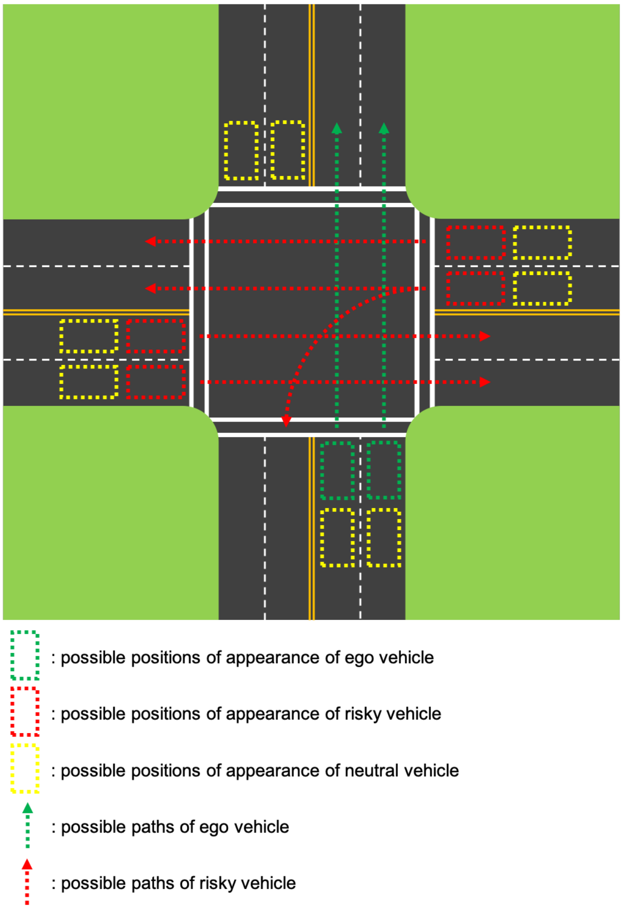

: possible positions of appearance of ego vehicle
: possible positions of appearance of risky vehicle
: possible positions of appearance of neutral vehicle
: possible paths of ego vehicle
: possible paths of risky vehicle

FIGURE 8. Visualization of all possible positions and paths of vehicles. Type of Scenario, position of appearance and paths of ego, risky and neutral vehicles are randomized at the beginning of every episode for obtaining robustness against various unseen situations.

where M is the mass of the vehicle, which is fixed at 1,500kg, and α is the fitted parameter related to the area of significant damage, which is listed in the Table 1.

F. STATE SPACE REPRESENTATION

As explained in Section II-D, 4 sequences of 36 simulated LIDAR data enter the actor network and the critic network as a part of state inputs. Unlike the other low-dimensional state inputs like the previous action a_{t-1} and the goal position, this high-dimensional LIDAR data contains spatio-temporal features; it has depth data along 36 angle indices spatially, stacked along 4 temporal sequences. For extracting the spatio-temporal features implicated in this high-dimensional LIDAR data, we used 1-D convolution layers as described in Fig. 3. This spatio-temporal feature extraction technique is necessary because the surrounding situation rapidly changes in our problem and our agent has to deal with it.

TABLE 1. Table of α value according to the area of most significant damage. The 3 types of area listed are described in Fig. 9.

Area of most significant damage (A)	α
Near side	2.698
Far side	2.038
Rear side	1.991

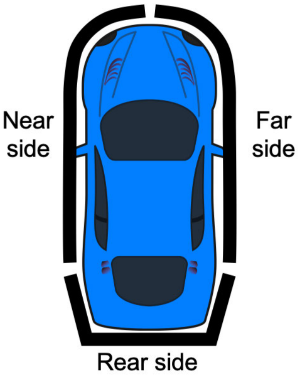


FIGURE 9. The area of significant damage is classified to 3 levels; near and far side to the driver and rear side of a vehicle.

III. EXPERIMENT

In this section, we first train two agents (goal-unaware agent and goal-aware agent) as described in the previous section using a deep deterministic policy gradient method for the three scenarios and compare their performance against human drivers and an autonomous emergency braking system. In the following we describe the experiment apparatus for human drivers and an autonomous emergency braking system and study the results with statistical analysis.

TABLE 2. Results of experiment. 30 trials were made for each scenario per each approach.

	scenario	our goal-aware agent	our goal-unaware agent	human drivers	autonomous emergency braking
avoidance percentage (%)	1	82.00	87.00	16.67	31.50
	2	92.50	89.00	16.67	5.30
	3	97.00	92.50	0.00	0.00
	all	92.17	87.83	11.11	5.30
average injury score	1	-1.89	-2.66	-13.20	-17.99
	2	-0.54	-0.88	-14.21	-19.22
	3	-0.43	-1.11	-13.97	-21.90
	all	-0.94	-1.52	-13.79	-19.67

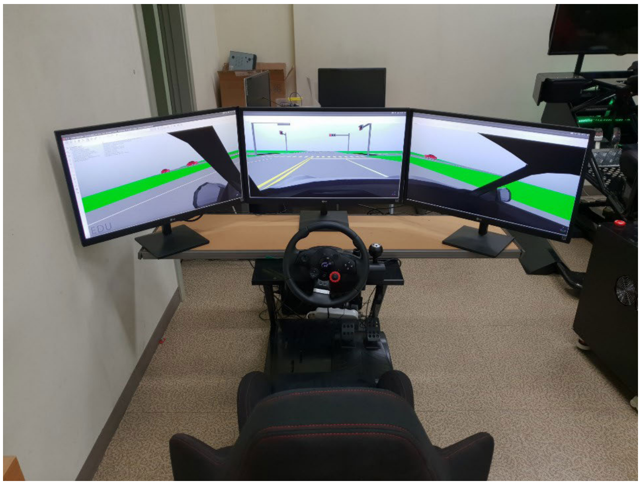


FIGURE 10. The experimental apparatus for human driver. A human driver can manipulate the steering wheel, gas pedal, and brake pedal and is available with the front, left and right view through 3 displays.

A. HUMAN DRIVERS

The experiment apparatus for human drivers is described in Fig. 10. A human driver can manipulate the steering wheel, gas pedal, and brake pedal and is available with the front, left and right view through 3 displays. Before experiment each subject is allowed to freely drive a simulated vehicle in an empty driveway for 10 minutes to get familiar with the manipulation of our experiment apparatus. Each subject is made to drive across one normal intersection without any risky vehicle and in the second intersection after seeing the green light encounters one of the three risky scenarios. Since we deal with unexpected collision scenarios, subjects are not informed of the risky situation in advance. For each scenario, the injury severity scores of 30 subjects are recorded.

B. AUTONOMOUS EMERGENCY BRAKING SYSTEM

Various vehicle manufacturers improve the safety of their vehicle by applying Autonomous Emergency Braking (AEB) systems. An AEB system automatically apply braking to the vehicle when a potential collision is detected by sensors. We implement a simple AEB system in our experimental simulator where the ego-vehicle is controlled to take a braking action when it detects an object in front of itself within 3 meters. Here we assume our problem is a stationary low speed scenario referring to [23], where the ego-vehicle's speed is between 10km/h to 50km/h before the accident. Then we evaluate its performance in our offered scenarios.

TABLE 3. Results of one-way ANOVA.

	p-value
goal-unaware agent vs. human drivers	0.013
goal-unaware agent vs. AEB system	0.028
goal-aware agent vs. human drivers	0.008
goal-aware agent vs. AEB system	0.019

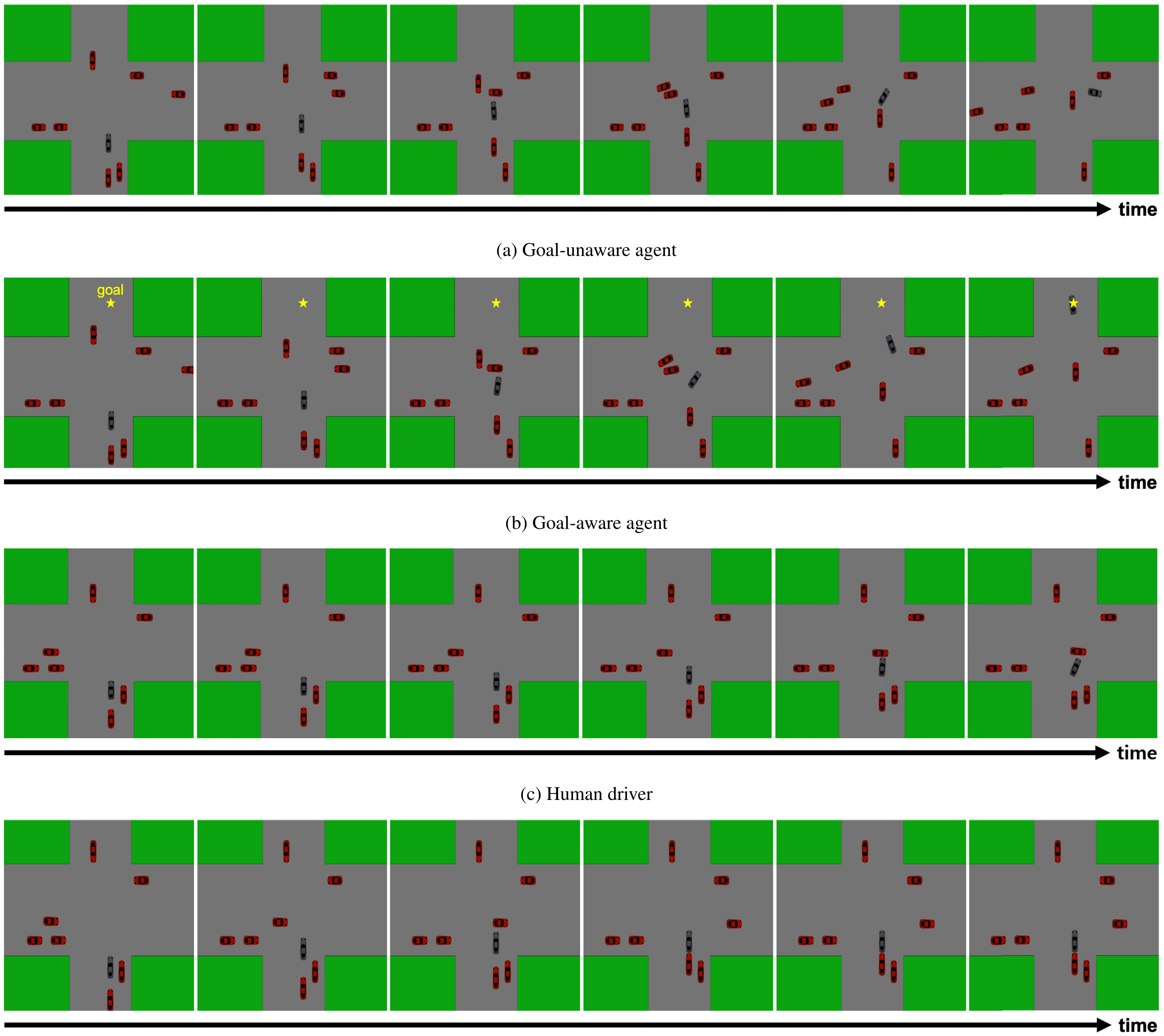
C. EVALUATION METRICS

We evaluate the performance of each method using following metrics in our experiment.

- Avoidance percentage: the percentage of trials where the ego-vehicle successfully avoids the rushing of the risky vehicle without any subsequent collision with neutral vehicles.
- Average injury score: the average injury severity score recorded throughout the experiment.

IV. RESULTS

Table 2 shows the results of experiment. The two reinforcement learning agents of ours outperformed the human driver and the AEB system in both the percentage of collision avoidance and the average injury score. The one-way ANOVA is conducted to see whether the differences of performance between the comparison groups are statistically significant. We had the p-values of the difference of the mean between the goal-unaware agent, the goal-aware agent, human drivers, and the AEB system shown in Table 3. The result shows that our two agents outperformed both human drivers and the AEB system in avoiding unexpected collisions by a statistically significant gap. Most human drivers had difficulties in even responding to the rush of risky vehicle and had themselves injured as shown in Fig. 11c. Although the AEB system succeeded in avoiding the risky vehicle in most of the experiment, it was not able to avoid subsequent accidents, for example a rear-end collision with a rear vehicle, as shown in Fig. 11d. AEB system can only consider the frontal distance and stop, which make the system vulnerable to accidents coming from its rear or lateral side.



(a) Goal-unaware agent

(b) Goal-aware agent

(c) Human driver

(d) Autonomous Emergency Braking system

FIGURE 11. 6 sequential pictures showing the behavior of the three comparison group, (a) Goal-unaware agent, (b) Goal-aware agent, and (c) Autonomous Emergency Braking system in scenario 1. The black vehicle is the ego-vehicle and the red vehicle dashing from the right side is the risky vehicle.

As for the two reinforcement learning agents of ours, both the goal-unaware agent and the goal-aware agent avoid the collision with the risky vehicle in most time. After the first avoidance against the risky vehicle, however, the goal-unaware agent often failed to avoid the subsequent accidents from the rear and lateral side as shown in Fig. 11a, while the goal-aware agent tended to drive forward avoiding the subsequent accidents until the end of the episodes as shown in Fig. 11b. We discuss some insight on these results of our two agents in the following.

The goal-unaware agent is trained to optimize a single-task problem, which minimizes the injury severity of the ego-vehicle. On the other hand, the goal-aware agent is trained to optimize a multi-task problem, in which the agent has to consider both minimizing the injury severity and approaching to the goal (driving across the intersection). Our reinforcement learning environment can be considered as a particular environment in which general traffic rules are applied. Vehicles other than ego-vehicle and the risky vehicle keep the traffic signal and drive straight in the middle of their lanes. Indeed, the difference of performance between the goal-unaware agent and the goal-aware agent may come from whether or not the ego-vehicle avoids the subsequent accident against the other neutral vehicles. Once the ego-vehicle avoids the collision against the risky vehicle, the situation becomes just normal driving situation, where every vehicle is supposed to abide by the traffic rules. The goal-aware agent is trained considering this intrinsic objective of driving, which is designed as the goal-approaching reward in (3) in our case. In the other hand, the goal-unaware agent is ignorant of this intrinsic rule of driving and trained to optimize only a single task of minimizing the injury severity, which could be a reason for underperforming than the goal-aware agent.

V. CONCLUSION

We synthesized the self-driving policy that minimizes the injury severity when unexpected traffic signal violation accidents occur at an intersection. We showed that our agents outperform both human drivers and the autonomous emergency braking system in the percentage of collision avoidance and the average injury severity by statistically significant gap. We also showed that the agent trained with the goal information performed slightly better and showed more desirable driving behaviors after the collision avoidance than the agent trained without the goal information.

However, there are limitations that, for example, we couldn't consider a lane detection since we used only LiDAR. To improve from these limitations, our future work can consider self-driving policy using the visual sensory input, which is more challenging task since it is in much higher dimensional state space. Some other state-of-the-art deep reinforcement learning method like the A3C might be used in that future work.

REFERENCES

[1] S. Singh, "Critical reasons for crashes investigated in the national motor vehicle crash causation survey," Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. DOT HS 812 506, 2018.

[2] W. G. Najm, R. Ranganathan, G. Srinivasan, J. D. Smith, S. Toma, E. Swanson, and A. Burgett, "Description of light-vehicle pre-crash scenarios for safety applications based on vehicle-to-vehicle communications," Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. DOT HS 811 731, 2013.

[3] N. Kaempchen, B. Schiele, and K. Dietmayer, "Situation assessment of an autonomous emergency brake for arbitrary vehicle-to-vehicle collision scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 678–687, Dec. 2009.

[4] T. Taleb, A. Benslimane, and K. Ben Letaief, "Toward an effective risk-conscious and collaborative vehicular collision avoidance system," *IEEE Trans. Veh. Technol.*, vol. 59, no. 3, pp. 1474–1486, Mar. 2010.

[5] R. Deng, B. Di, and L. Song, "Cooperative collision avoidance for overtaking maneuvers in cellular V2X-based autonomous driving," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4434–4446, May 2019.

[6] J. Ji, A. Khajepour, W. W. Melek, and Y. Huang, "Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 952–964, Apr. 2017.

[7] D. S. Breed, W. C. Johnson, and W. E. DuVall, "Intersection collision avoidance techniques," U.S. Patent 8 000 897, Aug. 16, 2011,

[8] D. Gerónimo, A. M. López, A. D. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 7, pp. 1239–1258, Jul. 2010.

[9] J. Park, H. Son, J. Lee, and J. Choi, "Driving assistant companion with voice interface using long short-term memory networks," *IEEE Trans. Ind. Informat.*, vol. 15, no. 1, pp. 582–590, Jan. 2019.

[10] S. A. Fayazi and A. Vahidi, "Mixed-integer linear programming for optimal scheduling of autonomous vehicle intersection crossing," *IEEE Trans. Intell. Veh.*, vol. 3, no. 3, pp. 287–299, Sep. 2018.

[11] L. Tai, G. Paolo, and M. Liu, "Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 31–36.

[12] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," Sep. 2015, *arXiv:1509.02971*. [Online]. Available: https://arxiv.org/abs/1509.02971

[13] M. Jaritz, R. de Charette, M. Toromanoff, E. Perot, and F. Nashashibi, "End-to-end race driving with deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2018, pp. 2070–2075.

[14] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2016, pp. 1928–1937.

[15] D. Isele, A. Cosgun, K. Subramanian, and K. Fujimura, "Navigating occluded intersections with autonomous vehicles using deep reinforcement learning," May 2017, *arXiv:1705.01196*. [Online]. Available: https://arxiv.org/abs/1705.01196

[16] A. E. Sallab, M. Abdou, E. Perot, and S. Yogamani, "End-to-end deep reinforcement learning for lane keeping assist," Dec. 2016, *arXiv:1612.04340*. [Online]. Available: https://arxiv.org/abs/1612.04340

[17] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[18] S.-Y. Oh, J.-H. Lee, and D.-H. Choi, "A new reinforcement learning vehicle control architecture for vision-based road following," *IEEE Trans. Veh. Technol.*, vol. 49, no. 3, pp. 997–1005, May 2000.

[19] K. Menda, Y.-C. Chen, J. Grana, J. W. Bono, B. D. Tracey, M. J. Kochenderfer, and D. Wolpert, "Deep reinforcement learning for event-driven multi-agent decision processes," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 4, pp. 1259–1268, Apr. 2019.

[20] C. Lu, H. Wang, C. Lv, J. Gong, J. Xi, and D. Cao, "Learning driver-specific behavior for overtaking: A combined learning framework," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 6788–6802, Aug. 2018.

[21] A. Sobhani, W. Young, D. Logan, and S. Bahrololoom, "A kinetic energy model of two-vehicle crash injury severity," *Accident Anal. Prevention*, vol. 43, no. 3, pp. 741–754, May 2011.

[22] E. Rohmer, S. P. N. Singh, and M. Freese, "V-REP: A versatile and scalable robot simulation framework," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 1321–1326.

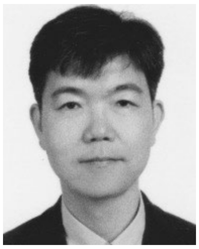
[23] W. Hulshof, I. Knight, A. Edwards, M. Avery, and C. Grover, "Autonomous emergency braking test results," in *Proc. 23rd Int. Tech. Conf. Enhanced Saf. Vehicles (ESV)*, May 2013, pp. 1–13.

MYOUNGHOE KIM received the B.S. degree in mechanical engineering from Yonsei University, Seoul, South Korea, in 2017, where he is currently pursuing the Ph.D. degree with the Machine Learning and Control Systems Laboratory. His research interests include deep reinforcement learning, self-driving vehicles, collision avoidance strategies, and multiagent systems.

SEONGWON LEE is currently pursuing the B.S. degree in mechanical engineering with Yonsei University, Seoul, South Korea. His research interests include machine learning and deep reinforcement learning, and self-driving vehicles.

JAEHYUN LIM received the B.S. degree in mechanical engineering from Yonsei University, Seoul, South Korea, in 2017, where he is currently pursuing the Ph.D. degree with the Machine Learning and Control Systems Laboratory. His research interest includes deep (inverse) reinforcement learning and path planning of mobile platforms.

JONGEUN CHOI (Member, IEEE) received the B.S. degree in mechanical design and production engineering from Yonsei University, Seoul, South Korea, in 1998, and the M.S. and Ph.D. degrees in mechanical engineering from UC Berkeley, in 2002 and 2006, respectively. He is currently the Head of the Machine Learning and Control Systems Laboratory and an Associate Professor with the School of Mechanical Engineering, Yonsei University. Since 2019, he has been the Chairperson of the Department of Vehicle Convergence Engineering, Yonsei University, funded by Hyundai Motor Company. Prior to joining Yonsei University, he worked for ten years as an Associate Professor with the Department of Mechanical Engineering, Michigan State University, from 2012 to 2016, and an Assistant Professor with the Department of Electrical and Computer Engineering, Michigan State University, from 2006 to 2012. His current research interests include machine learning, systems and control, system identification, and Bayesian methods, with applications to autonomous robots, self-driving vehicles, mobile sensor networks, (physical) human and robot interaction, and biomedical problems. He is a member of ASME. He co-organized and co-edited special issues on Stochastic Models, Control, and Algorithms in Robotics in JDSMC, from 2014 to 2015, and Deep-Learning Based Sensing Technologies for Autonomous Vehicles in Sensors, in 2018. He received the Best Conference Paper Award at the 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), in 2015. His articles were finalists for the Best Student Paper Award at the 24th American Control Conference (ACC), in 2005, and the Dynamic System and Control Conference (DSCC), in 2011 and 2012. He was a recipient of the NSF CAREER Award, in 2009. He has served as an Associate Editor for the IEEE ROBOTICS AND AUTOMATION LETTERS (RA-L), *Journal of Dynamic Systems, Measurement and Control* (JDSMC), and *International Journal of Precision Engineering and Manufacturing* (IJPEM). He serves as a Senior Editor for Ubiquitous Robots (UR), in 2020.



SEONG GU KANG received the B.S. and M.S. degrees in mechanical engineering from Yonsei University, in 1990 and 1992, respectively, and the Ph.D. degree in mechanical engineering from UC Berkeley, in 2005. He is currently a Research Professor with the Institute of Science and Technology, Korea University, Sejong. Before coming to Korea University, he worked for Samsung Electronics and Samsung Display. His research interests include machine learning applications, data-driven control for autonomous vehicles, mechanical equipment design, and energy harvesting technologies.