KING: Generating Safety-Critical Driving Scenarios for Robust Imitation via Kinematics Gradients

Niklas Hanselmann [1 2 3] **Katrin Renz** [2 3] **Kashyap Chitta** [2 3] **Apratim Bhattacharyya** [2 3] **Andreas Geiger** [2 3]

Abstract

Simulators offer the possibility of scalable development of self-driving systems. However, current driving simulators exhibit naïve behavior models for background traffic. Hand-tuned scenarios are typically used to induce safety-critical situations. An alternative approach is to adversarially perturb the background traffic trajectories. In this paper, we study this approach to safety-critical driving scenario generation using the CARLA simulator. We use a kinematic bicycle model as a proxy to the simulator's true dynamics and observe that gradients through this proxy model are sufficient for optimizing the background traffic trajectories. Based on this finding, we propose KING, which generates safety-critical driving scenarios with a 20% higher success rate than black-box optimization, which previous work relies on. Furthermore, we demonstrate that the generated scenarios can be used to fine-tune imitation learning agents, leading to improved collision avoidance.

1. Introduction

After years of steady progress, autonomous driving systems are getting closer to maturity (Janai et al., 2020). Due to the high consequences of failure, they have to satisfy extraordinarily high standards of robustness in the face of unseen and safety-critical scenarios. However, real-world data collection and validation for these situations lacks the necessary scalability (O' Kelly et al., 2018; Norden et al., 2019). To cover this long-tail of driving scenarios, simulation is a promising solution. Unfortunately, current simulators such as CARLA (Dosovitskiy et al., 2017) build on simple behavior models for background agents and do not provide the necessary diversity in traffic. This poses a major challenge in the adoption of driving agents trained in simulation using imitation learning (IL) (Pomerleau, 1988; Bojarski et al., 2016; Codevilla et al., 2018; 2019; Zhou et al., 2019; Ohn-Bar et al., 2020; Prakash et al., 2021; Chitta et al., 2021) or reinforcement learning (RL) (Chen et al., 2021; Toromanoff et al., 2020), which are often brittle to o.o.d. inputs (Filos et al., 2020). To induce safety-critical situations, hand-crafted scenarios are typically added to the simulation. Unfortunately, the scenarios have to be manually re-tuned to each driving agent, limiting scalability.

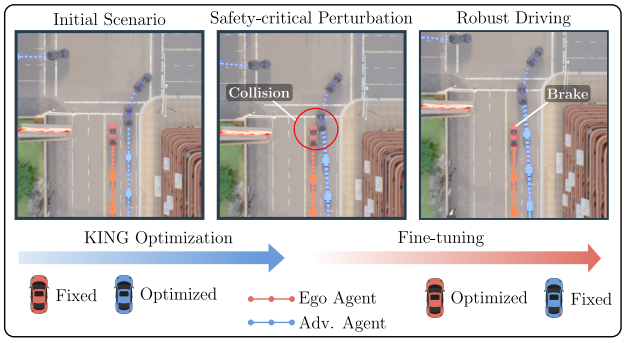


Figure 1. **Generating safety-critical scenarios.** Left: we propose KING, a novel gradient-based procedure for the generation of safety-critical perturbations of initial non-critical scenarios. Right: fine-tuning on these perturbations leads to a more robust agent.

Recent work (O' Kelly et al., 2018; Abeysirigoonawardena et al., 2019; Ding et al., 2020; 2021; Wang et al., 2021; Priisalu et al., 2022) has framed the problem of generating safety-critical scenarios through the lens of adversarial attacks, iteratively simulating the scenario and adjusting its parameters to increase a driving cost wrt. to the driving system under test. As simulators and self-driving stacks are often non-differentiable, these approaches have resorted to black-box optimization (BBO). In this work, we instead propose KING[1], a procedure that generates safety-critical scenarios via backpropagation. Through a simple approximation to the true gradient, KING can handle non-differentiable rendering functions and driving systems, while finding safety-critical perturbations more reliably than BBO-based alternatives. We use the generated scenarios fine-tune an end-to-end IL agent and show that this leads to improved robustness, reducing collisions by over 50%.

[1]Mercedes-Benz AG R&D [2]University of Tübingen [3]Max Planck Institute for Intelligent Systems. Correspondence to: Niklas Hanselmann <niklas.hanselmann@mercedes-benz.com>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

[1]https://lasnik.github.io/king/

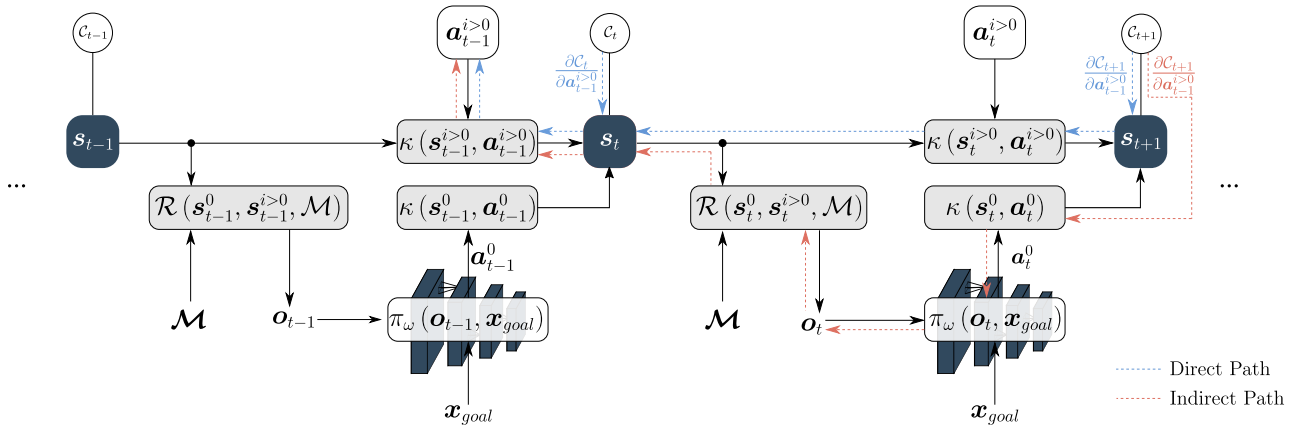


Figure 2. **Gradient paths.** To simulate a scenario, we render an observation $\mathbf{o}_t$ for the driving policy π_ω under attack using a rendering function $\mathcal{R}$. Both the driving policy and adversarial agents then take actions. The actions of the ego agent $\mathbf{a}_t^0$ depend on the observation and a goal location $\mathbf{x}_{goal}$. The actions of the adversarial agents $\mathbf{a}_t^{i>0}$ are the parameters to optimize over to a safety-critical perturbation. Given the actions for all agents and current traffic state $\mathbf{s}_t$, the next state $\mathbf{s}_{t+1}$ is computed using a differentiable kinematics model κ. Gradients from the cost at time t can then be propagated back to states at preceding timesteps. As shown, the derivative has components along two paths: an efficient direct path and a compute-intensive indirect path.

2. Safety-Critical Scenario Generation

We now outline our overall approach to the gradient-based generation of safety-critical scenarios for stress-testing and improving the robustness of IL-based driving agents.

Driving Agent. As the driving agents we consider (1) AIM-BEV, a neural planner acting on ground-truth bird's-eye view (BEV) visual abstractions similar to the AIM-VA model in (Chitta et al., 2021) and (2) TransFuser (Prakash et al., 2021), a state-of-the art image and LiDAR-based IL model. Formally, they are represented as a parameterized policy π_ω that takes in an observation $\mathbf{o}_t \in \mathbb{R}^{H_o \times W_o \times C_o}$ and goal location $\mathbf{x}_{goal} \in \mathbb{R}^2$ indicating the intended high-level route on the map, and plans a trajectory represented by four future 2D waypoints $\mathbf{w} \in \mathbb{R}^{4\times 2}$:

$$\pi_\omega\left(\mathbf{o}_t, \mathbf{x}_{goal}\right) : \mathbb{R}^{H_o \times W_o \times C_o} \times \mathbb{R}^2 \rightarrow \mathbb{R}^{4\times 2}. \quad (1)$$

For AIM-BEV, $\mathbf{o}_t$ is a BEV semantic occupancy grid encoding information on the road, lanes and other vehicles. For TransFuser it consists of camera and LiDAR data. Based on the predicted waypoints, the final actions $\mathbf{a}_t^0 \in [-1, 1]^2$ in the form of throttle and steering commands are produced by lateral and longitudinal controllers. Both models are trained on observation-waypoint pairs $(\mathbf{o}, \mathbf{w})$ drawn from a dataset $\mathcal{D}_{reg}$ of expert driving in regular traffic.

Safety-Critical Perturbation. To optimize for safety-critical perturbations of a non-critical scenario, we iteratively simulate it in closed-loop and adjust its parameters to be more challenging for the driving agent (or *ego agent*) under test. Importantly, as the scenario is simulated in closed loop, the ego agent can react to these perturbations.

Let $\mathcal{S} = \{\mathbf{s}_t\}_0^T$ be a sequence of traffic states instantiating a particular simulation, where s_t consists of the BEV position, orientation and speed of all agents at time t. Then, a simulation is unrolled based on a kinematics model $s_{t+1} = \kappa(a_t, s_t)$. Based on this framework, a scenario is parameterized by the sequence of actions $\{\mathbf{a}_t^{i>0}\}_t^T$ executed by other traffic participants (or *adversarial agents*), which determines their trajectory. The actions of the ego agent are obtained from the driving policy by rendering an observation $\mathbf{o}_t$ of the current state $\mathbf{s}_t$ using a rendering function $\mathbf{o}_t = \mathcal{R}\left(\mathbf{s}_t, \mathcal{M}\right)$, where $\mathcal{M}$ is a map describing the static aspects of the simulation. To find a safety-critical perturbation, the scenario's parameters are optimized to increase a driving cost $\mathcal{C}$ wrt. the ego agent, which is motivated by prior work (Abeysirigoonawardena et al., 2019; Ding et al., 2020; Wang et al., 2021):

$$\mathcal{C}(\mathcal{S}) = \phi_{col}^{ego}(\mathcal{S}) + \lambda\, \phi_{col}^{adv}(\mathcal{S}) + \gamma\, \phi_{dev}^{adv}(\mathcal{S}). \quad (2)$$

Here, collisions involving the ego agent are encouraged via an attractive potential ϕ_{col}^{ego}, and collisions between adversarial agents and deviations of the adversarial agents from the driveable area are discouraged via the repulsive potentials with ϕ_{col}^{adv} and ϕ_{dev}^{adv}. This is similar to commonly used cost functions in planning (Zeng et al., 2019; Sadat et al., 2020; Casas et al., 2021).

Kinematics Gradients: Given that the sequence of states $\mathcal{S}$ is unrolled based on the differentiable kinematics model, we can backpropagate costs at any timestep t to the set of actions $\{\mathbf{a}_{t-1}, \mathbf{a}_{t-2}, ..., \mathbf{a}_0\}$ at previous timesteps. In the full unrolled computation graph of the simulation, partial derivatives of the cost at any timestep can be taken wrt. the actions in preceding timesteps by recursively applying the chain rule along two paths: a direct path through only the kinematics model and an indirect path, which additionally

	1 Agent			2 Agents			4 Agents			Overall		
Method	**CR** ↑	$t_{50\%}$ ↓	**s/it** ↓	**CR** ↑	$t_{50\%}$ ↓	**s/it** ↓	**CR** ↑	$t_{50\%}$ ↓	**s/it** ↓	**CR** ↑	$t_{50\%}$ ↓	**s/it** ↓
Random Search	62.50	**9.25**	1.30	68.75	7.38	1.35	68.75	15.22	1.48	66.67	9.66	1.38
Bayesian Optimization	63.75	11.88	1.46	68.75	10.01	1.66	63.75	22.12	2.06	65.00	14.34	1.73
SimBA (Guo et al., 2019)	60.00	14.14	1.30	71.25	14.35	1.35	61.25	19.68	1.48	64.17	15.84	1.38
CMA-ES (Hansen & Ostermeier, 2001)	67.50	9.34	1.31	75.00	**6.73**	1.36	62.50	9.39	1.52	68.33	8.17	1.40
Bandit-TD (Ilyas et al., 2019)	37.50	-	3.87	30.00	-	4.39	21.25	-	5.02	29.58	-	4.43
KING Direct + Indirect	78.75	19.33	3.17	72.50	14.68	3.25	76.25	14.67	3.40	75.83	16.14	3.27
KING (Ours)	**86.25**	9.98	1.78	**82.50**	6.96	1.88	**78.75**	**6.40**	2.03	**82.50**	**7.78**	1.90

Table 1. **Critical scenario generation on CARLA.** Mean CR, $t_{50\%}$ and s/it for different optimization techniques in three traffic settings, as well as aggregated metrics. KING finds collisions in over 80% of the initializations, significantly outperforming all baselines. Using only the direct path (Ours) leads to the highest CR and is faster than using gradients from both the direct and indirect paths.

	Held-out KING scenarios	CARLA scenarios	
Dataset	**CR** ↓	**DS** ↑	**CR** ↓
No Fine-tuning	100.00±0.00	86.74±0.67	17.48±1.86
$\mathcal{D}_{reg}$	57.14±0.00	86.85±0.62	19.51±0.00
$\mathcal{D}_{crit} \cup \mathcal{D}_{reg}$	**28.57**±0.00	**90.20**±0.00	**8.13**±0.70

Table 2. **Robust training for AIM-BEV.** Results shown are the mean and std over 3 evaluation seeds.

involves the driving policy π_ω and renderer $\mathcal{R}$. This is illustrated in Fig. 2.

With KING, we propose an approximation to the true gradients, which only considers the direct path and stops gradients through the indirect path. While this introduces an error in the gradient estimation, we empirically find it to work well while leading to several advantages. Firstly, it enables gradient-based generation in the common case where the rendering function or driving policy is non-differentiable, preventing gradients to be taken wrt. the indirect path. Secondly, even when all components are differentiable, taking gradients wrt. to the indirect path involves backpropagating through the driving policy and rendering function (dotted red arrows in Fig. 2) - a significant computational overhead. We investigate this setting for AIM-BEV where both the driving policy and rendering function are differentiable in Section 3.1 and show that given a fixed computational budget, this overhead leads to results worse than KING. We hypothesize that utilizing gradients through both paths becomes more important as the driving policy becomes robust to attacks.

Robust Training for IL: We are further interested in improving robustness by augmenting the original training data with the generated safety-critical scenarios. To this end, we pursue a simple yet effective strategy: (1) we generate a large set of safety-critical scenarios, (2) we filter these for scenarios in which a priviliged rule-based expert algorithm finds a safe alternate trajectory, (3) we collect a dataset of observation-waypoint pairs $\mathcal{D}_{crit}$ using the expert, and (4) we fine-tune the policy π_ω on a mix of the safety-critical data $\mathcal{D}_{crit}$ and the original dataset $\mathcal{D}_{reg}$.

3. Experiments

We now present the research questions we aim to answer in our experimental study.

Can gradient-based attacks outperform black-box optimization (BBO) for safety-critical scenario generation? We are interested in reducing the optimization time needed to take a set of non-critical scenario initializations and find interesting scenarios. Given the computational overhead of computing gradients and performing a backward pass, we analyze the gains that can be achieved for this task with gradient-based attacks over BBO in Section 3.1.

Are gradient-based attacks applicable to non-differentiable simulators? Our main experiments are conducted using a differentiable simulator that renders the BEV grid inputs for AIM-BEV. In Section 3.1, we aim to investigate the applicability of KING to non-differentiable rendering functions, such as CARLA's camera and LiDAR sensors.

Can we improve robustness by augmenting the training distribution with critical scenarios? We are interested in the analyzing robustness of the fine-tuned IL model that uses the data augmentation strategy described in Section 2. In Section 3.2, we investigate this on both the regular benchmark (hand-crafted scenarios) and held-out safety-critical test scenarios generated by KING.

3.1. Comparison to BBO

In this section, we analyze the efficacy of KING for generating safety-critical scenarios, by comparing it with several BBO baselines.

Experimental Setup. As initializations we use 80 scenarios accross several CARLA towns with non-critical traffic that mimics the behaviour in CARLA. We explicitly control the traffic density at one, two and four adversarial agents and use a simulation horizon of 20 seconds. The scenario generation is evaluated using three metrics: (1) the collision rate (CR), which is the number of initializations for which

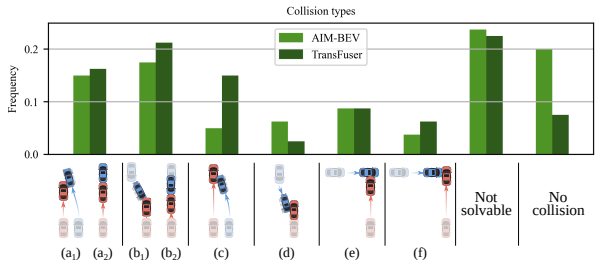
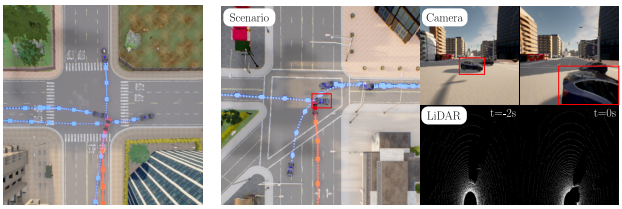


Figure 3. **Collision types.** We observe that KING generates a diverse set of challenging but solvable scenarios.



(a) AIM-BEV (b) TransFuser (Prakash et al., 2021)

Figure 4. **Qualitative examples of safety-critical scenarios generated by KING.** Ego agent in red, adversarial agent in blue. Best viewed zoomed in.

a critical perturbation was found within a computation budget of 180 GPU seconds (2) the time taken to achieve 50% CR ($t_{50\%}$) and (3) the runtime needed to complete one optimization step (s/it). We specifically opt for a computational budget in terms of wall clock time rather than iterations to account for the differing computational expense between the methods.

Results. We compare KING against several BBO baselines in Table 1. In particular, besides Random Search and Bayesian Optimization, we consider SimBA (Guo et al., 2019), CMA-ES (Hansen & Ostermeier, 2001) and Bandit-TD (Ilyas et al., 2019). SimBA is a variant of Random Search that greedily maximizes the objective and CMA-ES is a state-of-the-art evolutionary algorithm. Finally, Bandit-TD computes numerical gradients by integrating priors into a finite differences approach.

KING obtains a significantly higher CR than the BBO baselines in all traffic densities, increasing the number of scenarios for which a safety-critical perturbation is found by over 20%. Among the BBO baselines, CMA-ES attains the best overall scores with respect to both CR and $t_{50\%}$. Interestingly, as we increase the traffic density to four agents, there exists a large gap in the $t_{50\%}$ between KING and all baselines, highlighting the limitations of BBO in scaling to higher dimensional search spaces. Additionally, we find that our proposed gradient approximation is reasonable, and due to the lower computational expense even outperforms the exact gradient (direct + indirect path) given the same computational budget.

Analysis of Safety-Critical Scenarios. In this section, we provide additional analysis on the safety-critical scenarios generated by KING for both AIM-BEV and TransFuser. Specifically, we show the distribution of the resulting scenarios with a traffic density of $N = 4$ agents in Fig. 3. For both driving agents, we first filter out the set of scenarios where KING is unable to find a collision ("No Collision") as well as those that are not solvable by the rule-based expert ("Not Solvable"). We cluster the remaining scenarios using k-means (similar to (Rempe et al., 2021)) to obtain 6 clusters of failure modes such as cut-ins (a_1), rear-ends (a_2) and unsafe behavior in unprotected turns (e,f). This highlights the diversity of the generated scenarios. Additionally, from the frequency of scenarios with "No collision" in Fig. 3, we observe that both AIM-BEV and TransFuser collide in at least 80% of the scenarios. Finally, the large amount of collisions for TransFuser indicate that KING can achieve promising results when applied out-of-the-box to driving simulators with non-differentiable rendering functions.

3.2. Evaluating Robustness after Fine-Tuning

Finally, we analyze the utility of the generated scenarios in augmenting the original, non-critical training distribution to yield more robust driving agents.

Experimental Setup. We use 300 scenarios generated by KING, from which we hold out 20% for evaluation, and fine-tune AIM-BEV as described in Section 2. We evaluate the driving performance in terms of Collision Rate (CR) and Driving Score (DS), the official CARLA Leaderboard metric, on held-out KING scenarios as well as CARLA's hand-crafted scenarios in a dense urban setting. As baselines we use the original model, as well as one that has been fine-tuned on regular data only. This is intended to show the effect of adopting an overall different - but not more robust - driving style on the KING scenarios, which remain fixed after optimization.

Results. As shown in Table 2, the simple strategy of fine tuning on a mixture of critical and regular data $\mathcal{D}_{crit} \cup \mathcal{D}_{reg}$ is an effective way of learning from the scenarios generated by KING. After fine-tuning, AIM-BEV shows significantly safer driving, reducing the collision rates on CARLA scenarios by over 50%.

4. Conclusion

We propose a novel gradient-based safety-critical scenario generation procedure, KING, which achieves significantly higher success rates compared to existing BBO-based attacks while being more efficient. By augmenting the training data with scenarios from KING, we are able to significantly improve the collision avoidance of an imitation learning-based driving agent.

References

Abeysirigoonawardena, Y., Shkurti, F., and Dudek, G. Generating adversarial driving scenarios in high-fidelity simulators. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*, 2019.

Bojarski, M., Testa, D. D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., Zhang, J., Zhang, X., Zhao, J., and Zieba, K. End to end learning for self-driving cars. *arXiv.org*, 1604.07316, 2016.

Casas, S., Sadat, A., and Urtasun, R. Mp3: A unified model to map, perceive, predict and plan. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Chen, D., Koltun, V., and Krähenbühl, P. Learning to drive from a world on rails. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2021.

Chitta, K., Prakash, A., and Geiger, A. Neat: Neural attention fields for end-to-end autonomous driving. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2021.

Codevilla, F., Miiller, M., López, A., Koltun, V., and Dosovitskiy, A. End-to-end driving via conditional imitation learning. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*, 2018.

Codevilla, F., Santana, E., López, A. M., and Gaidon, A. Exploring the limitations of behavior cloning for autonomous driving. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019.

Ding, W., Xu, M., and Zhao, D. Learning to collide: An adaptive safety-critical scenarios generating method. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*, 2020.

Ding, W., Chen, B., Li, B., Eun, K. J., and Zhao, D. Multimodal safety-critical scenarios generation for decision-making algorithms evaluation. *IEEE Robotics and Automation Letters (RA-L)*, 6(2):1551–1558, 2021.

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., and Koltun, V. CARLA: An open urban driving simulator. In *Proc. Conf. on Robot Learning (CoRL)*, 2017.

Filos, A., Tigas, P., McAllister, R., Rhinehart, N., Levine, S., and Gal, Y. Can autonomous vehicles identify, recover from, and adapt to distribution shifts? In *Proc. of the International Conf. on Machine learning (ICML)*, 2020.

Guo, C., Gardner, J. R., You, Y., Wilson, A. G., and Weinberger, K. Q. Simple black-box adversarial attacks. In *Proc. of the International Conf. on Machine learning (ICML)*, 2019.

Hansen, N. and Ostermeier, A. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 2001.

Ilyas, A., Engstrom, L., and Madry, A. Prior convictions: Black-box adversarial attacks with bandits and priors. In *Proc. of the International Conf. on Learning Representations (ICLR)*, 2019.

Janai, J., Güney, F., Behl, A., and Geiger, A. *Computer Vision for Autonomous Vehicles: Problems, Datasets and State of the Art*, volume 12. Foundations and Trends in Computer Graphics and Vision, 2020.

Norden, J., O'Kelly, M., and Sinha, A. Efficient black-box assessment of autonomous vehicle safety. *arXiv.org*, 1912.03618, 2019.

Ohn-Bar, E., Prakash, A., Behl, A., Chitta, K., and Geiger, A. Learning situational driving. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

O' Kelly, M., Sinha, A., Namkoong, H., Tedrake, R., and Duchi, J. C. Scalable end-to-end autonomous vehicle testing via rare-event simulation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Pomerleau, D. ALVINN: an autonomous land vehicle in a neural network. In *Advances in Neural Information Processing Systems (NIPS)*, 1988.

Prakash, A., Chitta, K., and Geiger, A. Multi-modal fusion transformer for end-to-end autonomous driving. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Priisalu, M., Pirinen, A., Paduraru, C., and Sminchisescu, C. Generating scenarios with diverse pedestrian behaviors for autonomous vehicle testing. In *Proc. Conf. on Robot Learning (CoRL)*, 2022.

Rempe, D., Philion, J., Guibas, L. J., Fidler, S., and Litany, O. Generating useful accident-prone driving scenarios via a learned traffic prior. In *arXiv.org*, volume 2112.05077, 2021.

Sadat, A., Casas, S., Ren, M., Wu, X., Dhawan, P., and Urtasun, R. Perceive, predict, and plan: Safe motion planning through interpretable semantic representations. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2020.

Toromanoff, M., Wirbel, E., and Moutarde, F. End-to-end model-free reinforcement learning for urban driving using implicit affordances. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Wang, J., Pun, A., Tu, J., Manivasagam, S., Sadat, A., Casas, S., Ren, M., and Urtasun, R. Advsim: Generating safety-critical scenarios for self-driving vehicles. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Zeng, W., Luo, W., Suo, S., Sadat, A., Yang, B., Casas, S., and Urtasun, R. End-to-end interpretable neural motion planner. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Zhou, Y., Sun, P., Zhang, Y., Anguelov, D., Gao, J., Ouyang, T., Guo, J., Ngiam, J., and Vasudevan, V. End-to-end multi-view fusion for 3d object detection in lidar point clouds. In *Proc. Conf. on Robot Learning (CoRL)*, 2019.