Journal of Robotics and Control (JRC)
Volume 5, Issue 1, 2024
ISSN: 2715-5072, DOI: 10.18196/jrc.v5i1.20371

Key Factors that Negatively Affect Performance of Imitation Learning for Autonomous Driving

Estiko Rijanto [1*], Nelson Changgraini [2], Roni Permana Saputra [3], Zainal Abidin [4]
[1, 3] Research Center for Smart Mechatronics, National Research and Innovation Agency (BRIN), Bandung 40135, Indonesia
[2] Graduate School of Engineering, University of Tokyo, Japan
[4] Department of Mechanical Engineering, Institut Teknologi Bandung (ITB), Bandung, Indonesia
Email: [1] estiko.rijanto@brin.go.id, [2] nelsoonchanggraini@gmail.com, [3] roni.permana.saputra@brin.go.id,
[4] zapauitb@gmail.com
*Corresponding Author

***Abstract*—Conditional imitation learning (CIL) has proven superior to other autonomous driving (AD) algorithms. However, its performance evaluation through physical implementations is still limited. This work contributes a systematic evaluation to identify key factors potentially improving its performance. It modified convolutional neural network parameter values, such as reducing the number of filter channels and neuron units, and implemented the model into a vision-based autonomous vehicle (AV). The AV has front-wheel steering with an Ackermann mechanism since it is commonly used by passenger cars. Using the Inertia Measurement Unit, we measured the vehicle's location and yaw angle along the experimental route. The AV had to move autonomously through new road sectors in the morning, afternoon, and night. First, an overall performance evaluation was carried out. The results showed a 99% success rate from 648 evaluation experiments under different conditions in which the 1% failure rate happened at new intersections. Then, a turning performance evaluation was conducted to identify key factors leading to failure at new intersections. They include fast speed, dazzling light reflection, late navigation command change instant, and the untrained turning driving pattern. The AV never failed while driving on the trained routes. It had a 100% success rate when driving slower, even under various lighting conditions and at various driving patterns, including untrained intersections. Although this study is limited to identifying key factors at three constant speeds, the results become the foundation for future research to improve CIL performance for AD, including by incorporating multimodal fusion and multi-route networks.**

Keywords—Autonomous Driving; Convolutional Neural Networks; Front-Wheel Steering; Imitation Learning.

I. INTRODUCTION

Significant research progress has been achieved in autonomous driving (AD) of ground vehicles [1]–[14]. Chen et al. categorized two significant paradigms for vision-based AD: mediated perception approaches that parse an entire scene to make a driving decision and behavior reflex approaches that directly map an input image to a driving action by a regressor [15]. For behavior reflex approaches, an artificial neural network was designed to control an autonomous navigation test vehicle for road following [16]–[19].

A learning system that takes raw color images from forward-pointing cameras and maps them to a set of steering angles through a single trained function was termed end-to-end learning by the authors in [20], [21]. They developed a 6-layer convolutional neural network (CNN) for vision-based obstacle avoidance of an off-road 1/10 scale electric truck. In [22], a CNN framework was adopted to develop an end-to-end controller that manages a full-scale car following the lane on local roads based on image input. The authors developed a method for determining which elements in the image most influence steering decisions [23]. The framework was also used in [24] to build a low-cost modular automated guided vehicle (AGV) capable of autonomously following the lane in a specific fixed route. Furthermore, several methods have been proposed to improve the effectiveness of the end-to-end AD approach. For instance, a CNN was combined with a feedforward network with a fully connected hidden layer for lane following control of a 1/5-scale car, as presented in [25].

Most early research studies on imitation learning (IL) for AD have focused on lane following and obstacle avoidance problems. Later, more research pushed urban driving with nontrivial road layouts and traffic [26]–[30]. Codevilla et al. proposed an IL method that maps camera images and incorporates high-level navigation input to control an autonomous vehicle (AV) to navigate the intersections, retrospectively known as conditional imitation learning (CIL) [31]. This method used a deep learning architecture for the image processing module, which consists of 8-layer CNN and 2-layer Full Connected Network (FCN). It was successfully implemented using a 1/5 scale truck in a field experiment. Sauer et al. proposed a direct perception approach, called conditional affordance learning (CAL), that maps video input to an intermediate representation and combines it with high-level directional inputs using specialized task networks to produce affordances [32]. The authors demonstrated that CAL outperformed CIL when tested on the CARLA's simulator [33]. However, the work has not been proven in field experiments.

Chen et al. proposed a two-stage learning method involving a privileged agent and a purely vision-based sensorimotor agent [34]. The authors followed the prior work by Codevilla et al., in which the network branched into four heads, each producing a K-channel heatmap. It outperforms CIL and CAL when tested using the CARLA simulator. However, it still also needs to be proven in field experiments. Recently, CIL has been adopted to process multimodal inputs: RGB and depth modalities [35]. The experiments in

the CARLA simulator proved that the CIL with early fusion outperformed CAL.

CIL falls into the end-to-end method for AD. A few review articles on AD based on end-to-end methods have been published [36]–[42]. Reference [43] provides a comprehensive overview of state-of-the-art hardware-software practices. The authors pointed out that end-to-end methods are an emerging trend in AD technology. The authors in [44] concluded that researchers rely heavily on data generated from simulated environments. In [45], the authors underlined that one of the most challenging tasks with end-to-end deep driving models remains explainability in decision-making. Meanwhile, according to the author in [46], the requirement for explainability in deep-learning-based self-driving models is influenced by multiple factors, including the individual seeking explanations, their level of knowledge, and the amount of time available for analyzing the explanation. Several research efforts have been undertaken to enhance the comprehensibility of the self-driving model based on deep learning [47]–[58].

The authors reviewed end-to-end driving [38], and they described the standard learning methods in end-to-end driving as IL [59] and reinforcement learning (RL) [60]–[63]. In [37], the authors reviewed 17 articles published from 2017 to 2021 regarding end-to-end AD in urban environments. They compared the performances of IL and RL approaches using two benchmarks in the CARLA simulator. It was found that the most effective approach was IL-based architecture for the CoRL2017 benchmark and RL-based architecture for the NoCrash benchmark. Nevertheless, which approach is leading is yet to be more conclusive. Other authors in [64] surveyed IL techniques for end-to-end AD and compared 34 articles published from 2016 to 2022. Only two articles implemented the controllers in real-world experiments in urban driving: [31], [65]. Both articles rely on CIL. Field experiments are necessary since simulation can not capture real-world complexities.

Although CIL has been implemented in field experiments and has proven superior to other algorithms, its performance evaluation is still limited [31], [65]. For example, in the work presented in [31], the authors did not distinguish clearly between training and evaluation routes in their physical implementation, nor did they systematically evaluate the effect of lighting. Moreover, they did not consider the effects of vehicle speed and navigation command change instant [26]. The authors collected data for training from human demonstrations over 30 driving hours in a densely populated urban environment where the drivers were free to choose random routes [30]. The training data is imbalanced, with the majority driving straight and a significant portion stationary. They selected two routes for testing. The testing environment contains uncontrollable time-dependent factors, e.g., weather, lighting, and road users. These randomness and uncontrollable conditions made it less systematic and challenging to identify key factors more deterministically.

This work provides research contributions as follows:

1) It modified the CIL algorithm by reducing the number of filter channels and neuron units. It developed an AD controller for a 1/10 scale AV that mechanically resembles a full-scale car with the Ackermann front-wheel steering. The camera model is valuable for scaling up to a full-scale car.
2) It performed field experiments in a more deterministic manner by evaluating the effects of critical factors: (a) experimental lighting conditions in the morning, afternoon, and night; (b) vehicle speed; (c) navigation command change instant at intersections; (d) the clear separation between training and evaluation road sectors.
3) It explores the possibilities to improve the performance of AD in urban environments by considering perception accuracy and robustness, vehicle dynamics, and real-time implementation.

The remainder of this paper is organized as follows. Section two presents CIL. The experimental platform is described in Section 3, including the AV and camera model. Section 4 describes the training, validation, and performance evaluation experiment scenarios. Section 5 reports the results and discussions. Finally, Section 6 presents our conclusion.

II. METHOD

This work was carried out sequentially, as illustrated in the flowchart in Fig. 1. First, we briefly overviewed the CIL. We modified it to reduce computing load while maintaining performance. Second, we developed an experimental platform that allows efficient experiments but still inherits the characteristics of a real car. Third, we established a systematic experimental performance evaluation method. Fourth, we presented experimental results and discussion. The results include evaluating overall performance and turning performance at new intersections. We investigated the effects of critical factors: road pattern, lighting, speed, and navigation command change instant. Finally, we conclude. In the following, these aspects of methodology are explained in comprehensive detail.

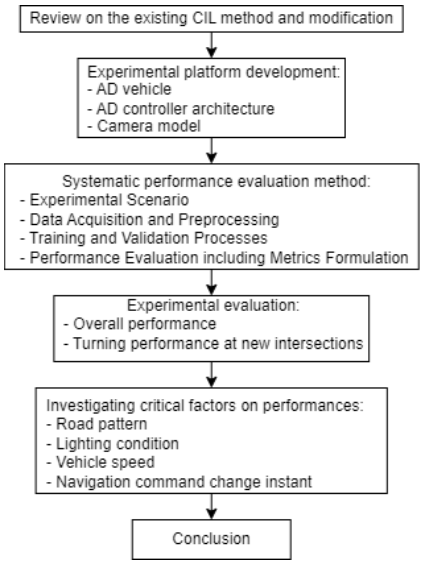


Fig. 1. Flow chart of the research method

A. Conditional Imitation Learning

This section briefly provides an overview of imitation and conditional imitation learning. We then describe our model of conditional imitation learning used in this paper. Imitation

learning trains a controller to mimic an expert using a dataset $D = \{\langle o_i, a_i \rangle\}_{i=1}^{N}$ generated by the expert. In each step, i, the expert receives an observation o_i and takes an action a_i. The dataset is composed of N pairs of observations and actions. The objective is to find the parameter values of a model approximator $F(o_i; \theta)$ that fits the mapping of observations to actions as expressed in (1).

$$\min_{\theta} \sum_{i} l(F(o_i; \theta), a_i) \quad (1)$$

It requires an assumption that a function E exists that maps observations to the expert's actions: $a_i = E(o_i)$. However, when an autonomous vehicle approaches an intersection, the driver's subsequent action is explained by the observations and is affected by the driver's intention.

The driver's intention is exposed to the controller by introducing an additional command input c [31]. During training, the expert provided commands. They provided information about the expert's decision-making. A driver or a navigation system can provide commands to affect the controller's behavior at the test time. The training dataset becomes $D = \{\langle o_i, c_i, a_i \rangle\}_{i=1}^{N}$. The objective of conditional imitation learning (CIL) is given by (2).

$$\min_{\theta} \sum_{i} l(F(o_i, c_i; \theta), a_i) \quad (2)$$

A deep artificial neural network expresses the controller $F(o_i, c_i; \theta)$. The network takes images as the input. By adopting the branched network architecture of command-conditional imitation learning from Codevilla et al., we assume a discrete set of commands, $C = \{c^0, \ldots, c^K\}$, and introduce a particular branch A^i for each command c^i.

Fig. 2 illustrates the network architecture used in this paper for command-conditional imitation learning. The particular branch A^i learns sub-policies that correspond to the navigational commands. We set three modules for decision-making at an intersection that enable going straight or following the lane, turning left, and turning right. The image module is implemented as a combination of CNN and FCN, whereas the command module is an FCN.

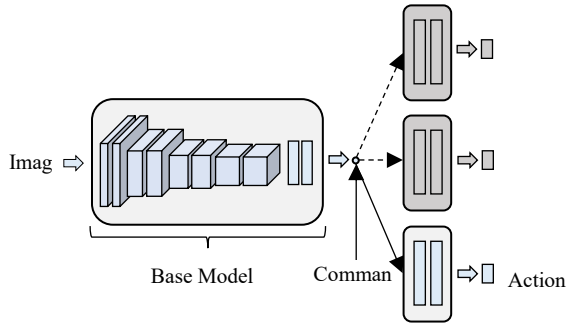


Fig. 2. Network architecture for CIL

We adapted the architecture of CIL originally proposed in [31] with modifications to the network details. The input image of our CIL model has dimensions of 100×220 pixels. We constructed the base model using eight CNN layers and two FCN layers. A batch normalization accompanies each CNN layer. The first CNN layer has a kernel size of five, followed by a kernel size of three in the remaining layers. The first, third, fifth, and seventh CNN layers have a stride of two, whereas the remaining layers have a stride of one. The two FCN layers each contain 128 neuron units. Then, the model starts to branch, with each branch specializing in each navigation command. Each branched model consists of two FC hidden layers with neuron sizes of 256 and 512 for the left, right, and straight models, respectively. We applied a rectified linear unit activation function after each hidden layer and batch normalization after all convolutional layers.

Some differences exist between our model and that used in [31]. First, the seventh CNN layer of their model has a stride of one. Second, they applied a dropout layer after each convolutional layer. However, applying dropout after the convolutional layer decreased model performance in our case. Therefore, we did not apply a dropout layer to the convolutional layers. Finally, the number of filter channels in the convolutional layers and neuron units in the fully connected layers are much smaller in our model than theirs.

Our model assumes a constant vehicle speed, so the controller's action is only the steering angle. This assumption makes the vehicle dynamics from the steering angle to the yaw rate a linear time-invariant system [66]. Therefore, the effect of speed on performance can be analyzed systematically.

B. Experimental Platform

We built an AV by retrofitting a 1/10 scale radio-controlled car, model HG P408 US Military Vehicle (Fig. 3). It weighs 5.7 kg, has a width of 0.225 m, and has a turning radius of 1.3 m.

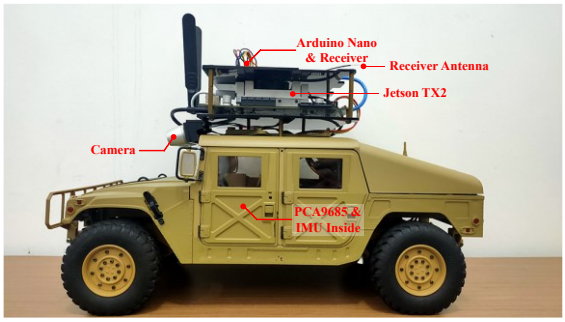


Fig. 3. Autonomous driving vehicle

We mounted a forward-pointing RGB camera on the front top of the vehicle. The camera has a frame resolution of 1920×1080 pixels, a frame rate of 30 fps, a field of view (FoV) (H×V) of 69° × 42°, and a sensor resolution of two MP. The central processing computer is Nvidia Jetson TX2, which oversees data acquisition, processing it, and sending control commands to the steering servo and electronic speed controller (ESC). The receiver unit receives the navigational command from the remote controller and sends it to the microcontroller via a digital input/output channel. The Jetson TX2 receives the navigational command, the camera's image, and measurement data from the inertial measurement unit (IMU) via a USB cable (Fig. 4). It sends the steering and throttle values calculated by the autonomous controller through a PCA9685 servo driver.

We installed the Jetson Package into Jetson TX2, including Linux for Tegra, TensorRT, CUDA, cuDNN, and several computer vision libraries. TensorRT, CUDA, and cuDNN are libraries published by NVIDIA. We also installed other libraries: Intel librealsense and pyrealsense to access the Intel® RealSenseTM camera, OpenCV for real-time computer vision, and TensorFlow. We use Python scripts for data collection, neural network training, neural network testing, and other purposes.

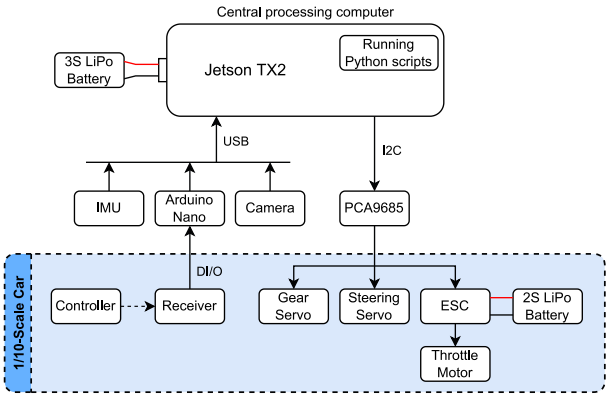


Fig. 4. Schematic of the autonomous driving architecture

We developed a camera model to obtain parameter values related to the images, as illustrated in Fig. 5. The x-axis extends in the vehicle's forward direction, the y-axis points to the vehicle's left, and the z-axis faces upward, perpendicular to the ground. The camera's optical axis points to a certain point on the road and captures the entire area inside its FoV. FoVV denotes the camera's FoV in the vertical direction. H_c denotes the camera's height on the road surface. The pitch angle θ is between the camera's optical axis and the horizontal line.

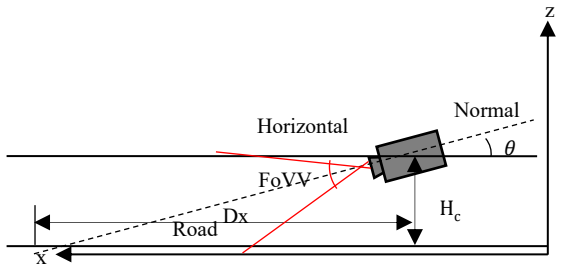


Fig. 5. Camera model of the camera fixed on the top of the AV

We define some essential parameters as follows:

1) Longitudinal distance D_x is the distance between the camera and the camera's focus point on the road, measured along the x-axis.
2) Short longitudinal distance D_{xs} is the distance between the camera and the lowest point viewed by the camera, measured along the x-axis.
3) Horizon distance D_h is the distance between the camera and the object the camera views on the horizontal line.
4) Horizon height H_h is the height of the highest point of the camera views at the horizon distance measured from the horizontal line along the z-axis.

The longitudinal distance D_x is given by (3)

$$D_x = \frac{H_c}{tan(\theta)} \tag{3}$$

The short longitudinal distance D_{xs} and horizon height H_h are obtained by incorporating the FoV angle along the horizontal axis, as given in (4) and (5)

$$D_{xs} = \frac{H}{tan\left(\theta + \frac{VoVV}{2}\right)} \tag{4}$$

$$H_h = D_h\, tan\left(\left|\theta - \frac{VoVV}{2}\right|\right) \tag{5}$$

where $tan\left(\theta + \frac{VoVV}{2}\right)$ represents the horizon height ratio.

When the vehicle runs at a speed of v_V and the sampling time of the computer is t_s, we can calculate the longitudinal displacement ΔD_{xs} using the following relationship.

$$\Delta D_{xs} = D_{xs}(k) - D_{xs}(k-1) = v_V t_s \tag{6}$$

We must determine the appropriate sampling time to get an acceptable longitudinal displacement value by considering the vehicle velocity. The camera continuously records the image at the focus point at time step k up to time step $k+n$, where n is the image repetition number given by (7).

$$n = \frac{D_x - D_{xs}}{v_V t_s} \tag{7}$$

Table I lists parameter values of the camera model. The mounted camera's pitch angle and height were determined based on practical considerations. We found the appropriate nominal speed of 0.4 m/s based on trial and error, considering the maximum capability of the computer vision of 30 fps. These parameter values become a foundation for scaling up the experiment to a full-scale car in the future.

TABLE I. PARAMETER VALUES OF THE CAMERA MODEL

Parameter	Value
Camera height H_c	0.18 m
Camera pitch angle	15°
Sampling rate	0.05 s
Longitudinal distance D_x	0.672 m
Short longitudinal distance D_{xs}	0.248 m
Nominal vehicle speed	0.4 m/s
Longitudinal displacement ΔD_{xs}	0.02 m
Image repetition number	22

C. Experimental Performance Evaluation

This section describes experimental scenarios, data acquisition and preprocessing, training and validation, and the performance evaluation process.

First, a systematic experimental scenario is explained, including experimental route, driving patterns, training road sectors, testing road sectors, and lighting conditions setting. Fig. 6 illustrates a top view of the route used in our experimental scenario. The plotted virtual numbers (1 to 16) represent the locations along the route to define the road sectors (RSs) and trajectory, where the AV navigated based

on the scenario. We set nine unique driving patterns (DPs) to drive the AV along the route. The patterns include lane following on a straight road (DP1), moving straight by passing an intersection on the left-hand side (DP2) or right-hand side (DP3), turning left (DP4) or right (DP5) at a curve, turning left (DP6) or right (DP7) at an intersection, and turning left (DP8) or right (DP9) facing a T junction.

Table II summarizes the experimental scenarios denoted by the road sector's numbers, with the trajectory fraction from one corresponding location to another and its driving pattern. The objective is to evaluate the AD performance at various driving patterns under different conditions.

The experimental scenarios consist of the following:

1) Training road sectors: RS1 (DP1), RS2 (DP6), RS3 (DP1), RS4 (DP5), RS5 (DP1), RS6 (DP3), RS7 (DP1), RS8 (DP1), RS9 (DP7), RS10 (DP1), RS11 (DP4), and RS12 (DP1). They are shown as the blue line in Fig. 6 (a) and indicated by T in Table II.
2) Validation testing road sectors: RS8 (DP1), RS12 (DP1), RS13 (DP4), RS14 (DP1), RS15 (DP6), RS16 (DP1), RS17 (DP9), and RS18 (DP3). The validation testing road sectors are shown as the red line in Fig. 6 (a) and indicated by E in Table II.
3) Performance evaluation road sectors: overall performance evaluation is carried out through the same road sectors as the validation sectors. The turning performance evaluation is conducted through road sectors shown as purple and green lines in Fig. 6 (b).

We selected evaluation road sectors that reflected seven driving patterns. The CIL model has already been trained to experience DP1 but has never been trained for DP9. Moreover, the evaluation road sectors differed from the training road sectors for DP3, DP4, DP6, and DP7.

In order to evaluate the CIL's efficacy under various lighting conditions, we conducted experiments with multiple lighting configurations. As illustrated in Fig. 6, we fixed light bulbs (L1 to L8) at 4 m high in the experiment area. In the middle between L7 and L8, apart at a distance of 4.5 m, we had a side field bulb L9. The bulbs L1 to L8 were turned on or off to represent various illumination conditions, whereas L9 was always on. Thus, the illumination becomes a controllable and independent variable. Between bulbs L1 and L6, there are glass windows, and the other sides of the experiment field are enclosed by walls. The effect of sunlight on the glass windows introduces stochastic characteristics, an uncontrollable independent variable. To determine the combined effects of lighting, we conducted experiments in the morning (07:00–10:00), afternoon (12:00–15:00), and night (18:00–23:00).

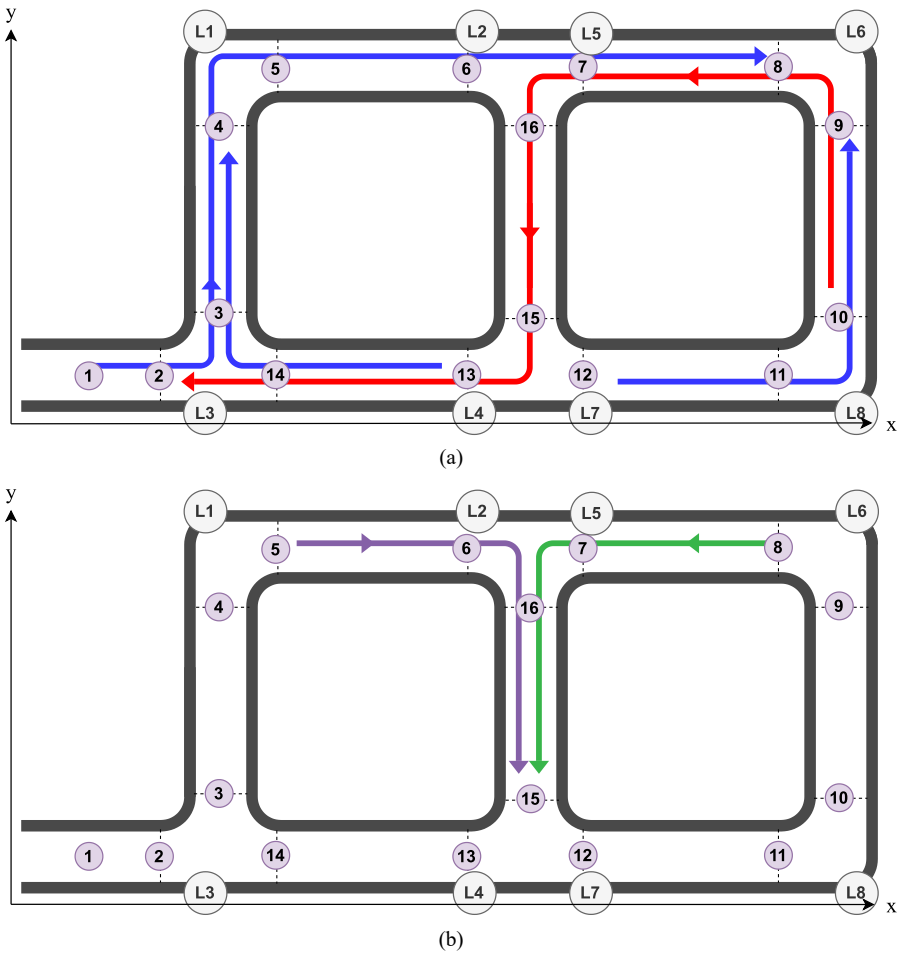


Fig. 6. Experiment route. (a) training, validation, and overall performance evaluation. (b) evaluation of turning performance

TABLE II. EXPERIMENT SCENARIO

Road sector (RS) number	Road sector trajectory	Driving Patterns								
		DP1	DP2	DP3	DP4	DP5	DP6	DP7	DP8	DP9
1	(1) to (2)	T								
2	(2) to (3)						T			
3	(3) to (4)	T								
4	(4) to (5)					T				
5	(5) to (6)	T								
6	(6) to (7)			T						
7	(7) to (8)	T								
8	(13) to (14)	T	E							
9	(14) to (3)							T		
10	(12) to (11)	T								
11	(11) to (10)				T					
12	(10) to (9)	T	E							
13	(9) to (8)				E					
14	(8) to (7)	E								
15	(7) to (16)						E			
16	(16) to (15)	E								
17	(15) to (13)									E
18	(14) to (2)			E						
19	(6) to (16)							E		

We trained the model using a particular nominal speed value to evaluate the effect of vehicle speed on performance. We then evaluated the trained model under three different speed values, i.e., the slower pace at 15 %, the nominal speed at 25 %, and the faster speed at 35 % of the throttle.

Furthermore, we evaluate the efficacy of turning under three different navigational command change instants: too early, the normal instant, and too late. The normal instant is provided when the vehicle approaches an intersection at approximately 60 cm. It is said too late if the distance is approximately 40 cm or less. Conversely, it is considered too early if the space is approximately 120 cm or more.

In contrast to our experimental scenarios, in the work by Codevilla et al., the authors did not distinguish between training and evaluation routes in their physical implementation. They collected most training data in sunny weather and evaluated their model in overcast weather conditions. They did not evaluate the effects of vehicle speed and navigation command change instant [31]. Our experimental scenario enables us to evaluate the hypothesis that the driving pattern, illumination, vehicle speed, and navigation command are independent variables. We expect these variables to be critical factors affecting the vehicle's position and yaw angle when running on the route.

Second, we present data acquisition and preprocessing. During training data collection, an expert manually operated the vehicle and directly observed the lanes while providing appropriate navigation commands following the experimental scenario via a remote controller. The command values and images were recorded synchronously. The raw image was recorded with a 640×360 pixels image dimension. The final dataset for training contains 10,652 observations. Table III to Table V summarize the statistics of the training dataset concerning the driving pattern, time, light condition, initial lateral position, and navigation command. The data was acquired in the morning, afternoon, and night with the setup lights on or off. We collected training data for three initial vehicle positions: the middle of the road, on the right, and the left sides. Meanwhile, the validation data was collected only with the initial position in the middle.

TABLE III. TRAINING DATASET: ROAD SECTOR AND DRIVING PATTERN

Road sector	Observation amount		Driving pattern (DP)
	-	%	
RS1	421	3.95	Straight following the lane (DP1)
RS3	1,736	16.30	
RS5	730	6.85	
RS7	1,103	10.35	
RS8	1,215	11.41	
RS10	848	7.96	
RS12	950	8.92	
RS6	430	4.04	Straight at the intersection on the right-hand side (DP3)
RS2	806	7.57	Turn left at the intersection (DP6)
RS11	755	7.09	Turn left at curve (DP4)
RS4	822	7.72	Turn right at curve (DP5)
RS9	836	7.85	Turn right at the intersection (DP7)
Total	10,652	100.00	

TABLE IV. TRAINING DATASET: TIME AND LIGHT CONDITIONS

Time of day	Observation amount					
	On		Off		Total	
	Ori.	Aug.	Ori.	Aug.	-	%
Morning	1,906	185	1,624	158	**3,873**	**36.4**
Afternoon	1,670	162	1,482	144	**3,458**	**32.4**
Night	1,492	145	1,535	149	**3,321**	**31.2**
Total	**5,086**	**492**	**4,641**	**451**	**10,652**	**100.0**

TABLE V. TRAINING DATASET: INITIAL LAT. POSITION AND NAV. COMMAND

Initial lat. position	Observation amount		Nav. command	Observation amount	
	-	%		-	%
Middle	3.305	31.03	Straight	8.067	75.73
Left	3.640	34.17	Left	1.325	12.44
Right	3.707	34.80	Right	1.260	11.83
Total	10.652	100.00	Total	10.652	100.00

Regarding the amount of driving pattern observation, DP1 encompasses 65.74%. Other patterns occupy from 4.04% to 7.85%. We performed image augmentation to provide perception robustness against variations in lighting conditions. The observation amount of original (Ori.) and augmented (Aug.) images are listed in Table IV. The observation amount of each augmented image is around 9.7% of the corresponding original image.

The original image dataset was preprocessed by cropping the region of interest, resizing, converting the color system, and augmenting it. Fig. 7 depicts three samples captured at different road sectors at night with the bulbs switched on. Fig. 8 shows the preprocessing results of the raw image in Fig. 7 (a). After cropping the upper section and resizing, the image's resolution became 220×100 pixels (Fig. 8(a)). The RGB image was then converted into a YUV image (Fig. 8(b)). The image in the YUV color system is more efficiently processed by a digital computer [67]. The image was then randomly augmented by adding Gaussian noise (Fig. 8 (c)).

Next, we describe the training and validation processes. We acquired training and validation data using the nominal speed and normal navigation command change instant. To accommodate a systematic analysis of critical factors, we associated the training dataset with three different models based on navigation command type as follows: model 1, which refers to the dataset recorded when the vehicle is moving along road sectors 1, 3, 4, 5, 6, 7, 8, 10, 11, and 12; model 2, which refers to those when the vehicle is moving along RS2; and model 3, which refers to RS9. We set the mini-batch size to 64, a learning rate decay of 0.001, and the epoch number to 65. The IL model was trained using the Adam optimizer [68]. We used the mean absolute error as the loss function. Given mini-batch size m_b and predicted and ground truth steering angles s_p and s_{gt}, we define the loss function $L(s_p, s_{gt})$ per mini-batch in (8).

$$L(s_p, s_{gt}) = \frac{1}{m_b}\sum_{i=1}^{m_b}|s_{pi} - s_{gti}| \quad (8)$$

The training process took approximately 45 minutes using a processor Intel® Core TM i7-7700HQ CPU @ 2.80 GHz (8 CPUs). It was equipped with 24 GB RAM, GTX 1050 GPU, CUDA V11.2, and Python 3.9.

We conducted a validation by running the vehicle through the validation road sectors six times. For each run, the number of observations was 225, 45, and 44 for model 1, model 2, and model 3, respectively. Fig. 9 plots the model loss of each epoch obtained from the training and validation phases. The model loss of the validation phase fluctuates relatively, particularly for the right-turn navigation command. It is likely because the data in the validation dataset has never been seen in the training dataset.

The final process is performance evaluation, which includes overall performance and turning performance at intersections. Here, performance metrics are formulated. The autonomous controller performance was evaluated by comparing the vehicle position with the road sector coordinates. We defined reward and penalty values to evaluate the overall performance throughout all the road sectors.

1) When the vehicle stays inside the road, it is said to be successful and is given a reward of 1.
2) When one of the front tires slightly gets out of the lane marking yet manages to get back to the track and continue moving autonomously, it is said to return safely. We assign zero points for this case.
3) It fails when the vehicle exits lane markings and does not return to the lane. We assign -1 point for this case.

We conducted the experiments three times for each evaluation road sector under specific controllable conditions, including speed, navigation command instant, and light on/off. The experiment was also under the effect of random light from the outside environment. The reward or punishment value P_i is summed to obtain the overall performance indicator I_{p1} throughout the evaluation road sectors, as given by (9). n_1 denotes the total number of observations.

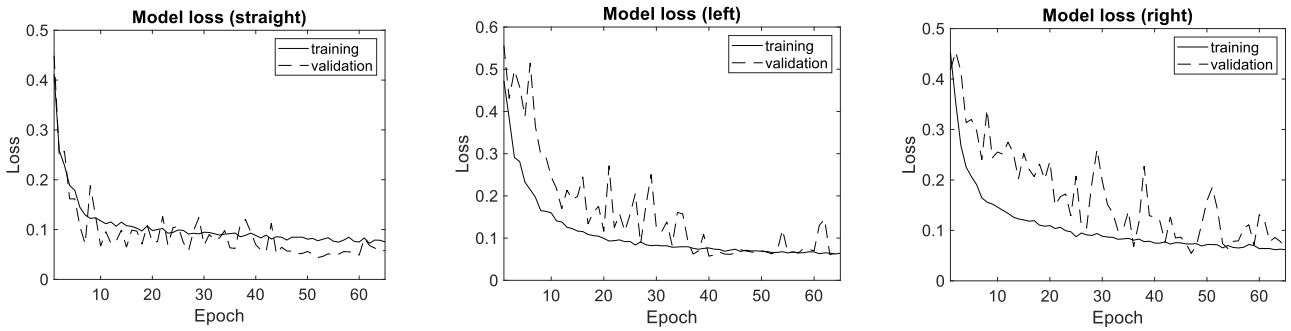
$$I_{p1} = \frac{1}{n_1}\sum_{i=1}^{n_1}(P_i) \quad (9)$$

(a) (b) (c)

Fig. 7. Example of raw images, (a) At road sector 3, (b) At road sector 6, (c) At road sector 4

(a) (b) (c)

Fig. 8. Example of pre-processed images, (a) Cropped and resized image, (b) YUV image, (c) Augmented image



(a) (b) (c)

Fig. 9. Model losses during training and validation phases, (a) model 1, (b) model 2, (c) model 3

Moreover, we introduced a more detailed measurement to evaluate the turning performance of the vehicle in terms of the location and yaw angle at a specific intersection. Before starting an evaluation experiment, we placed the x-axis of the vehicle parallel to the road lanes. We initially positioned the vehicle in the middle of the road and measured its position as it moved through the intersection.

The turning performance indicator in terms of location is given by (10), where (x_{ri}, y_{ri}) and (x_i, y_i) represent the reference and vehicle locations at each time step i, respectively. n_2 denotes the total number of observations.

$$I_{p2} = \sqrt{\frac{1}{n_2}\sum_{i=1}^{n_2}(x_i - x_{ri})^2 + (y_i - y_{ri})^2} \quad (10)$$

We first set the vehicle's initial yaw angle to $\varphi_v(0)$. Then, the yaw angle $\varphi_v(t)$ was measured relative to the initial yaw angle. We evaluated the turning performance indicator in terms of the yaw angle at the intersections by observing the plot of the yaw angle and the qualitative description.

III. RESULT AND DISCUSSION

First, we discuss the results of overall performance evaluation experiments. Table VI summarizes the overall performance of the CIL implementation in the experimental scenario. Effects of critical factors on performance are investigated. They include driving patterns (DPs), lighting conditions, vehicle speed, and navigation command change instant. M, A, and N represent morning, afternoon, and night. L1 and L0 denote the field bulbs on and off. S1, S2, and S3 refer to the vehicle speed slower than the nominal speed, the nominal speed, and faster than the nominal speed. C1, C2, and C3 refer to the navigation command change instant that is too late, timely, and too early compared to the normal instant. They apply only to RS15 and RS17. For other road sectors, we use C0.

Recall that statistics of the training dataset are displayed in Table III, Table IV, and Table V. Statistics of the evaluation dataset are explained in the paragraph below (8), and Table VI shows that each evaluation experiment was conducted three times.

The evaluation results demonstrate that the vehicle could autonomously drive successfully through road sectors 8, 12, 13, 14, 16, and 18 in all experiments. However, it failed five times when traversing the new intersections at road sectors 15 and 17. Except for road sector 8, they are new road sectors for the vehicle, as they have never been traveled during training. Nevertheless, the vehicle experienced the same driving patterns at other locations during the training session, except road sector 17 (RS17) with DP9.

Even though the vehicle had never been trained to pass through DP9 at RS17, during the evaluation, out of 162 experiments, it slightly deviated from the route three times and returned. The conditions occurred under the following scenarios: ML1-S3C1, AL0-S3C2, and NL1-S3C1.

During training, the vehicle had never traveled through RS15. However, it had been trained at the same driving pattern in RS2 even under different lighting conditions, as RS15 is close to glass windows. During the evaluation, out of 162 trials, the vehicle escaped from the lane once but returned under AL1-S3C1 and failed once under the conditions of NL1-S3C1.

From the overall performance evaluation, the AV failed once when it turned left at a new intersection (RS15) near the glass windows. It occurred under the following specific conditions (NL1-S3C1): at night, with the field's bulbs switched on, at a faster speed, and with a navigation command change moment that occurred too late. Under the specific conditions (L1-S3C1), it got out slightly and returned to the lane twice when turning right at a new intersection with an untrained driving pattern (RS17) and once when turning

left at a new intersection (RS15). It also got out lightly and returned to the lane once when turning right at the new intersection with an untrained driving pattern (RS17) under the specific conditions (AL0-S3C2): in the afternoon with the field's bulbs off, faster speed, and normal navigation command change instant.

It is worthwhile to note that evaluation in the morning and night with the bulbs off under different conditions yielded 108 out of 108 successful autonomous driving, respectively. It can be concluded that the CIL model produced a success rate of 99.1% from 648 experiments under different conditions of driving patterns, lighting, vehicle speeds, and navigation command change instants.

TABLE VI. OVERALL PERFORMANCE EVALUATION RESULT

No	Autonomous Driving Conditions: Time | Light | - | Speed | Command Instant	The Road Sector, which is characterized by the Driving Pattern (Table II)								Total
		12	13	14	15	16	17	8	18	
1	ML1-S1C1				3		3			6
2	ML1-S1C2 or ML1-S1C0	3	3	3	3	3	3	3	3	24
3	ML1-S1C3				3		3			6
4	ML1-S2C1				3		3			6
5	ML1-S2C2 or ML1-S2C0	3	3	3	3	3	3	3	3	24
6	ML1-S2C3				3		3			6
7	ML1-S3C1				3		2			5
8	ML1-S3C2 or ML1-S2C0	3	3	3	3	3	3	3	3	24
9	ML1-S3C3				3		3			6
10	ML0-S1C1				3		3			6
11	ML0-S1C2 or ML0-S1C0	3	3	3	3	3	3	3	3	24
12	ML0-S1C3				3		3			6
13	ML0-S2C1				3		3			6
14	ML0-S2C2 or ML0-S2C0	3	3	3	3	3	3	3	3	24
15	ML0-S2C3				3		3			6
16	ML0-S3C1				3		3			6
17	ML0-S3C2 or ML0-S3C0	3	3	3	3	3	3	3	3	24
18	ML0-S3C3				3		3			6
19	AL1-S1C1				3		3			6
20	AL1-S1C2 or AL1-S1C0	3	3	3	3	3	3	3	3	24
21	AL1-S1C3				3		3			6
22	AL1-S2C1				3		3			6
23	AL1-S2C2 or AL1-S2C0	3	3	3	3	3	3	3	3	24
24	AL1-S2C3				3		3			6
25	AL1-S3C1				2		3			5
26	AL1-S3C2 or AL1-S3C0	3	3	3	3	3	3	3	3	24
27	AL1-S3C3				3		3			6
28	AL0-S1C1				3		3			6
29	AL0-S1C2 or AL0-S1C0	3	3	3	3	3	3	3	3	24
30	AL0-S1C3				3		3			6
31	AL0-S2C1				3		3			6
32	AL0-S2C2 or AL0-S2C0	3	3	3	3	3	3	3	3	24
33	AL0-S2C3				3		3			6
34	AL0-S3C1				3		3			6
35	AL0-S3C2 or AL0-S3C0	3	3	3	3	3	2	3	3	23
36	AL0-S3C3				3		3			6
37	NL1-S1C1				3		3			6
38	NL1-S1C2 or NL1-S1C0	3	3	3	3	3	3	3	3	24
39	NL1-S1C3				3		3			6
40	NL1-S2C1				3		3			6
41	NL1-S2C2 or NL1-S2C0	3	3	3	3	3	3	3	3	24
42	NL1-S2C3				3		3			6
43	NL1-S3C1				1		2			3
44	NL1-S3C2 or NL1-S3C0	3	3	3	3	3	3	3	3	24
45	NL1-S3C3				3		3			6
46	NL0-S1C1				3		3			6
47	NL0-S1C2 or NL0-S1C0	3	3	3	3	3	3	3	3	24
48	NL0-S1C3				3		3			6
49	NL0-S2C1				3		3			6
50	NL0-S2C2 or NL0-S2C0	3	3	3	3	3	3	3	3	24
51	NL0-S2C3				3		3			6
52	NL0-S3C1				3		3			6
53	NL0-S3C2 or NL0-S3C2	3	3	3	3	3	3	3	3	24
54	NL0-S3C3				3		3			6

Second, since the unsuccessful autonomous driving during the evaluation session happened at new intersections, we discuss further in more detail the results of turning performance at new intersections. We measured the turning performance when the AV turned right at the intersection along RS19 and turned left at the intersection along RS15. The AV positions, yaw angles, yaw rates, and steering angles are depicted in Fig. 10 and Fig. 11. The reference trajectory was the middle line of the turning curve.

Fig. 10 depicts the experimental results obtained from three different navigation command change instants under the same conditions: nominal speed, night, and the experiment field bulbs were switched on. The solid red line in Fig. 10(a) denotes the reference trajectory. The dotted, dotted-dashed, and dashed lines represent the trajectories when the navigation command change instant is too late, normal, and too early, respectively. The corresponding yaw angles, yaw rates, and steering angles are plotted in Fig. 10(b), Fig. 10(c), and Fig. 10(d), respectively. These conditions correspond to NL1-S2C1, NL1-S2C2, and NL1-S2C3 in Table VI. We observe similar dynamical patterns among the experimental results – the AV locations remained close to the references without undergoing drastic change.

Fig. 11 plots the experimental results obtained from three different vehicle speeds under the same conditions: normal navigation command instant, at night, and the experiment field bulbs were on. The solid red line in Fig. 11(a) denotes the reference trajectory. The dotted, dotted-dashed, and dashed lines represent the trajectories when the vehicle speed is faster, nominal, and slower, respectively. In Fig. 11(b), Fig. 11(c), and Fig. 11(d), the dotted, dotted-dashed, and dashed lines represent the yaw angles, yaw rates, and steering angles of the corresponding conditions. These conditions correspond to NL1-S3C2, NL1-S2C2, and NL1-S1C2 in Table VI. It can be seen from Fig. 11 that the vehicle locations remained close to the references. However, the vehicle experienced overshoot and undershoot in the location and yaw angle responses when it was faster than the nominal speed. The steering angle rapidly increased from the straight moving-steering angle and decreased back to the straight moving-steering angle to maintain the AV inside the lane. The steering angle moved to the opposite angle to compensate for the overshoot before returning to the straight-moving steering angle.

During turning performance evaluation experiments, the AV never escaped from the road; in other words, it achieved a 100% success rate. To deepen our evaluation, we calculated quantitative turning performance indicators regarding location (Table VII). During the right turn at RS19, the yaw angles gradually changed from approximately 0° to -90° between 3.5 s and 4.5 s. When turning left at RS15 at nominal and slower speeds, the vehicle did not overshoot or undershoot. The yaw angles changed from approximately 180° to 270° within 4 s at the nominal speed and 9 s at the slower speed.

TABLE VII. TURNING PERFORMANCE INDICATOR IN TERMS OF LOCATION

Description of Performance Indicator I_{p2}	Value (cm)
Under too late nav command change instant	4.4
Under normal nav command change instant	6.5
Under too early nav command change instant	6.0
Under a faster speed	7.2
Under nominal speed	5
Under a slower speed	5.5

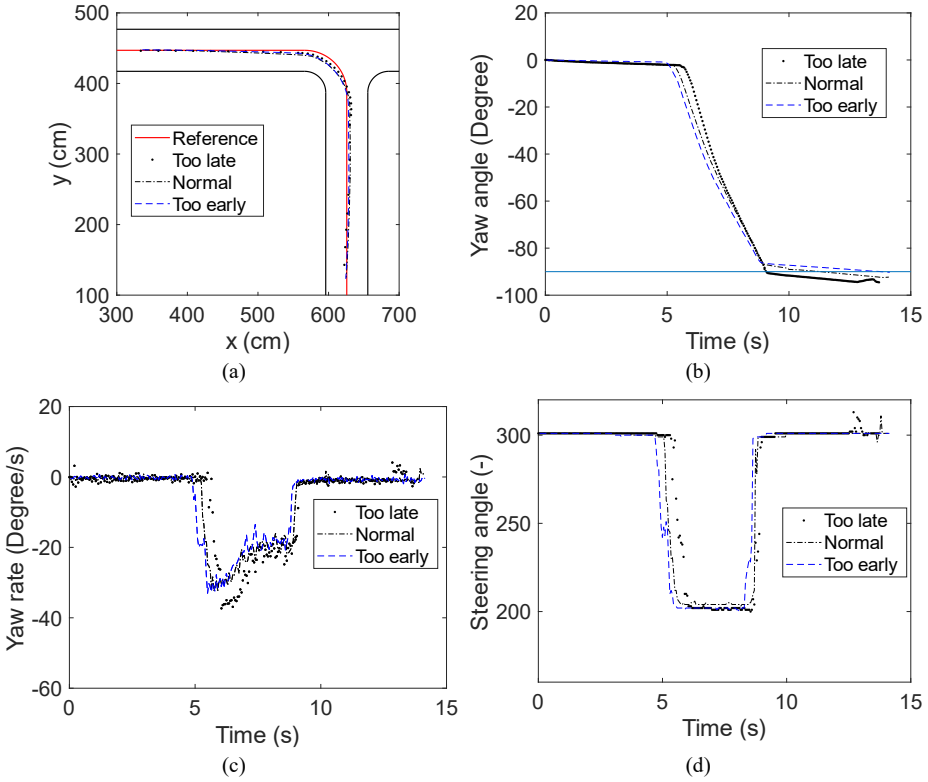


Fig. 10. (a) Locations, (b) yaw angle, (c) yaw rate, and (d) steering angle along RS19 under three navigation command change instants

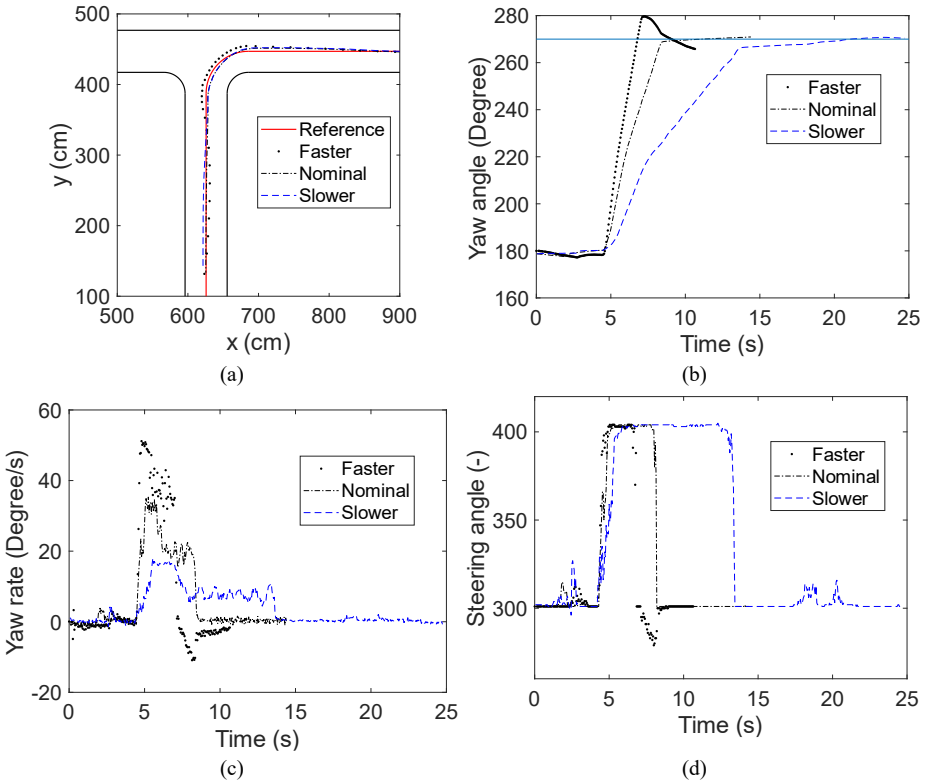


Fig. 11. (a) Locations, (b) yaw angle, (c) yaw rate, and (d) steering angle along RS15 under three different vehicle speeds

The main findings of our study are summarized as follows:

1) The AV could not maintain itself inside the road five times out of 648 experiments when it turned at new intersections.
2) The AV could not maintain itself inside the road at new intersections because of the adverse effects of dazzling light reflection, faster speed, and too-late command change instant.
3) The AV could not maintain itself inside the road at new intersections because of the untrained driving pattern combined with faster speed.

We use 73 references relevant to this work on autonomous driving of ground vehicles from several databases, including Science Direct (three articles and one book), IEEE Xplore (30 articles), Springer (12 articles), MDPI (seven articles), Wiley (two articles), Frontiers (one article), Scopus (11 articles), and others (six articles). Fifty-six articles were published in journals, ten articles in proceedings, and one book was published by Elsevier. This limited number of references indicates that the research topic of autonomous driving based on the end-to-end approach is still an infant. Only two articles on end-to-end autonomous driving reported physical experimental results in urban driving [31], [65]. In [31], the authors did not distinguish between training and evaluation routes, nor did they systematically evaluate the effect of lighting. Moreover, they did not consider the effects of vehicle speed and navigation command change instant. In [65], the testing environment contained uncontrollable time-dependent factors, e.g., weather, lighting, and road users. Also, the drivers selected the experiment routes randomly and controlled both steering angle and vehicle speed. These uncontrollable conditions, route randomness, and time-varying speed made identifying key factors more difficult.

Time-constant speed in our experiments may be a limitation of this study and, at the same time, becomes the strength since it enables systematical analysis of speed effects on performance by comparing three different time-constant speed values: low speed, nominal speed, and fast speed.

This study's second and third main findings contribute to AD by suggesting avenues for enhancing AD technology. For example, the second main finding motivated us to employ an if-then logic to avoid a turning failure, i.e., if the controller identifies dazzling light reflections before turning, drive slower, and do not change the navigation command too late. The third main finding stimulated us to employ a second if-then logic: if the autonomous controller identifies a new intersection with an untrained driving pattern, then do not drive at a faster speed. However, these two logics require more accurate and robust object identification capabilities. Some researchers proposed multimodality fusion and training data augmentation to enhance perception capability, target recognition and tracking, and semantic segmentation [35], [69]–[73].

The original CIL model was modified using multimodality fusion in the CARLA simulator, including color images (RGB)-stereo depth fusion [35] and RGB-LiDAR point cloud fusion [69]. A multimodality fusion from camera and Radar was developed to process a real-world dataset for target tracking based on a switchable dual-level

long short-term memory (LSTM) network [70]. They validated the method in three illumination conditions: day, dusk, and night modes. However, they did not report running time.

A homography augmentation using the DeepLabv3+ network from stereo-images was developed and proven to outperform six state-of-the-art Deep CNNs regarding accuracy, precision, recall, and runtime [71]. This method is potentially developed for collision-free space detection algorithms for autonomous driving.

CIL model implementation in real-world urban driving also necessitates efficient running time. Besides multimodal fusion and training data augmentation, exploring more powerful preprocessing methods combined with multi-route networks to answer this challenge is also interesting.

IV. CONCLUSION

Two failure conditions decreased the success rate to 99% out of 648 experiments. One is turning at a new intersection, coupled with the combination of three factors: dazzling light reflection, faster speed, and too-late command change instant. The other is turning at a new intersection with an untrained driving pattern coupled with faster speed.

Under controllable conditions, we need to ensure a success rate of 100%. Based on our knowledge obtained in this study, we can embed the following two if-then logics into the autonomous controller to avoid any turning failure: if the controller identifies dazzling light reflections before turning, then drive slower and do not change the navigation command too late; if the autonomous controller identifies a new intersection with an untrained driving pattern, then do not drive at a faster speed.

The two logics require more accurate and robust object identification capabilities. Implementing the autonomous controller in real time requires efficient running time. For future work, we intend to develop a model incorporating multimodal fusion, training data augmentation, powerful preprocessing, and multi-route networks.

One limitation of this study is that the dazzling light reflection happened accidentally during the experiment. To develop a robust CIL model against dazzling light reflections, we need to be able to reconstruct several dazzling light reflections and systematically evaluate the model's performance against such dazzling light reflections. The other limitation is that no obstacles exist in the road sectors. A natural extension of this study is to develop a CIL model that can avoid obstacles.

ACKNOWLEDGMENT

This research was done prior to Nelson Changgraini joining The University of Tokyo. The authors thank O. Mahendra, R. D. Firmansyah, and A. Nugroho, who helped build the measurement instrument on the 1/10 scale AV.

REFERENCES

[1] Ó. Pérez-Gil *et al*., "Deep reinforcement learning based control for Autonomous Vehicles in CARLA," *Multimed. Tools Appl.*, vol. 81, no. 3, pp. 3553–3576, 2022.

[2] J. Laconte, A. Kasmi, R. Aufrère, M. Vaidis, and R. Chapuis, "A Survey of Localization Methods for Autonomous Vehicles in Highway Scenarios," *Sensors (Basel)*, vol. 22, no. 1, p. 247, 2021.

[3] L. Fridman *et al*., "MIT Advanced Vehicle Technology Study: Large-Scale Naturalistic Driving Study of Driver Behavior and Interaction With Automation," *IEEE Access*, vol. 7, pp. 102021–102038, 2019.

[4] Y.-B. Chang, C. Tsai, C.-H. Lin, and P. Chen, "Real-Time Semantic Segmentation with Dual Encoder and Self-Attention Mechanism for Autonomous Driving," *Sensors (Basel)*, vol. 21, no. 23, p. 8072, 2021.

[5] E. Horváth, C. Pozna, and M. Unger, "Real-time lidar-based urban road and sidewalk detection for autonomous vehicles," *Sensors*, vol. 22, no. 1, 2022.

[6] C. Sun, X. Zhang, Q. Zhou, and Y. Tian, "A Model Predictive Controller With Switched Tracking Error for Autonomous Vehicle Path Tracking," *IEEE Access*, vol. 7, pp. 53103–53114, 2019.

[7] S. Teng *et al*., "Motion Planning for Autonomous Driving: The State of the Art and Future Perspectives," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 6, pp. 3692–3711, 2023.

[8] M. Chghaf, S. Rodriguez, and A. El Ouardi, "Camera, LiDAR and Multi-modal SLAM Systems for Autonomous Ground Vehicles: A Survey," *J. Intell. Robot Syst.*, vol. 105, no. 1, p. 2, 2022.

[9] C. Liu, H. Liu, L. Han, and C. Xiang, "New Integrated Multi-Algorithm Fusion Localization and Trajectory Tracking Framework of Autonomous Vehicles under Extreme Conditions with Non-Gaussian Noises," *International Journal of Automotive Technology*, vol. 24, no. 1, pp. 259–272, 2023.

[10] Q. Song, K. Tan, P. Runeson, and S. Persson, "Critical scenario identification for realistic testing of autonomous driving systems," *Software Quality Journal*, vol. 31, no. 2, pp. 441–469, 2023.

[11] C. Gómez-Huélamo *et al*., "360o real-time and power-efficient 3D DAMOT for autonomous driving applications," *Multimed. Tools Appl.*, vol. 81, no. 19, pp. 26915–26940, 2022.

[12] F. Tener and J. Lanir, "Investigating intervention road scenarios for teleoperation of autonomous vehicles," *Multimed. Tools Appl.*, pp. 1-17, 2023.

[13] E. O. Appiah and S. Mensah, "Object detection in adverse weather condition for autonomous vehicles," *Multimed. Tools Appl.*, pp. 1-27, 2023.

[14] Md. M. Rana and K. Hossain, "Connected and Autonomous Vehicles and Infrastructures: A Literature Review," *International Journal of Pavement Research and Technology*, vol. 16, no. 2, pp. 264–284, 2023.

[15] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "DeepDriving: Learning Affordance for Direct Perception in Autonomous Driving," *Proceedings of the IEEE international conference on computer vision*, pp. 2722-2730, 2015.

[16] D. A. Pomerleau, "ALVINN: An Autonomous Land Vehicle in a Neural Network," in *Advances in Neural Information Processing Systems*, vol. 1, 1988.

[17] D. A. Pomerleau, "Efficient Training of Artificial Neural Networks for Autonomous Navigation," *Neural Comput.*, vol. 3, no. 1, pp. 88–97, 1991.

[18] M. Al-Qizwini, I. Barjasteh, H. Al-Qassab, and H. Radha, "Deep learning algorithm for autonomous driving using GoogLeNet," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 89–96, 2017.

[19] Q. Liu, S. Song, H. Hu, T. Huang, C. Li, and Q. Zhu, "Extended model predictive control scheme for smooth path following of autonomous vehicles," *Frontiers of Mechanical Engineering*, vol. 17, no. 1, p. 4, 2022.

[20] U. Muller, J. Ben, E. Cosatto, B. Fleep, and Y. LeCun, "Autonomous off-road vehicle control using end-to-end learning," *Courant Institute/CBLL, Arlington, VA, USA, Tech. Rep. DARPA-IPTO Final technical Report*, vol. 458, 2004.

[21] U. Muller, J. Ben, E. Cosatto, B. Flepp, and Y. Cun, "Off-Road Obstacle Avoidance through End-to-End Learning," in *Advances in Neural Information Processing Systems*, vol. 18, 2005.

[22] M. Bojarski *et al*., "End to End Learning for Self-Driving Cars," *arXiv:1604.07316*, 2016.

[23] M. Bojarski *et al*., "Explaining how a deep neural network trained with end-to-end learning steers a car," *arXiv preprint arXiv:1704.07911*, 2017.

[24] L. A. Curiel-Ramirez *et al*., "End-to-End Automated Guided Modular Vehicle," *Applied Sciences*, vol. 10, no. 12, p. 4400, 2020.

[25] Y. Pan *et al.*, "Agile Autonomous Driving using End-to-End Deep Imitation Learning," *arXiv:1709.07174*, 2017.

[26] S. Nozari, A. Krayani, P. Marin-Plaza, L. Marcenaro, D. M. Gomez, and C. Regazzoni, "Active Inference Integrated With Imitation Learning for Autonomous Driving," *IEEE Access*, vol. 10, pp. 49738–49756, 2022.

[27] Y. Pan *et al*., "Imitation learning for agile autonomous driving," *Int J Rob Res*, vol. 39, no. 2–3, pp. 286–302, 2020.

[28] S. Teng, L. Chen, Y. Ai, Y. Zhou, Z. Xuanyuan, and X. Hu, "Hierarchical Interpretable Imitation Learning for End-to-End Autonomous Driving," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 1, pp. 673–683, 2023.

[29] H. Tian, C. Wei, C. Jiang, Z. Li, and J. Hu, "Personalized Lane Change Planning and Control By Imitation Learning From Drivers," *IEEE Transactions on Industrial Electronics*, vol. 70, no. 4, pp. 3995–4006, 2023.

[30] J. Ying and Y. Feng, "Full Vehicle Trajectory Planning Model for Urban Traffic Control Based on Imitation Learning," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2676, no. 7, pp. 186–198, 2022.

[31] F. Codevilla, M. Muller, A. Lopez, V. Koltun, and A. Dosovitskiy, "End-to-End Driving Via Conditional Imitation Learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4693–4700, 2018.

[32] A. Sauer, N. Savinov, and A. Geiger, "Conditional Affordance Learning for Driving in Urban Environments," in *Proceedings of The 2nd Conference on Robot Learning*, vol. 87, pp. 237–252, 2018.

[33] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An Open Urban Driving Simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, vol. 78, pp. 1–16, 2017.

[34] D. Chen, B. Zhou, V. Koltun, and P. Krähenbühl, "Learning by Cheating," in *Proceedings of the Conference on Robot Learning*, vol. 100, pp. 66–75, 2020.

[35] Y. Xiao, F. Codevilla, A. Gurram, O. Urfalioglu, and A. M. Lopez, "Multimodal End-to-End Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 537–547, 2022.

[36] P. S. Chib and P. Singh, "Recent Advancements in End-to-End Autonomous Driving using Deep Learning: A Survey," *IEEE Transactions on Intelligent Vehicles*, pp. 1–18, 2023.

[37] D. Coelho and M. Oliveira, "A Review of End-to-End Autonomous Driving in Urban Environments," *IEEE Access*, vol. 10, pp. 75296–75311, 2022.

[38] A. Tampuu, T. Matiisen, M. Semikin, D. Fishman, and N. Muhammad, "A Survey of End-to-End Driving: Architectures and Training Methods," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 4, pp. 1364–1384, 2022.

[39] A. Amini, I. Gilitschenski, J. Phillips, J. Moseyko, R. Banerjee, S. Karaman, and D. Rus, "Learning Robust Control Policies for End-to-End Autonomous Driving From Data-Driven Simulation," *IEEE Robot Autom. Lett.*, vol. 5, no. 2, pp. 1143–1150, 2020.

[40] X. Wang, Z. Ning, S. Guo, and L. Wang, "Imitation Learning Enabled Task Scheduling for Online Vehicular Edge Computing," *IEEE Trans. Mob. Comput.*, vol. 21, no. 2, pp. 598–611, 2022.

[41] S. Kuutti, R. Bowden, Y. Jin, P. Barber, and S. Fallah, "A Survey of Deep Learning Applications to Autonomous Vehicle Control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 712–733, 2021.

[42] W. Wang, L. Wang, C. Zhang, C. Liu, and L. Sun, "Social Interactions for Autonomous Driving: A Review and Perspectives," *Foundations and Trends® in Robotics*, vol. 10, no. 3–4, pp. 198–376, 2022.

[43] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A Survey of Autonomous Driving: Common Practices and Emerging Technologies," *IEEE Access*, vol. 8, pp. 58443–58469, 2020.

[44] U. M. Gidado, H. Chiroma, N. Aljojo, S. Abubakar, S. I. Popoola, and M. A. Al-Garadi, "A survey on deep learning for steering angle prediction in autonomous vehicles," *IEEE Access*, vol. 8, pp. 163797–163817, 2020.

[45] A. O. Ly and M. Akhloufi, "Learning to Drive by Imitation: An Overview of Deep Behavior Cloning Methods," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 2, pp. 195–209, 2021.

[46] É. Zablocki, H. Ben-Younes, P. Pérez, and M. Cord, "Explainability of Deep Vision-Based Autonomous Driving Systems: Review and Challenges," *Int. J. Comput. Vis.*, vol. 130, no. 10, pp. 2425–2452, 2022.

[47] L. Arras, A. Osman, and W. Samek, "CLEVR-XAI: A benchmark dataset for the ground truth evaluation of neural network explanations," *Information Fusion*, vol. 81, pp. 14–40, 2022.

[48] M. Borg *et al*., "Safely Entering the Deep: A Review of Verification and Validation for Machine Learning and a Challenge Elicitation in the Automotive Industry," *Journal of Automotive Software Engineering*, vol. 1, no. 1, p. 1, 2019.

[49] Z. Zhang, R. Tian, R. Sherony, J. Domeyer, and Z. Ding, "Attention-Based Interrelation Modeling for Explainable Automated Driving," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 2, pp. 1564–1573, Feb. 2023.

[50] P. P. Angelov, E. A. Soares, R. Jiang, N. I. Arnold, and P. M. Atkinson, "Explainable artificial intelligence: an analytical review," *WIREs Data Mining and Knowledge Discovery*, vol. 11, no. 5, 2021.

[51] H. Mankodiya, D. Jadav, R. Gupta, S. Tanwar, W.-C. Hong, and R. Sharma, "OD-XAI: Explainable AI-Based Semantic Object Detection for Autonomous Vehicles," *Applied Sciences*, vol. 12, no. 11, p. 5310, 2022.

[52] M. P. S. Lorente, E. M. Lopez, L. A. Florez, A. L. Espino, J. A. I. Martínez, and A. S. de Miguel, "Explaining Deep Learning-Based Driver Models," *Applied Sciences*, vol. 11, no. 8, p. 3321, 2021.

[53] J. Kim, A. Rohrbach, Z. Akata, S. Moon, T. Misu, Y. Chen, T. Darrell, and J. Canny, "Toward explainable and advisable model for self-driving cars," *Applied AI Letters*, vol. 2, no. 4, 2021.

[54] X. Bai, X. Wang, X. Liu, Q. Liu, J. Song, N. Sebe, and B. Kim, "Explainable deep learning for efficient and robust pattern recognition: A survey of recent developments," *Pattern Recognit.*, vol. 120, p. 108102, 2021.

[55] P. Angelov and E. Soares, "Towards explainable deep neural networks (xDNN)," *Neural Networks*, vol. 130, pp. 185–194, 2020.

[56] K. Muhammad, A. Ullah, J. Lloret, J. Del Ser, and V. H. C. de Albuquerque, "Deep learning for safe autonomous driving: Current challenges and future directions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4316–4336, 2020.

[57] V. Belle and I. Papantonis, "Principles and practice of explainable machine learning," *Front Big Data*, p. 39, 2021.

[58] A. Pereira and C. Thomas, "Challenges of machine learning applied to safety-critical cyber-physical systems," *Mach. Learn. Knowl. Extr.*, vol. 2, no. 4, pp. 579–602, 2020.

[59] Z. Huang, C. Lv, Y. Xing, and J. Wu, "Multi-Modal Sensor Fusion-Based Deep Neural Network for End-to-End Autonomous Driving With Scene Understanding," *IEEE Sens. J.*, vol. 21, no. 10, pp. 11781–11790, 2021.

[60] L. Chen, X. Hu, B. Tang, and Y. Cheng, "Conditional DQN-Based Motion Planning With Fuzzy Logic for Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 4, pp. 2966–2977, 2022.

[61] J. Chen, S. E. Li, and M. Tomizuka, "Interpretable End-to-End Urban Autonomous Driving With Latent Deep Reinforcement Learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 5068–5078, 2022.

[62] M. Ahmed, A. Abobakr, C. P. Lim, and S. Nahavandi, "Policy-Based Reinforcement Learning for Training Autonomous Driving Agents in Urban Areas With Affordance Learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 12562–12571, 2022.

[63] C. Huang, R. Zhang, M. Ouyang, P. Wei, J. Lin, J. Su, and L. Lin, "Deductive Reinforcement Learning for Visual Autonomous Urban Driving Navigation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 12, pp. 5379–5391, 2021.

[64] L. Le Mero, D. Yi, M. Dianati, and A. Mouzakitis, "A Survey on Imitation Learning Techniques for End-to-End Autonomous Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14128–14147, 2022.

[65] J. Hawke *et al*., "Urban Driving with Conditional Imitation Learning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 251–257, 2020.

[66] M. Abe. *Vehicle Handling Dynamics: Theory and Application*. Butterworth-Heinemann. 2015.

[67] M. Podpora, G. P. Korbas, and A. Kawala-Janik, "YUV vs RGB-Choosing a Color Space for Human-Machine Interaction.," in *FedCSIS (Position Papers)*, pp. 29–34, 2014.

[68] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[69] H. M. Eraqi, M. N. Moustafa, and J. Honer, "Dynamic Conditional Imitation Learning for Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 22988–23001, 2022.

[70] M. Cao, R. Wang, N. Chen, and J. Wang, "A Learning-Based Vehicle Trajectory-Tracking Approach for Autonomous Vehicles With LiDAR Failure Under Various Lighting Conditions," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 2, pp. 1011–1022, 2022.

[71] R. Fan *et al*., "Learning Collision-Free Space Detection From Stereo Images: Homography Matrix Brings Better Data Augmentation," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 1, pp. 225–233, 2022.

[72] K. Bayoudh, R. Knani, F. Hamdaoui, and A. Mtibaa, "A survey on deep multimodal learning for computer vision: advances, trends, applications, and datasets," *Vis. Comput.*, vol. 38, no. 8, pp. 2939–2970, 2022.

[73] J. Kaur and W. Singh, "A systematic review of object detection from images using deep learning," *Multimed. Tools Appl.*, vol. 83, no. 4, pp. 12253–12338, 2024.