A New Open-Source Off-road Environment for Benchmark Generalization of Autonomous Driving

ISAAC HAN, DONG-HYEOK PARK, and KYUNG-JOONG KIM
School of Integrated Technology, Gwangju Institute of Science and Technology, Gwangju 61005 South Korea (e-mail: issac7778@gm.gist.ac.kr)
Corresponding author: Kyung-Joong Kim (kjkim@gist.ac.kr)

This work was supported by the Defense Acquisition Program Administration and Agency for Defense Development under contract UD180026RD and the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (2021R1A4A1030075)

ABSTRACT Recently, deep neural networks have greatly improved autonomous driving. However, as a great deal of training data is required, most studies have employed simulators. Generalization of such driving is key in terms of safety. The simulated environments feature only small variations in favorable conditions and thus cannot be used for benchmarking. Therefore, we developed a new open-source (OpenAI Gym-like) off-road environment featuring differently structured forests, plateaus, deserts, and snowfields. The dynamic topographical structures make the off-road environment a very challenging generalization problem. Our off-road environment can precisely evaluate autonomous driving in terms of generalization. Additionally, we proposed an evaluation method based on the success rate of driving tasks, enabling effective driving ability measurement. Furthermore, we evaluate the performance of existing end-to-end driving methods in our off-road environment. The results show that the end-to-end driving methods lack generalization ability and fail to generalize to unseen environments. Our off-road environment can help autonomous driving researchers develop a better, generalizable driving system. Unreal engine-level assets and codes are available at github.com. We briefly introduce our model in https://www.youtube.com/watch?v=YIKd5bcHcGA.

INDEX TERMS Generalization, Autonomous Driving, Reinforcement Learning, Off-road Environments, Imitation Learning

I. INTRODUCTION

Recently, deep neural networks (DNNs) [1] have greatly improved autonomous driving [2]. DNNs approximate complex functions by processing high-dimensional sensory data such as images. Autonomous driving research has developed in two directions. Modular pipelines sequentially process data [3], [4], and end-to-end methods directly produce motion controls [5]. The modular pipeline features an NN-based perception module, a rule-based planner, and a maneuver controller. The end-to-end methods are trained via imitation learning (IL) or reinforcement learning (RL).

However, autonomous driving systems must be generalized to adverse conditions and wide terrain/weather variations. It is important to distinguish valuable features in sensory inputs when dealing with various conditions because generalization directly affects safety. Additionally, adversarial attacks may slightly change the inputs, confusing the predictions [6], [7]. Such attacks exploit samples very similar to the original samples but change the (correct) predictive labels of the trained model. Several approaches are used to test the robustness of autonomous vehicles to such attacks. Adversarial test cases can be generated via simulation [8] or by modifying the samples [9]. In addition, many researchers have studied training autonomous driving on simulations and transferring to the real world [10], [11]. This task requires the ability to generalize because it is impossible to build simulators that are exactly the same as the real world.

Although simulators are widely used for training and evaluating autonomous driving, open-source simulators are limited to benchmark generalization. Several open-source simulators are available for car racing, such as TORCS [12] and high-fidelity autonomous driving [13], [14], [15]. These
I. Han et al.: A New Open-Source Off-road Environment for Benchmark Generalization of Autonomous Driving

FIGURE 1. A flowchart of the off-road generalization evaluations. The environment consists of Unreal engine assets and Python scripts.

Simulators provide many environments on racing tracks and urban cities, but they are similar to each other. Thus, the environments lack adverse conditions and terrain/weather varieties. Training and evaluating driving systems under limited conditions may cause overfitting, so the systems poorly generalize to unseen environments.

Benchmarks are available that measure the generalization afforded by RL. Recently, the poor generalization afforded by RL was highlighted in several studies [16]. RL does not even permit transfers among very similar environments. The authors of [17] quantified RL generalization through procedurally generating environments. The authors of [18] developed the ProcGen environment to benchmark RL generalization. However, ProcGen is unsuitable for autonomous driving benchmark generalization. The environment is very different from driving tasks; it consists of simple two-dimensional video game tasks. It is natural for evaluating the performance of driving systems with driving tasks.

We present an open-source integrated framework for autonomous driving benchmark generalization. Our framework supports the end-to-end process of driving experiments: training, evaluating, analyzing and visualizing. Fig. 1 represents the framework compositions. The contribution of our study is clearly described as follows.

First, to effectively benchmark generalization, we utilize off-road environments that provide diverse and adverse conditions. The off-roads feature forests, plateaus, deserts, snowfields, and combinations. Each off-road offers visual diversity and various structures. The graphical and structural diversity makes it difficult for agents to generalize in an unseen environment, making it an effective measure of generalization.

Second, for accurate evaluation and analysis, we define tasks according to the traveled distance and propose a method for evaluating autonomous driving performance based on the success rate of these tasks. In addition, the framework offers visualization tools and various information for detailed analysis of experiments. Activation map visualization and bird-eye views are available, and the user can analyze an experiment with various measurements and driving errors.

Third, we prove that current end-to-end driving methods poorly generalize to unseen environments and analyze the reason for the failure through driving errors and activation map visualization. For training driving systems, the framework provides various baselines to build end-to-end driving, such as IL, RL and LBC (learning by cheating). We evaluate the generalizations of the three end-to-end driving methods in our environments. These methods failed to generalize to unseen off-road environments.

II. BACKGROUND

A. END-TO-END DRIVING

Importantly, an end-to-end driving system does not require complex, sequential data processing when steering, thus affording fast and intuitive decision-making [19]. Conditional IL (CIL) has been employed [20]. This addresses the fundamental problem of IL; it is impossible to control the agent, rendering such learning impractical. An CIL method is based on high-level commands that cause agents to turn at certain intersections. The authors of [21] developed an autonomous, vision-based obstacle avoidance system for off-road environments. Human driver data was employed for IL; agents thus trained were exposed to various conditions to enhance generality. However, only a few training sets were used. Agents engaged in online IL perform well in real-world experiments. Off-road environments are also considered in this paper, but we focus on measuring generalization ability.
The authors of [22] proposed an end-to-end method for race driving. They trained and tested their method on various road structures, graphics and physics. They proved that their method generalizes well on unseen tracks. However, they used the World Rally Championship 6 (WRC6) environment for their experiments, which is not for research use and not open source; thus, it is unsuitable to use it for benchmark generalization. Using multimodal inputs can improve end-to-end driving [23]. They trained their end-to-end driving agent with RGB images, depth images and measurements. Therefore, multimodality is an important factor in driving simulators. In addition, uncertainty-aware driving through uncertainty prediction has been studied in terms of end-to-end driving [24]. It can measure the reliability of driving agent decisions with this method, and it is important for autonomous driving safety. Our environment provides various graphical features; thus, it also can be used as a test bed for uncertainty-aware driving algorithms.

B. GENERALIZATION IN RL
RL (a form of machine learning) features rewards obtained by interacting with the environment. RL is frequently used for end-to-end driving. RL training commonly employs simulated environments. However, RL affords only poor generalization. Generalization problems can be divided into three categories: visual changes and differences in dynamics and content. Visual changes include changes in texture, background color, and object colors. The authors of [16] showed that slight changes in input images triggered catastrophic collapse in RL agents. To measure the generalization of such agents, the authors of [18] proposed a benchmark that uses procedural generation to create content. Generalization ability depends on network architecture, stochasticity, and regularization. To improve visual generalization, many data-augmentation-based methods have been proposed. The authors of [25] developed a random data-augmentation network to improve visual generalization [25]; the empirical results were good. Additionally, a convolutional network was used to produce both original and augmented images via feature-matching loss. This addressed the high variance problem in domain randomization techniques. The authors of [26] applied various image augmentation techniques to DRL. An augmented data method (RAD) was optimal in terms of sample efficiency and generalization. Simple DRL regularization using augmented images was used to build general RL agents [27]. The value function was regularized by averaging the value functions of various augmented images. Importantly, such data-augmentation-based methods are not associated with auxiliary losses and do not require world model approximation.

III. METHODS
A. OFF-ROAD ENVIRONMENT
We utilized CARLA, an open-source driving simulator for autonomous driving systems. CARLA provides various sensory inputs and a suite of functions for training autonomous vehicle systems. The overview of our off-roads is shown in Fig. 2. The image and rendered images are shown in Fig. 5. Additionally, various weather situations are available (Fig. 4). We created eight off-road environments using an Unreal engine. Quixel Megascan assets (three-dimensional trees and rocks) are employed to draw off-road maps (which are not city-like grids). The off-road maps are ported to a CARLA-compatible environment. No lanes or guardrails mark the road boundaries. The maps feature diverse landscapes and geographic features (Fig. 2). The details follow.

- **Structures.** The maps contain different lanes. For mountains 1–4, there are no forks; only lane-keeping can be measured. The desert, snowfield, plateau, and combined maps feature forks; it is possible to train and test conditional policy. Unlike cities, off-roads are not standardized. They vary, as do the surroundings. Most city roads are gridded, but off-roads are not. Therefore, the agent must adjust steering more precisely than when in a city.

- **Dynamic elevation.** City roads may rise or fall slowly; off-road slopes may change very rapidly. The off-road elevation is visualized in Fig. 3. In cities, throttle adjustments are necessary only when there are obstacles in the road or when turning. Off-road, speed is constantly adjusted. Additionally, off-road slopes affect the viewing angle, thus, the areas of sky and ground, and peripheral vision. The agent must be robust to such changes.

- **Visual diversity.** A real-world off-road environment is accompanied by many visual changes. The surrounding elements in deserts, snowfields, plateaus, and mountains are very different, and the colors are diverse. Additionally, the angles of the principal light sources and illumination intensities vary. Mountains 1–4 feature forests, grasses of different textures, and roads. In the desert, most colors are red and yellow. The snowfield is entirely white; the trees and cliffs are snow-covered. The highlands (a plateau) feature large and small roadside rocks of texture and color similar to those of bare ground. The combined field features all forests, deserts, snowfields, and plateaus. The agent must recognize roads or obstacles that differ visually. Weather changes add further diversity.

- **Real-world similarity.** The off-road benchmark is very similar to real off-road environments. Most autonomous driving research is focused on cities; however, off-road autonomous driving is required when rallying. Real off-road driving is dynamic and dangerous and is associated with high risks of injury or damage. Therefore, any simulator learning must be transferred to the real world rallying.

The measurements are listed in Table 1; these include CARLA server data and the waypoints. Agents were trained using these data. These are included in the birds-eye views, which show the location of the agent and the waypoints, and other current data. A sample birds-eye image is shown.
FIGURE 2. Birds-eye views of eight off-road environments. The upper images are mountains 1–4; the bottom images are desert, snowfield, highlands, and combined.

FIGURE 3. Side-penetrating views of eight off-road environments with respect to roads. The Y-axis is height, and the X-axis is width. The upper images are mountains 1–4; the bottom images are desert, snowfield, highlands, and combined. Many hills and slopes can be seen in all off-roads.

TABLE 1. Measurements provided by the environment. They provide various details of the current state of the environment. Additionally, they can support experiments in aspects of supervision and evaluation of autonomous driving systems.

| Measurements               | Description                                           |
|----------------------------|-------------------------------------------------------|
| speed                      | current speed of agent                                |
| location                   | current (x, y) coordinates of agent                  |
| goal waypoint              | front waypoint of agent                               |
| distance from goal waypoint| Euclidean distance between goal waypoint and current location |
| progress                   | (number of completed waypoints) / (number of total waypoints) |
| next waypoints             | coordinates of next k waypoints                      |
| previous waypoints         | coordinates of previous k waypoints                  |
| waypoints heading          | angle of next waypoint and previous waypoint          |
| current heading            | current agent heading                                |
| distance from center       | distance between lane center and agent location       |
| lane width                 | lane width at current location                        |
| collision                  | collision damage                                      |
| nospeed time step          | number of time steps in which the speed is lower than a threshold |

in Fig. 6. Additionally, Table 2 represents the road lengths in the environment, and Table 3 shows the environmental parameters used in this paper.
I. Han et al.: A New Open-Source Off-road Environment for Benchmark Generalization of Autonomous Driving

FIGURE 4. Various weather on Mountain 2. Clear noon, clear sunset, and midrainy noon (from left). These weather changes are a feature of CARLA and can be used in our environment as well. On clear noon, there are no clouds, and the sun is in the middle of the sky, while the sun is on the side of the sky and also on the clear sunset. It is cloudy and rainy on midrainy noon.

FIGURE 5. Various sensory inputs of CARLA that are also available in our environment. Left image: RGB; right image: depth. The depth image is ground-truth information. It can support autonomous driving experiments in terms of perception.

FIGURE 6. Birds-eye view of off-road driving in aspects of waypoints. The blue points are whole waypoints (left/middle/right), the orange points are previous waypoints, and the green points are the next waypoints that the agent should pass by. Additionally, the brown line represents the current heading of the car, and the pink line represents waypoint heading, which is the angle between the next waypoint and a previous waypoint. Further details are in Table 1.

B. DRIVING BENCHMARK

We propose a performance evaluation benchmark to accurately measure the driving agent performance. Generalization ability measured using various weather conditions and environments. Of the eight off-road environments and six weather conditions, the agents were trained using only “mountain 1” and “clear noon” weather. For simplicity, some available CARLA weather conditions were removed. The new (unseen) weather conditions were “clear sunset”, “midrainy sunset”, “midrainy noon”, “wet cloudy noon” and “wet cloudy sunset”. For each condition, 10 trials were performed; thus, 3,360 trials used either algorithm. In addition, we defined lane-keeping tasks, which require driving a certain distance while keeping the lane. The tasks are designed to increase the difficulty, and the target mileage ranges from low to high. The 7 tasks are defined as follows: 100 m, 200 m, 300 m, 400 m, 500 m, 600 m and whole lap. The whole lap task requires driving the entire map to be successful. The algorithms are evaluated through the success rate for these tasks.

“Success” is defined as travel for a certain distance from a starting point while avoiding

- Collision
- Out-of-lane episodes
- Speeds lower than 1 km/h for 20 steps in an episode

TABLE 2. Road length of each environment. Mountains 1-4 are of varying lengths, but other off-roads are similar in length. Various off-road lengths contribute to driving ability evaluation precision.

| Maps            | Road lengths |
|-----------------|--------------|
| Mountain 1      | 1,200 m      |
| Mountain 2      | 670 m        |
| Mountain 3      | 3,800 m      |
| Mountain 4      | 7,000 m      |
| Desert          | 1,800 m      |
| Snowy field     | 1,830 m      |
| Highlands       | 2,020 m      |
| Combined fields | 2,390 m      |
TABLE 3. Environmental parameters used in this paper. There are physical parameters (road adhesion, etc.) and vehicle parameters (Max RPM, MOI, etc.). Additionally, there are visual parameters such as field of vision.

| Environmental parameters                  | Value            |
|-------------------------------------------|------------------|
| Road adhesion                             | 0.7              |
| Field of vision                           | 90°              |
| Max RPM                                   | 3,600            |
| MOI                                       | 1.0K g/m²        |
| Damping rate full throttle                | 0.15K g/m²/s     |
| Damping rate zero throttle                | 2.0K g/m²/s      |
| Damping rate zero throttle                | 0.35K g/m²/s     |
| Differential type                         | Limited slip 4 W |
| Front rear split                          | 0.45             |
| Front left right split                    | 0.5              |
| Rear left right split                     | 0.5              |
| Center bias                               | 1.3              |
| Front bias                                | 1.3              |
| Rear bias                                 | 1.3              |
| Automatic transmission                    | True             |
| Gear switch time                          | 0.5 sec          |
| Gear autobox latency                      | 2.0 sec          |
| Vehicle mass                              | 2,353 Kg         |
| Vehicle drag coefficient                  | 0.3              |
| Chassis width                             | 194 cm           |
| Chassis height                            | 185.5 cm         |
| Linear damping                            | 0.01             |
| Angular damping                           | 0.0              |
| Wheel shape radius                        | 39               |
| Wheel shape width                         | 10               |
| Wheel mass                                | 20               |
| Wheel damping rate                        | 0.25             |
| Wheel steer angle                         | 45.0             |
| Tire lat stiff max                        | 2.0              |
| Tire lat stiff value                      | 17.0             |
| Tire long stiff value                     | 1,000.0          |
| Tire suspension force offset              | 0.0              |
| Tire suspension max raise                 | 10.0             |
| Tire suspension max drop                  | 10.0             |
| Tire suspension natural frequency         | 7.0              |
| Max brake torque                          | 1,500.0          |
| Max hand brake torque                     | 3,000.0          |

Naturally, exiting a lane and moving too slowly fail. However, collisions can be one of the causes of failure in off-road environments. Because there are many obstacles near the lane, the agent can collide with the obstacle before removing the lane. The start point is randomly selected for every episode. The evaluation algorithm is clearly represented in Algorithm 1.

IV. AUTONOMOUS DRIVING EXPERIMENTS

IL, RL and LBC were tested on the off-road benchmark. They were used to build end-to-end models in a recent autonomous driving study. Previously, the models were tested in terms of navigation or obstacle avoidance. As we focus on generalization, the agents were evaluated with a simple lane-keeping task. The agent received RGB (red green blue) images from a front-facing camera and speed information; the agent could steer, throttle, and brake. To learn rapidly, the throttle range was set to -1 to 1; if the value was negative, braking was excessive, and the throttle was then set to zero. The IL and RL training details follow. The network structures, which was the IMPALA [28] convolutional architecture, were identical; this affords both good generalization and computational efficiency [18]. The Adam optimizer was employed to optimize weights. The input images were normalized to certain means (0.485, 0.456, 0.406 for each RGB channel) and standard deviations (0.229, 0.224, 0.225 for each RGB channel).

A. IMITATION LEARNING

IL builds an agent that imitates the inputs and actions of an expert [29], [30]; this is rapid if the data is sufficient. Many studies have applied imitation learning to driving tasks, and researchers have developed conditional imitation learning, which can control direction high-level commands [31], [20]. A hard-coded expert is used to collect data. Two types of ILs are trained: vanilla ILs and randomized ILs (RILs). The RIL was trained with various image augmentation methods for generalization. A total of 515,382 data tuples (RGB images, speeds, and actions) were collected and used as training data. Given the lane structures (generally straight), most steer...
values were near zero. To resolve this imbalance, the steer label distributions were uniformly adjusted by resampling. The IL model and RIL model were trained with a learning rate of $10^{-4}$ and batch size of 60. The IL model and RIL model were trained for only one epoch. They were trained with the same loss function, the MAE (mean absolute error). The loss function is as follows:

$$\mathcal{L}_{IL,RIL} = \frac{1}{n} \sum_{i=1}^{n} |y_i - f(x_i; \theta)|$$

$\theta$ is the weight of the neural network, $y_i$ is a label steer and $x_i$ is an input image. $n$ is the number of samples.

### B. REINFORCEMENT LEARNING

RL approximates an optimal policy using a DNN that is rewarded [32]. Recently, RL has achieved dramatic improvements in various real-world domains, such as Go [33], StarCraft [34] and DOTA [35]. DRLs are often used to build autonomous driving systems [36], [37], [22]. The Asynchronous Advantage Actor-Critic (A3C) algorithm [38], which is a policy-based algorithm frequently employed to handle continuous control problems, was used to build an RL agent. The RL agent trained for 250,000 steps. Various environmental measurements were utilized to build the reward function. The reward function consists of a weighted sum of five terms: distance toward next waypoint $d$ in m, distance toward the center of lane $c$ in m, angle of the car $\alpha$ in radians, angle between next waypoint and previous waypoint $w$ in radians, and speed $v$ in m/s. The reward was clipped from -1 to 1.

$$r_t = (d_{t-1} - d_t) - (w_t - a_t) - 0.1c_t + 0.05v_t$$

The hyperparameters of the RL model are shown in Table 4.

### TABLE 4. Hyperparameters of the RL model

| Hyperparameters       | values |
|-----------------------|--------|
| entropy coefficient   | 0.01   |
| value loss coefficient| 0.5    |
| gamma                 | 0.99   |
| learning rate         | 0.00001|
| gradient clip threshold| 0.5  |
| n step                | 20     |

Because RLs agent shows poor performance with color image inputs and do not converge, the RL agent was trained with grayscale images. The RL loss function was as follows:

$$\mathcal{L}_{policy} = -\log \pi(a_t|s_t)(R_t - V(s_t))$$

$$\mathcal{L}_{value} = (R_t - V(s_t))$$

$$\mathcal{L}_{entropy} = -\sum_a \pi(a|s_t) \log \pi(a|s_t)$$

$$\mathcal{L}_{RL} = \mathcal{L}_{policy} + c \cdot \mathcal{L}_{value} + d \cdot \mathcal{L}_{entropy}$$

$c$ and $d$ are hyperparameters, which are the value loss coefficient and entropy coefficient, respectively.

### C. LEARNING BY CHEATING

Learning by cheating (LBC) [39] is a state-of-the-art method for building an end-to-end driving agent through two steps. First, a privileged agent trained through cheating learns with privileged information provided by the environment. Second, sensorimotor agents imitate privileged agents. Since we have hard-coded experts, it is used instead of training the privileged agent. Similar to IL and RIL, two types of LBC were trained: vanilla LBC and randomized LBC (RLBC). The RLBC was trained with various image augmentation methods for generalization. The LBC and RLBC trained for 50 epochs, with a learning rate of 0.0001.

### V. RESULTS AND DISCUSSION

The results were analyzed in four categories: learning performance, generalization, Grad-CAM visualization and driving errors. First, we focus on the learning performance of the four algorithms and the evaluation results on the training conditions. Second, the results were analyzed in terms of generalization. Three perspectives were analyzed: new weather, new off-roads and new off-roads and weather. Finally, we analyze the reason for generalization failure in more detail through Grad-CAM visualization and driving errors.

#### Learning performance

Fig. 7 represents the learning performance of the algorithms. The four IL-based algorithms (IL, RIL, LBC, RLBC) performed almost perfectly in training conditions. However, RL performed poorly. The cause of the poor RL performance is expected to be a sharp change in the steering value. The IL-based method maintained a stable steering value, but the RL changed the steering value very sharply. RL agents changed the steering value significantly to avoid negative rewards and approach the center of the lane, but this caused greater acceleration, causing the car to move to the other side of the lane, again resulting in negative rewards. Interestingly, LBC and RLBC learn very rapidly. They achieved a very high reward (0.6) in the early training phase. Additionally, RIL learns slower than IL because of data augmentation. One interesting thing is that RLBC generally failed faster than LBC. Both LBC and RLBC were trained for the same number of episodes, but RLBC experienced nearly 70,000 steps, while LBC experienced nearly 90,000 steps. An episode consists of sequence steps, and the episode ends when the driving errors are met. If an agent trained for the same number of episodes and experienced fewer steps, the agent failed faster and made more mistakes. This can also be explained by the effect of data augmentation.

#### Generalization

Table 5 shows the percentages of successfull episodes under four conditions on seven tasks. Surprisingly, the results on training conditions and new weather were extremely different from the results on new off-roads and new off-roads-with-new weather. Under conditions similar to training, nonrandomized models (IL, LBC) performed better; under conditions very different from training, randomized models (RIL, RLBC) performed better. All methods show close performance to training conditions in terms of new weather conditions. Generalization was excellent. However, generalization failed for new off-roads and new off-roads-with-new weather conditions. Specifically, RIL outperformed IL on new off-roads and new off-roads-with-new weather conditions.

---

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
conditions. The same phenomenon was observed in LBC and RLBC. Thus, input image randomization improved generalization. The success percentages by the off-road environment are shown in Table 6. All agents performed well on the mountains but usually failed in the desert, snowfield, highlands, and combined field. This is because mountains 2–4 are more similar to the training environment than are the other four environments. Although mountains 3–4 are visually very similar to mountain 1, most agents failed to generalize to mountains 3–4 in the whole lap task. This is because, unlike mountain 1, mountains 3 and 4 have large hills, which makes it difficult for the agent to control speed.

Grad-CAM visualization. We prepared CNN activation maps with the aid of Grad-CAM [40], a gradient-based attention visualization method. The Grad-CAM results are shown in Fig. 8. During training, all algorithms usually focused on the road. RL focused more strongly than the others. In the new environment, all algorithms seemed to focus on random points and failed to distinguish important visual features such as the lane route. Surprisingly, all agents focused not only on roads but also on the sky. This may be explained by the structure of mountain 1, which is the training condition. On mountain 1, most roads surrounded by trees could infer the lane routes from the shape of the sky. However, in terms of generalization, making decisions based on the shape of the sky is inefficient. The shape of the sky cannot change if there are no trees surrounding the road or other obstacles. Actually, in deserts, snowy fields, highlands and combined fields, it cannot use the shape of the sky to estimate the lane route. Therefore, generalization failure in these environments can be explained by inference based on the shape of the sky.

Driving errors. Driving errors are shown in Table 7 and Table 8. Under training conditions, IL, LBC and RLBC perfectly succeeded in all episodes without any failure. However, in new off-roads and new off-roads-with-new weather, failure was usually an out-of-lane episode (steering control failure). The failures in mountains 3–4 were mostly low-speed episodes. Speed has nothing to do with steering control. In practice, simply increasing the throttle value can solve the speed problem. However, too high of a speed can cause unstable driving. Additionally, in the desert, snowy field, and highlands combined field, most episodes were out of lane for all algorithms. This means that generalization failure caused incorrect steering control.

VI. CONCLUSIONS AND FUTURE WORKS

Generalization is essential to ensure safe autonomous driving. We developed an off-road benchmark to assess generalization. We used different off-roads, which provide visually diverse graphics and various structures. The results of autonomous driving experiments indicate that our off-road environment precisely measures the generalization ability of autonomous driving, and the current end-to-end driving methods lack generalization ability. The agents successfully generalize to unseen weather but completely fail on some off-roads. Further improvements in autonomous driving can be evaluated on the environment in terms of generalization. In addition, the environment can be used to build autonomous driving systems for off-road driving. To the best of our knowledge, this paper is the first to present a simulated off-road environment for autonomous driving. The environment provides various realistic off-road environments, such as mountains, deserts, snowy fields, and highlands. Previous simulated environments were usually urban cities or racing tracks. In the future, as RL is promising in terms of end-to-end driving, various generalization methods featuring DRL will be employed. Generalization of different dynamics is also a future work. A benchmark is required because dynamics can change in the real world. For example, a car may skid in the rain and move in an unintended direction. Autonomous driving systems must deal with such situations.

REFERENCES

[1] I. Goodfellow, Y. Bengio, and A. Courville, Deep learning. MIT press, 2016.
[2] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, “A survey of deep learning techniques for autonomous driving,” Journal of Field Robotics, vol. 37, no. 3, pp. 362–386, 2020.
[3] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, “A survey of motion planning and control techniques for self-driving urban vehicles,” IEEE Transactions on intelligent vehicles, vol. 1, no. 1, pp. 33–55, 2016.
[4] U. Franke, “Autonomous driving,” Computer Vision in Vehicle Technology: Land, Sea & Air, pp. 24–54, 2017.
[5] S. Hecker, D. Dai, and L. Van Gool, “End-to-end learning of driving models with surround-view cameras and route planners,” in Proceedings of the european conference on computer vision (eccv), 2018, pp. 435–453.
[6] T. Dreesis, S. Jha, and S. A. Seshia, “Semantic adversarial deep learning,” in International Conference on Computer Aided Verification. Springer, 2018, pp. 3–26.
TABLE 5. Success percentages by the conditions.

| Training Conditions | New weather |
|---------------------|-------------|
| Tasks               | IL | RIL | RL | LBC | RLBC |
| 100 m               | 100 | 80 | 30 | 100 | 100 |
|                     | 98 | 96 | 20 | 100 | 100 |
| 200 m               | 100 | 100 | 0 | 100 | 100 |
|                     | 98 | 88 | 6 | 100 | 100 |
| 300 m               | 100 | 90 | 0 | 100 | 100 |
|                     | 98 | 88 | 0 | 100 | 100 |
| 400 m               | 100 | 100 | 0 | 100 | 100 |
|                     | 100 | 76 | 0 | 100 | 100 |
| 500 m               | 100 | 100 | 0 | 100 | 100 |
|                     | 100 | 80 | 0 | 98 | 100 |
| 600 m               | 100 | 100 | 0 | 100 | 100 |
|                     | 100 | 80 | 0 | 100 | 100 |
| Whole lap           | 100 | 100 | 0 | 100 | 100 |
|                     | 100 | 57 | 0 | 100 | 100 |

| New off-road        | IL | RIL | RL | LBC | RLBC |
|---------------------|----|-----|----|-----|------|
| 100 m               | 28 | 38 | 2 | 32 | 37 |
|                     | 36 | 38 | 11 | 32 | 45 |
| 200 m               | 31 | 34 | 0 | 24 | 38 |
|                     | 28 | 34 | 1 | 23 | 39 |
| 300 m               | 18 | 37 | 0 | 17 | 34 |
|                     | 22 | 30 | 0 | 24 | 32 |
| 400 m               | 20 | 34 | 0 | 22 | 27 |
|                     | 18 | 29 | 1 | 20 | 34 |
| 500 m               | 11 | 17 | 0 | 14 | 21 |
|                     | 17 | 25 | 0 | 19 | 30 |
| 600 m               | 14 | 18 | 0 | 17 | 20 |
|                     | 16 | 23 | 0 | 19 | 30 |
| Whole lap           | 0 | 2 | 0 | 0 | 0 |
|                     | 0 | 11 | 0 | 0 | 2 |

| Avg.                | 100 | 95 | 4 | 100 | 100 |
|                     | 99 | 80 | 3 | 99 | 100 |

| New off-road & weather | IL | RIL | RL | LBC | RLBC |
|------------------------|----|-----|----|-----|------|
| 100 m                  | 28 | 38 | 2 | 32 | 37 |
|                       | 36 | 38 | 11 | 32 | 45 |
| 200 m                  | 31 | 34 | 0 | 24 | 38 |
|                       | 28 | 34 | 1 | 23 | 39 |
| 300 m                  | 18 | 37 | 0 | 17 | 34 |
|                       | 22 | 30 | 0 | 24 | 32 |
| 400 m                  | 20 | 34 | 0 | 22 | 27 |
|                       | 18 | 29 | 1 | 20 | 34 |
| 500 m                  | 11 | 17 | 0 | 14 | 21 |
|                       | 17 | 25 | 0 | 19 | 30 |
| 600 m                  | 14 | 18 | 0 | 17 | 20 |
|                       | 16 | 23 | 0 | 19 | 30 |
| Whole lap              | 0 | 2 | 0 | 0 | 0 |
|                       | 0 | 11 | 0 | 0 | 2 |

| Avg.                  | 17 | 26 | 0 | 18 | 25 |
|                       | 19 | 27 | 2 | 19 | 30 |

FIGURE 8. Grad-CAM results of all algorithms
I. Han et al.: A New Open-Source Off-road Environment for Benchmark Generalization of Autonomous Driving

| TABLE 6. Success percentages by the off-road. |
|--------------------------------------------|
| **Tasks** | **Mountain 1** | **Mountain 2** | **Mountain 3** | **Mountain 4** | **Desert** | **Snowy field** | **Highlands** | **Combined field** |
|          | IL  | RIL | RL  | LBC | RLBC | IL  | RIL | RL  | LBC | RLBC | IL  | RIL | RL  | LBC | RLBC | IL  | RIL | RL  | LBC | RLBC |
| 100 m    | 98  | 93  | 21  | 100 | 100  | 85  | 83  | 36  | 45  | 83   | 85  | 83  | 36  | 45  | 83   | 85  | 83  | 36  | 45  | 83   |
| 200 m    | 100 | 90  | 5   | 100 | 100  | 61  | 80  | 5   | 3   | 66   | 61  | 80  | 5   | 3   | 66   | 61  | 80  | 5   | 3   | 66   |
| 300 m    | 98  | 88  | 0   | 100 | 100  | 16  | 80  | 3   | 3   | 46   | 16  | 80  | 3   | 3   | 46   | 16  | 80  | 3   | 3   | 46   |
| 400 m    | 100 | 80  | 0   | 100 | 100  | 6   | 78  | 6   | 0   | 53   | 6   | 78  | 6   | 0   | 53   | 6   | 78  | 6   | 0   | 53   |
| 500 m    | 100 | 83  | 0   | 98  | 100  | 5   | 71  | 0   | 0   | 36   | 5   | 71  | 0   | 0   | 36   | 5   | 71  | 0   | 0   | 36   |
| 600 m    | 100 | 83  | 0   | 100 | 100  | 1   | 80  | 1   | 0   | 28   | 1   | 80  | 1   | 0   | 28   | 1   | 80  | 1   | 0   | 28   |
| Whole lap| 100 | 65  | 0   | 100 | 100  | 1   | 68  | 1   | 0   | 15   | 1   | 68  | 1   | 0   | 15   | 1   | 68  | 1   | 0   | 15   |
| Avg.     | 99  | 83  | 3   | 99  | 100  | 25  | 77  | 7   | 7   | 47   | 25  | 77  | 7   | 7   | 47   | 25  | 77  | 7   | 7   | 47   |

[21] U. Muller, J. Ben, E. Cosatto, B. Flepp, and Y. L. Cun, “Off-road obstacle avoidance through end-to-end learning,” in *Advances in Neural Information Processing Systems*. Citeseer, 2006, pp. 739–746.

[22] M. Jaritz, R. De Charette, M. Toromanoff, E. Perot, and F. Nashashibi, “End-to-end race driving with deep reinforcement learning,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2070–2075.

[23] Y. Xiao, F. Codevilla, A. Gurram, O. Urfalioglu, and A. M. López, “Multimodal end-to-end autonomous driving,” *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[24] L. Tai, P. Yan, Y. Chen, C. Liu, H. Ye, and M. Liu, “Visual-based autonomous driving deployment from a stochastic and uncertainty-aware perspective,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 2622–2628.

[25] K. Lee, K. Lee, J. Shin, and H. Lee, “Network randomization: A simple technique for generalization in deep reinforcement learning,” in *International Conference on Learning Representations*, 2019.

[26] M. Laskin, K. Lee, A. Stooke, L. Prato, P. Abbeel, and A. Srinivas, “Reinforcement learning with augmented data,” *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[27] D. Yarats, I. Kostrikov, and R. Fergus, “Image augmentation is all you need: Regularizing deep reinforcement learning from pixels,” in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=GY6-6sTvGaf

[28] L. Espeholt, H. Soyer, R. Munos, K. Simonyan, V. Mnih, T. Ward, Y. Doron, V. Firoiu, T. Harley, I. Dunning *et al.*, “Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures,” in *International Conference on Machine Learning*. PMLR, 2018, pp. 1407–1416.

[29] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne, “Imitation learning: A survey of learning methods,” *ACM Computing Surveys (CSUR)*, vol. 50, no. 2, pp. 1–35, 2017.
TABLE 7. Error percentages by the conditions.

| Training conditions | New weather |
|----------------------|-------------|
| Tasks               | IL | RIL | RL | LBC | RLBC |
| Out of lane         | 0  | 33  | 25 | 0   | 0    |
| Collision           | 0  | 33  | 58 | 0   | 0    |
| No speed            | 0  | 33  | 16 | 0   | 0    |
| Tasks               | IL | RIL | RL | LBC | RLBC |
| Out of lane         | 0  | 92  | 9  | 100 | 0    |
| Collision           | 0  | 8   | 64 | 0   | 0    |
| No speed            | 0  | 100 | 25 | 0   | 0    |

TABLE 8. Error percentages by the various off-roads.

| Mountain1          | Mountain2          |
|--------------------|--------------------|
| Tasks              | IL | RIL | RL | LBC | RLBC |
| Out of lane        | 0  | 90  | 12 | 100 | 0    |
| Collision          | 0  | 10  | 63 | 0   | 0    |
| No speed           | 100| 1   | 24 | 0   | 0    |
| Tasks              | IL | RIL | RL | LBC | RLBC |
| Out of lane        | 93 | 77  | 44 | 85  | 82   |
| Collision          | 6  | 22  | 28 | 12  | 17   |
| No speed           | 0  | 27  | 1  | 0   | 0    |

| Mountain3          | Mountain4          |
|--------------------|--------------------|
| Tasks              | IL | RIL | RL | LBC | RLBC |
| Out of lane        | 0  | 1   | 11 | 36  | 34   |
| Collision          | 34 | 41  | 74 | 51  | 71   |
| No speed           | 65 | 56  | 14 | 13  | 2    |
| Tasks              | IL | RIL | RL | LBC | RLBC |
| Out of lane        | 0  | 24  | 26 | 10  | 40   |
| Collision          | 2  | 10  | 45 | 14  | 68   |
| No speed           | 96 | 65  | 28 | 75  | 2    |

| Desert             | Snowy field        |
|--------------------|--------------------|
| Tasks              | IL | RIL | RL | LBC | RLBC |
| Out of lane        | 91 | 82  | 56 | 86  | 81   |
| Collision          | 9  | 17  | 31 | 13  | 19   |
| No speed           | 0  | 0   | 12 | 0   | 0    |
| Tasks              | IL | RIL | RL | LBC | RLBC |
| Out of lane        | 88 | 87  | 79 | 85  | 89   |
| Collision          | 11 | 12  | 14 | 11  | 11   |
| No speed           | 0  | 0   | 6  | 2   | 0    |

| Highlands          | Combined field     |
|--------------------|--------------------|
| Tasks              | IL | RIL | RL | LBC | RLBC |
| Out of lane        | 90 | 90  | 53 | 88  | 90   |
| Collision          | 9  | 10  | 17 | 9   | 10   |
| No speed           | 0  | 0   | 29 | 2   | 0    |
| Tasks              | IL | RIL | RL | LBC | RLBC |
| Out of lane        | 95 | 92  | 59 | 91  | 90   |
| Collision          | 4  | 7   | 18 | 7   | 9    |
| No speed           | 0  | 0   | 21 | 0   | 0    |

[30] Y. Pan, C.-A. Cheng, K. Saigol, K. Lee, X. Yan, E. A. Theodorou, and B. Boots, “Imitation learning for agile autonomous driving,” The International Journal of Robotics Research, vol. 39, no. 2-3, pp. 286–302, 2020.
[31] J. Hawke, R. Shen, C. Gurau, S. Sharma, D. Reda, N. Nikolov, P. Mazur, S. Micklethwaite, N. Griffiths, A. Shah et al., “Urban driving with conditional imitation learning,” in 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2020, pp. 251–257.
[32] R. S. Sutton and A. G. Barto, Reinforcement learning: An introduction, MIT press, 2018.
[33] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton et al., “Mastering the game of go without human knowledge,” nature, vol. 550, no. 7676, pp. 354–359, 2017.
[34] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev et al., “Grandmaster level in starcraft ii using multi-agent reinforcement learning,” arXiv preprint arXiv:1912.06680, 2019.
[35] C. Berner, G. Brockman, B. Chan, V. Cheung, P. Döhlak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse et al., “Dota 2 with large scale deep reinforcement learning,” arXiv preprint arXiv:1912.06680, 2019.
[36] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. Al Sallab, S. Yogamani, and P. Pérez, “Deep reinforcement learning framework for autonomous driving,” Electronic Imaging, vol. 2017, no. 19, pp. 70–76, 2017.
[37] 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 International conference on machine learning. PMLR, 2016, pp. 1928–1937.
[38] D. Chen, B. Zhou, V. Koltun, and P. Krähenbühl, “Learning by cheating,” in Conference on Robot Learning. PMLR, 2020, pp. 66–75.
[39] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-cam: Visual explanations from deep networks via gradient-based localization,” in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.