CLEAR: Closed-Loop Reinforcement Learning at Scale for End-to-End Autonomous Driving

Yunxiao Shi✉ Hong Cai✉ Mohammad Ghavamzadeh Fatih Porikli
Qualcomm AI Research
{yunxshi, hongcai}@qti.qualcomm.com
Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc

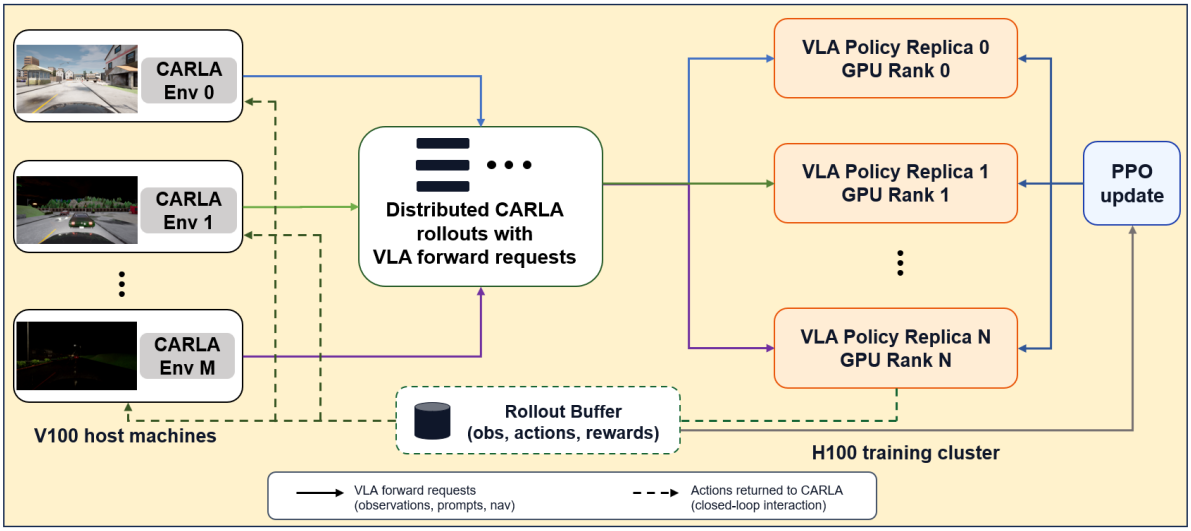

Figure 1. CLEAR system pipeline. We conduct closed-loop RL finetuning on top of VLAs pretrained using imitation learning. With our heterogeneous finetuning pipeline, we dramatically scale up the number of simulators [15] running in parallel and PPO [28, 36] updates.

1. Introduction

End-to-end autonomous driving (E2E-AD) [7, 8, 18, 23, 24] aims to directly map raw sensor information to driving actions, and has emerged as a powerful alternative to traditional modular approaches [10, 11, 17, 21, 29]. Recently, with the rapid advancement of multi-modal large language models (MLLMs) [1, 2, 9, 12, 31, 34], researchers have proposed the paradigm of Vision-Language-Action (VLA) models for E2E-AD, where it seeks to integrate visual perception, language understanding and action prediction within a single policy, and has demonstrated strong results on various benchmarks [4–6, 14, 22, 32].

Existing VLA-based policies [16, 19, 27, 38] largely adopts imitation learning (IL) as the primary approach, *i.e.*, the learning objective typically is only to optimize certain distance metrics (*e.g.*, L_2 error) *w.r.t.* ground-truth expert trajectories. The benefit of IL is that it can scale with data. However, the distribution shift between open-loop training and closed-loop inference often leads to compromised policies, exhibiting suboptimal performance in closed-loop planning.

A recent line of research [20, 25, 26, 37] explores closed-loop training using Reinforcement Learning (RL) [33]. However, they only consider the task of privileged planning, where perfect perception ground-truths (*e.g.* 3D boxes [26], BEV segmentation maps [20]) and global visibility of the environment (*e.g.* HD/SD maps) are required as inputs, which is unrealistic to assume that such information is always available.

In this work, we propose CLEAR, a system that effectively enables and efficiently performs closed-loop RL finetuning of VLA policies for E2E-AD. Given the size of VLA models, carrying out RL finetuning from scratch is often intractable in terms of the amount of compute needed. Therefore, we first pretrain CLEAR on a large set of expert trajectories using imitation learning, equipping it with baseline

Figure 2. Qualitative predictions of CLEAR on the CARLA longest6 v2 [20, 35] benchmark. Best viewed in color and zoomed in.

driving capabilities. Then, we freeze the VLA agent except the action head, which outputs driving actions, on top of which we finetune with Proximal Policy Optimization (PPO) [28], given its stable convergence and high asymptotic performance. A summary of the design of CLEAR is shown in Fig. 1.

2. Method

We detail the design of our CLEAR in the sections below.

VLA Open-Loop Pretraining. We use InternVL3-1B [40] as the backbone of our VLA driving policy, which consists of an InternViT-300M-448px vision encoder [9], and a Qwen2.5-0.5B [3] large language model (LLM). We extract the visual tokens as $v_e = \delta_{\downarrow}([\mathrm{V}_e(I_i)]_{i=0}^{N_i} \in \mathbb{R}^{(N_i M)\times d}$, where I_i are the input image tiles, V_e is the vision encoder, $\delta_{\downarrow}(\cdot)$ is the downsampling operator [9, 27, 30] which downsamples the number of tokens to $M = 256$. Here we set the number of image tiles $N_i = 2$ and d is the embedding dimension. Denote L_e and Nav_e the embeddings of the rest input modalities, we generate waypoints prediction as $W = \mathrm{MLP}(\mathrm{LLM}([\{v_e, L_e, Nav_e\}, q_w, q_p]))$, where q are the action queries introduced in [27].

Closed-Loop RL Finetuning. We make use of the CARLA driving simulator [15] release 0.9.15 to conduct closed-loop RL finetuning. Given the compute limitations, we freeze the vision encoder and LLM in the pretrained VLA entirely. Since CARLA environments expect driving actions, we replace the waypoints head with a small action head, which predicts steering $s \in [0, 1]$, and throttle/brake which are merged into a single action dimension $tb \in [-1, 1]$ since they are mutually exclusive. For reward design, we adopt the discounted Route Completion (RC) [20, 35] $r_t = RC_t \cdot (\Pi p_t) - T$ as our reward, where t denotes each time step, $p_t \in [0, 1]$ is the soft penalty factor, and $T = \{0, 1\}$ is the hard penalty (*e.g.*, collision, red light/stop sign infraction, etc) which terminates an episode. We utilize DD-PPO [36] to scale up PPO updates under a distributed setting. To further improve efficiency, we also disable the autoregressive language prediction path in the pretrained VLA during the RL finetuning stage.

Heterogeneous Finetuning Pipeline. The GPU rendering stack of CARLA is known for its instability and brittleness [13, 22]. We observed frequent CARLA server crashes during RL training especially in containerized environments (*e.g.*, docker). In addition, a single CARLA server takes about 6GB of GPU memory to initialize, which carves into the total memory budget (*e.g.*, an Hopper H100 GPU with 80GB of GPU memory) that the VLA agent also consumes. This prevents the scaling of the number of CARLA environments and also causes training instability. Therefore, we design a heterogeneous finetuning pipeline where we separate server (CARLA) and client (VLA policy). As shown in Fig. 1, we launch CARLA servers on standalone host machines (*e.g.*, carrying Tesla V100 GPUs), and put the VLA learner on high-performance H100 clusters. We construct SSH tunnels [39] to establish communication between server and client to facilitate learning. With such heterogeneous design, we are able to run 64 CARLA servers in parallel, and scale to 40M samples with a total batch size/mini-batch size of 16384/4096 for PPO updates, which lead to superior results.

3. Experiments

Datasets. We leverage the dataset collected by [27] for VLA pretraining, which totals 3.1 million samples at 4ps. For RL finetuning, we make use of the procedurally generated routes from a suite of parallel CARLA environments by [20] to perform rollouts. We evaluate on the CARLA longest6 v2 [20, 35] benchmark, which consists of 36 challenging long driving routes averaging 1-2 kilometers.

Table 1. Closed-loop planning performance on the CARLA longest6 v2 benchmark [20, 35]. NP stands for "Non-Privileged".

Method	Type	DS↑	SR(%)↑
InternVL2-1B	NP	13.11	0.00
InternVL3-1B	NP	18.43	0.00
CLEAR (InternVL2-1B)	NP	28.63	16.67
CLEAR (InternVL3-1B)	NP	33.91	19.44

Results. We summarize the evaluation results in Table 1. It is evident that even after large-scale imitation learning during the pretraining stage, the VLA agents perform poorly on the longest6 v2 benchmark [20, 35], with a 0% Success Rate (failing every route) due to long driving distances with complex scenarios. But after we conduct RL finetuning, CLEAR improves the performance of the VLA polices across the board, with significant jumps both in Driving Score and Success Rate, leading to a much more capable driver. This validates the effectiveness of the post-training RL framework proposed in our CLEAR.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1

[2] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 1

[3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Dang Kai, Peng Wang, Wang Shijie, Tang Jun, Zhong Humen, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Zhang Xi, Xie Tianbao, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 2

[4] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 1

[5] Holger Caesar, Juraj Kabzan, Kok Seang Tan, Whye Kit Fong, Eric Wolff, Alex Lang, Luke Fletcher, Oscar Beijbom, and Sammy Omari. nuplan: A closed-loop ml-based planning benchmark for autonomous vehicles. *arXiv preprint arXiv:2106.11810*, 2021.

[6] Wei Cao, Marcel Hallgarten, Tianyu Li, Daniel Dauner, Xunjiang Gu, Caojun Wang, Yakov Miron, Marco Aiello, Hongyang Li, Igor Gilitschenski, et al. Pseudo-simulation for autonomous driving. *Conference on Robot Learning*, 2025. 1

[7] Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):10164–10183, 2024. 1

[8] Shaoyu Chen, Bo Jiang, Hao Gao, Bencheng Liao, Qing Xu, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. *arXiv preprint arXiv:2402.13243*, 2024. 1

[9] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198, 2024. 1, 2

[10] Kashyap Chitta, Aditya Prakash, and Andreas Geiger. Neat: Neural attention fields for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15793–15803, 2021. 1

[11] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE transactions on pattern analysis and machine intelligence*, 45(11):12878–12895, 2022. 1

[12] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 1

[13] The CARLA community. The carla issues section. *https://github.com/carla-simulator/carla/issues*, 2022. 2

[14] Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, et al. Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. *Advances in Neural Information Processing Systems*, 37:28706–28719, 2024. 1

[15] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR, 2017. 1, 2

[16] Haoyu Fu, Diankun Zhang, Zongchuang Zhao, Jianfeng Cui, Dingkang Liang, Chong Zhang, Dingyuan Zhang, Hongwei Xie, Bing Wang, and Xiang Bai. Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 24823–24834, 2025. 1

[17] Jeffrey Hawke, Richard Shen, Corina Gurau, Siddharth Sharma, Daniele Reda, Nikolay Nikolov, Przemysław Mazur, Sean Micklethwaite, Nicolas Griffiths, Amar Shah, et al. Urban driving with conditional imitation learning. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 251–257. IEEE, 2020. 1

[18] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17853–17862, 2023. 1

[19] Jyh-Jing Hwang, Runsheng Xu, Hubert Lin, Wei-Chih Hung, Jingwei Ji, Kristy Choi, Di Huang, Tong He, Paul Covington, Benjamin Sapp, et al. Emma: End-to-end multimodal model for autonomous driving. *arXiv preprint arXiv:2410.23262*, 2024. 1

[20] Bernhard Jaeger, Daniel Dauner, Jens Beißwenger, Simon Gerstenecker, Kashyap Chitta, and Andreas Geiger. Carl: Learning scalable planning policies with simple rewards. *arXiv preprint arXiv:2504.17838*, 2025. 1, 2

[21] Xiaosong Jia, Penghao Wu, Li Chen, Jiangwei Xie, Conghui He, Junchi Yan, and Hongyang Li. Think twice be-

fore driving: Towards scalable decoders for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21983–21994, 2023. 1

[22] Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. *Advances in Neural Information Processing Systems*, 37:819–844, 2024. 1, 2

[23] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8350, 2023. 1

[24] Peidong Li and Dixiao Cui. Navigation-guided sparse scene representation for end-to-end autonomous driving. In *International Conference on Learning Representations (ICLR)*, 2025. 1

[25] Qifeng Li, Xiaosong Jia, Shaobo Wang, and Junchi Yan. Think2drive: Efficient reinforcement learning by thinking with latent world model for autonomous driving (in carla-v2). In *European conference on computer vision*, pages 142–158. Springer, 2024. 1

[26] Katrin Renz, Kashyap Chitta, Otniel-Bogdan Mercea, A Koepke, Zeynep Akata, and Andreas Geiger. Plant: Explainable planning transformers via object-level representations. *Conference on Robotic Learning*, 2022. 1

[27] Katrin Renz, Long Chen, Elahe Arani, and Oleg Sinavski. Simlingo: Vision-only closed-loop autonomous driving with language-action alignment. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11993–12003, 2025. 1, 2

[28] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 1, 2

[29] Hao Shao, Letian Wang, Ruobing Chen, Hongsheng Li, and Yu Liu. Safety-enhanced autonomous driving using interpretable sensor fusion transformer. In *Conference on Robot Learning*, pages 726–737. PMLR, 2023. 1

[30] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016. 2

[31] Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025. 1

[32] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020. 1

[33] Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*. MIT press Cambridge, 1998. 1

[34] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1

[35] The CARLA team. The carla leaderboard 2.0. *https://leaderboard.carla.org*, 2022. 2

[36] Erik Wijmans, Abhishek Kadian, Ari Morcos, Stefan Lee, Irfan Essa, Devi Parikh, Manolis Savva, and Dhruv Batra. Dd-ppo: Learning near-perfect pointgoal navigators from 2.5 billion frames. *International Conference on Learning Representations (ICLR)*, 2020. 1, 2

[37] Penghao Wu, Xiaosong Jia, Li Chen, Junchi Yan, Hongyang Li, and Yu Qiao. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. *Advances in Neural Information Processing Systems*, 35:6119–6132, 2022. 1

[38] Shuo Xing, Chengyuan Qian, Yuping Wang, Hongyuan Hua, Kexin Tian, Yang Zhou, and Zhengzhong Tu. Openemma: Open-source multimodal model for end-to-end autonomous driving. In *Proceedings of the Winter Conference on Applications of Computer Vision*, pages 1001–1009, 2025. 1

[39] Tatu Ylonen and Chris Lonvick. The secure shell (ssh) connection protocol. RFC 4254 (Standards Track), 2006. 2

[40] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 2