HMAD: Advancing E2E Driving with Anchored Offset Proposals and Simulation-Supervised Multi-target Scoring

Bin Wang[1], Pingjun Li[1], Jinkun Liu[2], Jun Cheng[1], Hailong Lei[1], Yinze Rong[2],
Huan-ang Gao[2], Kangliang Chen[1], Xing Pan[1], Weihao Gu[1,†]

[1]HAOMO.AI Technology Co., Ltd [2]Tsinghua University

{wangbin, guweihao}@haomo.ai

Abstract

End-to-end autonomous driving faces persistent challenges in both generating diverse, rule-compliant trajectories and robustly selecting the optimal path from these options via learned, multi-faceted evaluation. To address these challenges, we introduce HMAD, a framework integrating a distinctive Bird's-Eye-View (BEV) based trajectory proposal mechanism with learned multi-criteria scoring. HMAD leverages BEVFormer and employs learnable anchored queries, initialized from a trajectory dictionary and refined via iterative offset decoding (inspired by DiffusionDrive), to produce numerous diverse and stable candidate trajectories. A key innovation, our simulation-supervised scorer module, then evaluates these proposals against critical metrics including no at-fault collisions, drivable area compliance, comfortableness, and overall driving quality (i.e., extended PDM score). Demonstrating its efficacy, HMAD achieves a 44.5% driving score on the CVPR 2025 private test set. This work highlights the benefits of effectively decoupling robust trajectory generation from comprehensive, safety-aware learned scoring for advanced autonomous driving.

1. Introduction

End-to-end autonomous driving, aspiring to learn a direct mapping from sensor observations to driving actions [2, 3, 6, 12–14, 21, 25], presents a compelling vision for a unified and potentially more robust alternative to traditional modular pipelines. However, initial end-to-end paradigms, often centered on direct imitation learning to regress a single trajectory, fundamentally struggled with the inherent multimodality of real-world driving. This critical flaw frequently led to "mode collapse"—severely limiting the agent's capacity to explore diverse, valid driving options—and resulted in a pronounced brittleness in complex scenarios requiring nuanced adherence to varied traffic constraints, far beyond what simple behavioral cloning could achieve.

Recognizing these limitations, subsequent research increasingly explored multimodal planning, focusing on generating a diverse set of potential trajectories. While an advancement, the crucial step of selecting the optimal trajectory often remained a significant bottleneck. Many such systems reverted to employing separate, often non-differentiable, post-processing modules laden with predefined heuristics or fixed-weight cost functions [7–9, 13, 15–17, 22–24, 26]. This architectural choice not only fragments the end-to-end learning process, impeding true joint optimization across the perception-to-planning pipeline, but also inherently limits the system's adaptability, as fixed rules struggle to generalize across the vast spectrum of dynamic driving contexts. This critical gap highlights an urgent need for frameworks capable of both generating truly diverse, high-quality trajectories and evaluating them through an integrated, learned, and context-aware multi-faceted mechanism, ensuring decisions robustly balance all critical aspects of driving.

In this work, we introduce *HMAD*, a novel motion planning framework for the CVPR 2025 End-to-End Driving Challenge, addressing prior shortcomings by uniquely combining a distinct trajectory proposal strategy with a learned, simulation-supervised multi-criteria scoring module. First, BEVFormer [19] constructs rich Bird's-Eye-View (BEV) representations. Upon this BEV context, HMAD generates diverse and stable candidate trajectories—combating mode collapse—using learnable anchored queries (from a trajectory dictionary) refined by an iterative offset decoding process inspired by DiffusionDrive [20]. Critically, and in contrast to methods using inflexible heuristics or separate ranking modules, these diverse proposals are then evaluated by a sophisticated, fully differentiable score module. This scorer, inspired by the principle of simulation-supervised multi-target evaluation (e.g., Hydra-MDP [18]), performs cross-attention with BEV features to predict key interpretable driving scores (e.g., extended PDM score, no
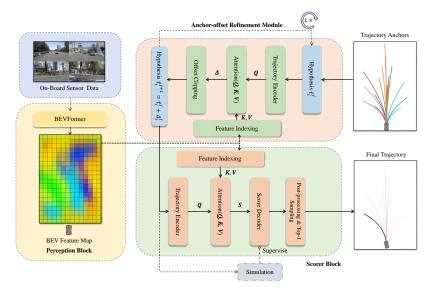
Figure 1. The Overall Architecture of HMAD.

at-fault collisions, drivable area compliance, and driving comfort) from simulator ground-truth, enabling learned, context-dependent trade-offs for nuanced selection within an end-to-end framework.

Furthermore, recognizing that many planning failures are concentrated in long-tail scenarios—such as unprotected turns, occluded junctions, sharp curves, and lane departures—we integrate hard case mining into our training regimen. This targeted data augmentation significantly enhances model robustness and its ability to generalize to these critical edge cases. Our system achieves a 44.5% driving score on the CVPR 2025 private test set, demonstrating the efficacy of our approach.

In summary, our contributions are as follows: **1)** We propose a BEV-based end-to-end driving framework featuring a distinctive trajectory generation mechanism that uses trajectory dictionary-initialized learnable queries and anchor-based offset decoding (inspired by DiffusionDrive [20]) to effectively overcome common issues of mode collapse and proposal instability. **2)** We design a dedicated, simulation-supervised scoring network that outputs interpretable driving metrics for multiple crucial criteria, enabling adaptive, learned trajectory ranking in contrast to fixed-heuristic methods. **3)** We enhance model robustness and generalization in complex urban scenarios through targeted hard example mining.

2. Method

2.1. BEV Representation from Multi-Camera Input

We adopt BEVFormer [19] as the perception backbone to transform multi-view camera inputs into a unified bird's-eye-view (BEV) representation. Given a set of time-synchronized images from surround-view cameras, BEV-Former first extracts image features using a shared backbone (e.g., ResNet-34 [11] + FPN), then uses deformable attention and a spatiotemporal transformer to fuse multi-camera and multi-frame features into a consistent BEV feature map $\mathcal{F}$. This BEV representation captures both spatial layout and temporal dynamics of the scene and serves as the queryable memory for downstream trajectory prediction and scoring modules.

2.2. BEV-aware Trajectory Decoding

Our trajectory generation process relies on learnable queries that interact with Bird's-Eye-View (BEV) features. These queries initialize diverse trajectory candidates, which are then refined through an iterative, BEV-aware decoding mechanism.

Learnable Anchored Query. We designed a set of learnable queries that are anchored to typical driving maneuvers. This approach aims to (1) enhance training convergence and (2) improve the diversity and coverage of the proposed candidate trajectories. These queries originate from a pre-constructed anchor trajectory dictionary:

$$\mathcal{A} = \left\{ a_i \in \mathbb{R}^{T \times 3} | i = 1, \dots N \right\} \quad (1)$$

where T denotes the number of future time steps and N is the total number of anchor trajectories. The dictionary is constructed by applying unsupervised clustering (e.g., K-means) to spatial positions of driving trajectories. Each anchor trajectory $a_i \in \mathcal{A}$ is further encoded into a fixed-dimensional embedding vector $\mathbf{Q}_i \in \mathbb{R}^d$ using a shared trajectory encoder.,

$$\mathbf{Q}_i = \text{TrajEnc}(\tau_i) \quad (2)$$

These embeddings are later used as inputs to the iterative refinement module described next.

Iterative Anchor-Offset Refinement with BEV Awareness. Instead of directly regressing full trajectory coordinates, our decoder refines trajectories by predicting offsets from their corresponding anchors a_i. This is accomplished using a multi-layer decoder architecture, where each layer iteratively improves the trajectory estimate based on BEV features.

Formally, given a trajectory query $\mathbf{Q}_i$, we initialize the trajectory hypothesis as its corresponding anchor $\hat{\tau}_i^0 = a_i$, and refine it over L decoder layers by predicting residual offsets $\Delta_i^j \in \mathbb{R}^{T \times 3}$ at each trajectory-oriented attention decoder layer $j = 0, 1 \ldots L - 1$:

$$\begin{cases} \Delta_i^j = \mathcal{D}_j(\mathbf{Q}_i, \hat{\tau}_i^j, \mathcal{F}) \\ \hat{\tau}_i^{j+1} = \hat{\tau}_i^j + \Delta_i^j \end{cases} \quad (3)$$

The core of each decoder module $\mathcal{D}_j(\cdot)$ is a trajectory-oriented attention mechanism that makes the refinement BEV-aware. For the current trajectory hypothesis $\hat{\tau}_i^j$ at layer j: (1) BEV context features relevant to $\hat{\tau}_i^j$ are gathered from $\mathcal{F}$ using a sampling function $\mathcal{G}(\cdot)$, yielding a feature sequence $\mathbf{f}_i^j = \mathcal{G}(\mathcal{F}, \hat{\tau}_i^j)$. This function $\mathcal{G}(\cdot)$ typically indexes features from $\mathcal{F}$ at multiple points along the path of $\hat{\tau}_i^j$. (2) This sampled feature sequence $\mathbf{f}_i^j$ is then projected to produce attention keys $\mathbf{K}_i^j$ and values $\mathbf{V}_i^j$. (3) The residual offset Δ_i^j is then decoded using an attention mechanism where the anchor-derived query $\mathbf{Q}_i$ attends to these BEV-informed keys and values:

$$\begin{cases} \mathbf{f}_i^j = \mathcal{G}(\mathcal{F}, \hat{\tau}_i^j) \\ \mathbf{K}_i^j, \mathbf{V}_i^j = \text{Proj}(\mathbf{f}_i^j) \\ \Delta_i^j = \text{Attn}(\mathbf{Q}_i, \mathbf{K}_i^j, \mathbf{V}_i^j) \end{cases} \quad (4)$$

To ensure training stability and prevent unrealistic trajectory modifications, the predicted offsets Δ_i^j are clipped to a predefined range: $\Delta_i^j \leftarrow \text{clip}(\Delta_i^j, -\delta_{\max}, \delta_{\max})$, where $\delta_{\max}$ is the maximum allowed displacement per layer.

This anchor-offset formulation offers several key benefits: (1) It leverages prior knowledge of common driving behaviors via anchors, while the iterative, BEV-aware attention allows for adaptive refinement based on the observed scene context, mimicking how human drivers adjust their intentions. (2) It naturally supports multi-modality through the use of diverse anchors, and the offset-based decoding, by flexibly refining each candidate, helps mitigate mode collapse often seen with direct trajectory regression. (3) It provides a structured and bounded prediction target, which, along with offset clipping, stabilizes training and improves data efficiency, especially when dealing with complex or rare driving scenarios.

2.3. Simulation-Supervised Trajectory Scoring

Selecting an optimal driving trajectory from multiple proposals requires a comprehensive evaluation across various, often conflicting, criteria encompassing safety, rule compliance, and overall driving quality. A single, monolithic score might fail to capture these diverse aspects. Therefore, to achieve a more nuanced and interpretable assessment, inspired by Hydra-MDP [18], we design a scorer network that learns to predict multiple, distinct aspects of trajectory performance in a multi-task fashion. Specifically, we utilize a simulator to generate detailed, ground-truth evaluations for each specific metric. Our scorer takes each predicted trajectory $\hat{\tau}_k$, encodes it using a trajectory embedding module (i.e., positional encoding + MLP), and then performs cross-attention with the BEV features to incorporate rich contextual information from the scene. The scorer is trained to predict several key metrics, including overall trajectory quality (i.e., the Extended PDM Score), collision avoidance, drivable area compliance, and comfortableness. Finally, at inference, the metric of overall trajectory quality determines the selection of the top-ranked trajectory.

2.4. Data Augmentation and Post-processing

To improve performance in challenging driving scenarios and ensure the selection of a single, optimal trajectory, we employ **targeted data augmentation** and a rigorous **post-selection process** in our submitted solution.

Firstly, to enhance model robustness, we perform *Hard Case Mining* by identifying difficult situations like unprotected turns, occluded junctions, sharp curves, and lane departures using human trajectory data and annotations from OpenScene [4]. These identified scenarios are then upsampled threefold during training to ensure the model learns effective strategies for these critical yet less frequent events.

Secondly, to ensure that the final chosen trajectory from the multiple candidates is verifiably safe and feasible in the immediate environment, we perform a detailed *post-processing step* in the 2D image space. This step aims to ground the planned trajectories against direct visual evidence. The method begins by estimating a valid driving distance envelope—both minimum and maximum travel distances—based on the ego vehicle's current speed and acceleration. Each proposed trajectory, along with a projected band representing the ego vehicle's width, is then mapped onto the 2D camera image space, implicitly using perspective transformation for accurate representation. Concurrently, the open-source model YOLOPv2 [10] analyzes the 2D image to identify the positions of critical lane lines and obstacles. Trajectories are subsequently filtered based on stringent criteria: any trajectory whose projected path falls outside the pre-calculated valid distance envelope, or whose ego-vehicle width band intersects with detected obstacles or deviates from the drivable area defined by lane lines, is dis-

Table 1. Ablation Results on the *"warmup two stage"* benchmark. All scores are in %. The number of decoding layers in the full model is 2, while conducting post-processing. Best scores for each metric are in **bold**. For the EPDMS column, values in red parentheses indicate the performance drop compared to the Full model.

Setup	Backbone	NC	DAC	DDC	TLC	EP	TTC	LK	HC	EC	EPDMS
Full model	ResNet34	**98.65**	**91.36**	**99.34**	**98.75**	64.93	**98.52**	75.29	93.32	53.82	**65.94**
- Hard case mining	ResNet34	89.89	80.70	91.87	98.39	**83.16**	88.70	80.07	**100.00**	66.20	59.01 (-6.93)
- Scorer	ResNet34	82.28	81.42	91.23	97.46	78.94	79.06	71.46	**100.00**	60.33	50.31 (-15.63)
#Decoder layer=1	ResNet34	91.68	81.98	94.78	96.79	78.19	88.11	76.37	**100.00**	70.08	59.00 (-6.94)
#Decoder layer=2	ResNet34	90.81	82.04	95.16	97.45	80.10	90.56	**80.14**	**100.00**	80.87	62.54 (-3.40)
#Decoder layer=4	ResNet34	89.32	83.99	96.01	97.60	75.64	88.60	72.98	**100.00**	**83.95**	59.25 (-6.69)

carded. From the remaining pool of valid trajectories, the one with the highest score, as assigned by our trajectory scorer module, is selected as the final output. Should this rigorous 2D filtering yield no compliant trajectories, a fallback to the highest-scoring trajectory from the original set is used to ensure continuous operation.

3. Experiment

3.1. Dataset and metrics

Dataset. We conduct our experiments mainly on the NAVSIM [5] dataset, which is specifically curated for evaluating end-to-end autonomous driving in scenarios requiring complex decision-making. NAVSIM is built upon the OpenScene [4] dataset, a compact and filtered version of nuPlan [1], retaining only essential sensor data and annotations sampled at 2 Hz.

Metrics. To evaluate planning performance in a consistent and safety-critical manner, our approach adopts the Extended Predictive Driver Model Score (EPDMS), introduced in NAVSIM v2. The EPDMS metric extends the original PDMS from NAVSIM v1 by incorporating additional sub-metrics and introducing mechanisms for more robust, realistic evaluation. In total, EPDMS includes four multiplicative penalty terms—No At-Fault Collisions (NC), Drivable Area Compliance (DAC), Driving Direction Compliance (DDC), Traffic Light Compliance (TLC), and false-positive filtering—and five weighted average metrics: Ego Progress (EP), Time-To-Collision (TTC), History Comfort (HC), Lane Keeping (LK), and Extended Comfort (EC). Formally, EPDMS is computed as:

$$\text{EPDMS} = \left(\prod_{m \in \{\text{NC,DAC,DDC,TLC}\}} \text{filter}_m(\text{agent, human}) \right) \cdot$$
$$\left(\frac{\sum_{m \in \{\text{TTC,EP,HC,LK,EC}\}} w_m \cdot \text{filter}_m(\text{agent, human})}{\sum_{m \in \{\text{TTC,EP,HC,LK,EC}\}} w_m} \right)$$
(5)

where the filter function is defined as:

$$\text{filter}_m(\text{agent, human}) = \begin{cases} 1.0 & \text{if } m(\text{human}) = 0 \\ m(\text{agent}) & \text{otherwise} \end{cases}$$
(6)

3.2. Implementation Details

Unless otherwise specified, we train our models on the navtrain split using 8 NVIDIA A100 GPUs with a total batch size of 256 for 30 epochs. The learning rate is set to 1×10^{-4}, and the weight decay is 0.0. The input to the model consists of four RGB camera views: front, front-left, front-right, and rear. The BEV feature covers a 64m×64m region around the ego vehicle, discretized into a 100×100 grid. To improve model robustness, we apply GridMask data augmentation with a probability of 0.5 during training. Following the configuration of DiffusionDrive [20], we initialize 20 trajectory queries using its predefined anchor set.

3.3. Results

We present the performance of our method and various ablations on the *"warmup two stage"* using the Extended Predictive Driver Model Score (EPDMS) and its sub-metrics. As shown in Tab. 1, our full model achieves the best overall performance with an EPDMS of 65.94. This model is built upon the configuration with 2 decoder layers, further enhanced by a post-processing module, which leads to consistent gains across safety (NC 98.65, TTC 98.52) and compliance (DAC 91.36, DDC 99.34).

Removing key components significantly impacts performance. The scorer removal results in an EPDMS drop to 50.31, show that the learning of trajectory scoring can effectively promote the model's understanding of the scene. Similarly, The removal of hard case mining has led to a decrease in the model's safety metrics in complex scenarios, such as NC, TTC, DAC. We further study the impact of decoder depth. Increasing from 1 to 2 layers improves most metrics, especially EP (from 78.19 to 80.10) and LK (from 76.37 to 80.14), demonstrating better learning capacity. However, increasing to 4 decoder layers leads to a slight drop in EPDMS to 59.25, likely due to overfitting.

References

[1] 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. 4

[2] 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

[3] Felipe Codevilla, Eder Santana, Antonio M. Lopez, and Adrien Gaidon. Exploring the limitations of behavior cloning for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 1

[4] OpenScene Contributors. Openscene: The largest up-to-date 3d occupancy prediction benchmark in autonomous driving. https://github.com/OpenDriveLab/OpenScene, 2023. 3, 4

[5] 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. 4

[6] Kairui Ding, Boyuan Chen, Yuchen Su, Huan-ang Gao, Bu Jin, Chonghao Sima, Wuqiang Zhang, Xiaohui Li, Paul Barsch, Hongyang Li, and Hao Zhao. Hint-ad: Holistically aligned interpretability in end-to-end autonomous driving. *Conference on Robot Learning (CoRL)*, 2024. 1

[7] Yuchao Feng, Wei Hua, and Yuxiang Sun. Nle-dm: Natural-language explanations for decision making of autonomous driving based on semantic scene understanding. *IEEE Transactions on Intelligent Transportation Systems*, 2023. 1

[8] Huan-ang Gao, Beiwen Tian, Pengfei Li, Xiaoxue Chen, Hao Zhao, Guyue Zhou, Yurong Chen, and Hongbin Zha. From semi-supervised to omni-supervised room layout estimation using point clouds. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2803–2810. IEEE, 2023.

[9] Huan-ang Gao, Beiwen Tian, Pengfei Li, Hao Zhao, and Guyue Zhou. Dqs3d: Densely-matched quantization-aware semi-supervised 3d detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21905–21915, 2023. 1

[10] Cheng Han, Qichao Zhao, Shuyi Zhang, Yinzi Chen, Zhenlin Zhang, and Jinwei Yuan. Yolopv2: Better, faster, stronger for panoptic driving perception. *arXiv preprint arXiv:2208.11434*, 2022. 3

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[12] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, Lewei Lu, Xiaosong Jia, Qiang Liu, Jifeng Dai, Yu Qiao, and Hongyang Li. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17853–17862, 2023. 1

[13] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, Lewei Lu, Xiaosong Jia, Qiang Liu, Jifeng Dai, Yu Qiao, and Hongyang Li. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 1

[14] 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. *ICCV*, 2023. 1

[15] Zhou Jiang, Zhenxin Zhu, Pengfei Li, Huan-ang Gao, Tianyuan Yuan, Yongliang Shi, Hang Zhao, and Hao Zhao. P-mapnet: Far-seeing map generator enhanced by both sdmap and hdmap priors. *arXiv preprint arXiv:2403.10521*, 2024. 1

[16] Alex Kendall, Jeffrey Hawke, David Janz, Przemyslaw Mazur, Daniele Reda, John-Mark Allen, Vinh-Dieu Lam, Alex Bewley, and Amar Shah. Learning to drive in a day. In *2019 international conference on robotics and automation (ICRA)*, 2019.

[17] Wenyi Li, Huan-ang Gao, Mingju Gao, Beiwen Tian, Rong Zhi, and Hao Zhao. Training-free model merging for multi-target domain adaptation. *arXiv preprint arXiv:2407.13771*, 2024. 1

[18] Zhenxin Li, Kailin Li, Shihao Wang, Shiyi Lan, Zhiding Yu, Yishen Ji, Zhiqi Li, Ziyue Zhu, Jan Kautz, Zuxuan Wu, et al. Hydra-mdp: End-to-end multimodal planning with multi-target hydra-distillation. *arXiv preprint arXiv:2406.06978*, 2024. 1, 3

[19] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 1, 2

[20] Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. *arXiv preprint arXiv:2411.15139*, 2024. 1, 2, 4

[21] Aditya Prakash, Kashyap Chitta, and Andreas Geiger. Multi-modal fusion transformer for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7077–7087, 2021. 1

[22] Chonghao Sima, Kashyap Chitta, Zhiding Yu, Shiyi Lan, Ping Luo, Andreas Geiger, Hongyang Li, and Jose M Alvarez. Centaur: Robust end-to-end autonomous driving with test-time training. *arXiv preprint arXiv:2503.11650*, 2025. 1

[23] Beiwen Tian, Mingdao Liu, Huan-ang Gao, Pengfei Li, Hao Zhao, and Guyue Zhou. Unsupervised road anomaly detection with language anchors. In *2023 IEEE international conference on robotics and automation (ICRA)*, pages 7778–7785. IEEE, 2023.

[24] Wenda Xu, Qian Wang, and John M Dolan. Autonomous vehicle motion planning via recurrent spline optimization. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021. 1

[25] Zhejun Zhang, Alexander Liniger, Dengxin Dai, Fisher Yu, and Luc Van Gool. End-to-end urban driving by imitating a reinforcement learning coach. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15222–15232, 2021. 1

[26] Yupeng Zheng, Chengliang Zhong, Pengfei Li, Huan-ang Gao, Yuhang Zheng, Bu Jin, Ling Wang, Hao Zhao, Guyue Zhou, Qichao Zhang, et al. Steps: Joint self-supervised nighttime image enhancement and depth estimation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4916–4923. IEEE, 2023. 1