Feedback-Guided Autonomous Driving

Jimuyang Zhang Zanming Huang Arijit Ray Eshed Ohn-Bar
Boston University
{zhangjim, huangtom, array, eohnbar}@bu.edu

Abstract

While behavior cloning has recently emerged as a highly successful paradigm for autonomous driving, humans rarely learn to perform complex tasks, such as driving, via imitation or behavior cloning alone. In contrast, learning in humans often involves additional detailed guidance throughout the interactive learning process, i.e., where feedback, often via language, provides detailed information as to which part of their trial was performed incorrectly or suboptimally and why. Motivated by this observation, we introduce an efficient feedback-based framework for improving behavior-cloning-based training of sensorimotor driving agents. Our key insight is to leverage recent advances in Large Language Models (LLMs) to provide corrective fine-grained feedback regarding the underlying reason behind driving prediction failures. Moreover, our introduced network architecture is efficient, enabling the first sensorimotor end-to-end training and evaluation of LLM-based driving models. The resulting agent achieves state-of-the-art performance in open-loop evaluation on nuScenes, outperforming prior state-of-the-art by over 8.1% and 57.1% in accuracy and collision rate, respectively. In CARLA, our camera-based agent improves by 16.6% in driving score over prior LIDAR-based approaches.

1. Introduction

Humans often learn new tasks through the synchrony of demonstration and feedback [46, 60]. Consider the task of driving, especially the challenging feat of merging onto a highway. Simply re-watching videos of a driver merging is often insufficient to learn complex longitudinal and latitudinal control tasks; it is immensely more helpful to receive feedback on when to accelerate, match the speed of other cars, and finally merge when safe.

Currently, the most common paradigm for training autonomous driving systems relies primarily on learning from expert demonstrations, e.g., via behavior cloning [2, 3, 15, 28, 30, 35–37, 37, 39, 95, 99, 108]. While this has fueled impressive improvements, these systems still fail to generalize to a wide range of novel scenarios [19, 37, 80, 86, 97]. We believe a possible reason could be the lack of feedback explaining why a certain action policy fails. Rich feedback from a teacher can be beneficial to learning the cause of a control failure in an out-of-domain scenario [6, 26, 59]. While humans can use the rich space of language to receive feedback, existing model pipelines lack such an intuitive interface. Some works often resort to complicated feedback pipelines like ranking reward functions [71]. Recently, multimodal large language models [50, 53, 61, 79, 84, 107] with their ability to converse, perceive, and act [22], seem to have the fundamental qualities that would allow rich language feedback to supervise and mitigate robustness issues in sensorimotor driving agents.

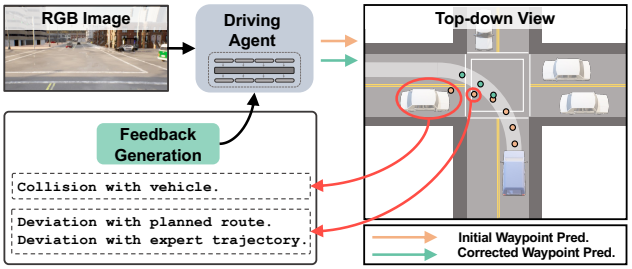


Figure 1. **Robust Sensorimotor Agent Training through Feedback Guidance.** Our study introduces a feedback-guided mechanism for training a sensorimotor driving policy. Specifically, our proposed approach leverages a Large Language Model to guide the driving policy learning task through structured critique and reasoning as language prompt, i.e., to effectively reflect and learn from prediction mistakes.

Hence, we propose FeD, a feedback-guided end-to-end sensorimotor driving agent that employs a multimodal large language model (MLLM) to leverage their rich language interface for user-control and refinement, as illustrated in Fig. 1. Our approach overcomes bottlenecks on recently-proposed contemporary MLLM-based driving methods [58, 90] using three key features: 1) Current models lack a language interface to refine predictions from explanations of failure. In contrast, we train an MLLM with a rich language understanding prior to refining its prediction from auto-generated explanations of failure - allowing a user to teach

and refine predictions using language at test time as well. 2) A recent GPT4-based driving method [58] employs complicated pipelines to translate images into intricate spatial state information at test time. However, we train our MLLM via distillation from a privileged agent with access to ground-truth Bird's Eye View (BEV) of the scene [89, 101]. This allows the MLLM to generalize more robustly to just raw images at test time. 3) MLLMs are often slow to iteratively generate a sequence of tokens, limiting their real-time deployment and closed-loop evaluation for a speed-critical application like driving [90]. To improve speed, we infer the waypoints from the token representations in a masked-token fashion [38, 57, 81]. This enables us to achieve a significantly higher frame rate, which is crucial for real-time deployment. Furthermore, all contemporary MLLM-based driving approaches [58, 90] use closed-source and expensive GPT API's that cannot be evaluated in real-time or be trained end-to-end. In contrast, we focus on adapting an open-source model, LLaVA [53], so that the community can easily extend and extensively analyze the various modules of our model. As a result, our novel agent enables efficient end-to-end closed-loop evaluation, i.e., on CARLA [21], for the first time to the best of our knowledge.

In summary, our contributions are as follows. We introduce FeD, a highly efficient MLLM-based sensorimotor driving model, enabled by three key improvements: 1) language-based feedback refining trained using auto-generated feedback data. Hence, our approach requires no additional data collection. 2) training the model via distillation from a privileged agent with Bird's Eye View (BEV) of the scene, allowing our model to robustly use just RGB data at test time. 3) predicting driving waypoints in a masked-token fashion from the waypoint tokens' internal representations, i.e., not relying on the slow sequentially generative process. In our experiments, we demonstrate state-of-the-art performance in open-loop and closed-loop evaluation settings, improving over prior methods by over 16% in performance, particularly benefiting from the additional auto-generated language-based feedback. Notably, FeD achieves a significant drop in infractions by over 33% with almost zero collisions with objects in CARLA.

2. Related Work

Imitation Learning for Autonomous Driving (AD): Recent advances in imitation learning (IL) for autonomous driving originate from ALVINN [65], a neural network trained to imitate the driving behavior of an ego vehicle. Since then, more elaborate IL-based approaches for autonomous driving have emerged [8, 9, 11, 27, 44, 49, 62, 70, 77, 78, 89, 96, 98, 99, 103, 104, 108]. However, today's behavioral cloning systems have yet to be successfully transferred to large-scale deployment in the real-world [2, 18, 76] and are plagued by an array of learning-based issues, such as shortcut learning [37] and overfitting to spurious correlations [19, 86]. In our work, we introduce richer feedback, such as language, to effectively supervise and mitigate robustness and efficiency issues in sensorimotor driving agents. Moreover, previous methods show that decomposing the imitation task into two stages - training a privileged agent and then learning to distill from the privileged agent [7, 9, 10, 83, 89, 99, 101] can significantly ease the difficult sensorimotor learning task. However, none of the approaches leverage the rich language priors of MLLMs and employ effective knowledge acquisition. Moreover, we demonstrate effective feedback to be more crucial for teaching complex planning skills to sensorimotor driving agents.

Vision-Language for Autonomous Driving: Due to the rich linguistic prior in LLMs, they are widely employed in autonomous driving tasks, e.g., for anomaly detection [45], and Bird's Eye View prediction [20]. Other works focus on augmenting driving datasets with various language descriptions [66, 88], or constructing environments for novel driving scenarios [24, 31, 85, 100, 102]. Recently, a growing number of contemporary works leverage MLLMs for driving decisions [12, 16, 17, 23, 58, 73, 87, 90]. Specifically, Mao et al. [58] proposes a motion planner and Xu et al. [90] presents an interpretable end-to-end driving system, both using the closed-source GPT4. We use an open-source LLaVA [53] to spur further open development. More importantly, Mao et al. [58] estimates intricate state information, which we bypass using a teacher-student distillation approach. While Xu et al. [90] operates on images, their approach is slow due to an autoregressive action generation approach. In contrast, we predict waypoints all at once, and leave action control to a separate module, allowing us to improve frame rate and perform closed-loop evaluation. Other related methods [17, 23, 87] that conduct closed-loop evaluation use a privileged simulator [47], constraining their applicability in real-world sensorimotor scenarios.

Large Multimodal Language Models for Control: Recent advances in LLMs [22, 50, 53, 61, 63, 79, 84, 107] provide an intuitive approach for incorporating general feedback and task specifications with scene topology and planning [4, 75, 82]. MLLMs, that accept images and language [50, 53, 61, 107], have shown impressive downstream task performance for robotics and control - for predicting control policies [51, 52], designing rewards [93] and design and motion planning [41] for embodied AI [22, 74, 92]. Instruction tuning, in-context learning [48, 106], and custom tokens have powered a lot of these directions to tune the rich prior of these models for either a specific task, e.g., robotic action control [4], or generalist action agents [69]. However, efficiently leveraging such powerful models in the context of autonomous driving policies remains minimally explored, with concurrent solutions leveraging inefficient

pipelines with multiple stages which cannot be optimized end-to-end nor accommodate standard closed-loop evaluation [12, 54, 58, 87, 90], e.g., on CARLA [1, 21].

Explainability and Refinement: Explaining deep network predictions, both in an introspective [64, 72] manner and by rationalization-based approaches [25, 29] for human-AI collaboration has also been long studied for increasing human trust in robotics [42, 91] and improving users' mental models of vision-language agents [68]. However, there is a very limited exploration into whether explanations of failure cases can improve model performance, especially in the realm of robotic control and driving. While recent works have focused on explanations for robotic failures [54] and object navigation [79], they also do not show if we can use these explanations to make the network refine the predictions. Here, we show that explanations help FeD refine its predictions by training it to take the past failure in the context while predicting the next waypoints. We hypothesize that grounding the model in language can provide more structured failure reasoning, i.e., in contrast to the coarse supervision provided by simply incorporating additional auxiliary losses alongside the imitation objective [2, 14, 99].

3. Method

Towards facilitating robust and efficient training of sensorimotor driving agents, we leverage pre-trained MLLMs that can ease model training through effective feature distillation and feedback reasoning fine-tuning. In this section, we first formulate the sensorimotor learning problem in Sec. 3.1. Next, we propose a framework for adapting MLLMs to the end-to-end driving policy learning task via distillation and efficient inference in Sec. 3.2. Given the proposed network structure, we propose a fine-tuning process that facilitates effective failure reasoning in Sec. 3.3. An overview of our model and training process can be seen in Fig. 2.

3.1. Problem Setting

Our formulation follows the standard goal-driven navigation task based on CARLA [10, 21]. MLLMs are not traditionally trained to process dense spatial information required for driving. Thus, to ease the training of the sensorimotor agent, we leverage rich supervision over the internal features via knowledge distillation from a privileged agent $f^p_{\boldsymbol{\psi}}: \mathcal{X}^p \rightarrow \mathcal{Y}$, which has access to privileged observations $\mathbf{x}^p = (\mathbf{I}, v, c, \mathbf{g}, \mathbf{E}) \in \mathcal{X}^p$, where the environmental information $\mathbf{E} = \{\mathbf{L}, \mathbf{O}, \mathbf{T}\}$ contains the ground-truth planned route centerline $\mathbf{L} \in \mathbb{R}^{N_l \times 2}$, object locations $\mathbf{O} \in \mathbb{R}^{N_o \times 2}$, and traffic light locations with their states $\mathbf{T} \in \mathbb{R}^{N_t \times 3}$. N_l, N_o, N_t are the number of centerline points, objects, and traffic lights, respectively.

Given the privileged driving agent, our goal is to train a student sensorimotor policy function $f^s_{\boldsymbol{\theta}}: \mathcal{X}^s \rightarrow \mathcal{Y}$ that maps a set of sensory observations $\mathcal{X}^s$ to a set of navigational decision $\mathcal{Y}$. In our setting, we assume the access to the sensory observations $\mathbf{x}^s = (\mathbf{I}, v, c, \mathbf{g}) \in \mathcal{X}^s$ of a front camera image $\mathbf{I} \in \mathbb{R}^{W \times H \times 3}$, ego vehicle speed $v \in \mathbb{R}$, a categorical navigational command $c \in \mathbb{N}$ (i.e., turn left, turn right, follow, forward, left lane changing and right lane changing), and an intermediate GNSS (Global Navigation Satellite System) coordinate goal $\mathbf{g} \in \mathbb{R}^2$ [9, 99]. Given the observations at each time step, sensorimotor agent $f^s_{\boldsymbol{\theta}}$ learns to predict a set of K waypoints $\mathbf{y}^s \in \mathcal{Y}$ in the future. Nonetheless, most of this information is not easily conveyable to an LLM without a proper architecture and training process, discussed next.

3.2. An End-to-End LLM-based Driver

In this section, we introduce our framework for training sensorimotor driving skills to LLMs. We closely follow a MLLM architecture LLaVA [53], with an additional waypoints prediction head consisting of multi-layer perceptron (MLP), as shown in Fig. 2. This enables FeD to be initialized from pre-trained LLaVA-7B weights [53], benefiting from the large diverse image-text corpus used to train MLLMs. However, we find off-the-shelf LLaVA to perform poorly in intricate spatial reasoning tasks, addressed below.

Token Prediction Mechanism: We note that our proposed architecture does not leverage generative sequence prediction as in most related approaches, e.g., [58, 90], but instead draws inspiration from more efficient methodologies based on masked token prediction [57]. Formally, current LLM-based autonomous driving models are trained to generate a sequence of text tokens $s_1 \dots s_n \in \mathcal{S}$ by modeling the probability of the next token given all seen tokens $P(s_n | s_1, \dots, s_{n-1})$. At the inference time, given the tokenized input prompts, the model samples from the distribution recursively until the end token is reached: $\tilde{s}_n \sim P(s | s_1, \dots, s_{n-1})$. Such language generation process results in long inference time and instability, which is critical for real-time on-road applications and closed-loop evaluation. Moreover, several recent related methods,e.g., [17, 23, 87], conduct closed-loop evaluations in a simplistic highway simulations [47] with privileged state information such that their applicability in more complex and real-world scenarios remains unknown. Therefore, we present FeD, the first efficient end-to-end LLM-based sensorimotor driving agent.

Vision Encoder: The front camera image $\mathbf{I}$ is processed by a CLIP ViT vision encoder [67], whose output image features $\mathbf{Z}$ are converted into language embeddings $\mathbf{U} \in \mathbb{R}^{512 \times 4096}$ by a trainable projection matrix $\mathbf{A}$

$$\begin{aligned} \mathbf{Z} &= \text{VisionEncoder}(\mathbf{I}) \\ \mathbf{U} &= \mathbf{A} \cdot \mathbf{Z} \end{aligned} \tag{1}$$

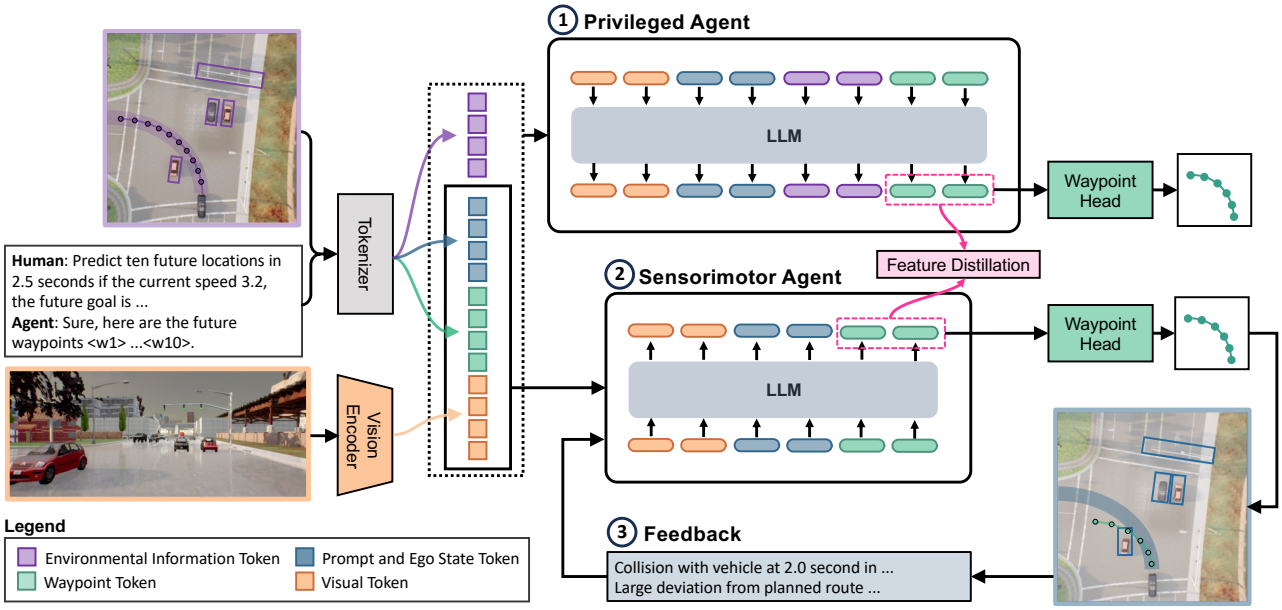


Figure 2. **Network Architecture and Training Process of FeD.** Our goal is to train a sensorimotor agent to map front camera images (orange) and ego vehicle state information (blue) encoded as language tokens, and predict a set of future waypoints. This is accomplished by introducing new waypoint tokens (green) as part of the input prompt. Our introduced tokens also enable us to leverage the rich output embeddings from the LLM for the prompt to perform direction waypoint prediction, i.e., as opposed to slow and inefficient sequential generation. Our training is done in two stages. First, to ease the challenging sensorimotor learning task, we introduce a privileged agent that additionally takes ground truth environmental information (purple) and provides rich supervision for training the sensorimotor agent through feature distillation. Subsequently, the sensorimotor agent is fine-tuned with prompt-based feedback to enable efficient failure reasoning, i.e., effective reflection on its own mistakes.

Language Encoder: Given the language prompt shown in Fig. 3, we first compute language embeddings $\mathbf{Q}$ which is concatenated with visual embeddings $\mathbf{U}$. We then encode the concatenated embeddings $[\mathbf{U}, \mathbf{Q}]$ with an LLM,

$$\mathbf{P} = \text{LLM}([\mathbf{U}, \mathbf{Q}]) \tag{2}$$

where $\mathbf{P}$ is a softmax probability over the entire vocabulary space at each step. For our LLM embeddings we leverage pre-training with LLaMA [84].

Waypoint Prediction Head: Our efficient MLP-based waypoints prediction head takes as input the features of the last hidden layer $\mathbf{H}^s \in \mathbb{R}^{K\times 4096}$ from the MLLM that corresponds to the K waypoint tokens in the language prompt (discussed later) and outputs waypoints $\mathbf{y}^s \in \mathbb{R}^{K\times 2}$

$$\mathbf{y}^s = \text{WaypointHead}(\mathbf{H}^s) \tag{3}$$

To emphasize, we propose to directly compute the waypoints from the output embeddings of those tokens. This bypasses the need for recursive token-by-token generation and expensive sampling strategies like beam search, leading to a more efficient inference process. Specifically, our proposed architecture is able to achieve a frame rate of 2.7 frames per second (FPS), which is more than 20× faster comparing to generating waypoints as language tokens recursively that runs at around 0.1 FPS.

Prompt Design for Sensorimotor Agent: As shown in Fig. 3, for the sensorimotor agent, we wrap ego-vehicle speed v and short-term goal $\mathbf{g}$ with flag tokens indicating the beginning and the end of the text span. We further provide the categorical command as natural language, i.e., turn left, turn right, go straight, follow the lane, change lane to the left, change lane to the right. We additionally introduce K waypoint tokens, i.e., "<w1>...<wk>", whose corresponding features out of the last hidden layer from LLM are used for final waypoints prediction. We introduce 512 image patch tokens "<im_patch>" as placeholders, whose embedding features will later be replaced by visual embeddings $\mathbf{U}$ before inputting it to the LLM.

Prompt Design for Privileged Agent: For the privileged agent, we additionally provide parameterized environmental information. Specifically, all the surrounding objects, i.e., vehicles, and pedestrians within 30 meters range in front of the ego vehicle can be represented by its location in BEV. We discretize the BEV into a 96×96 grid and each cell of the grid can be represented by a location token $\texttt{<loc\{i\}>}|_{\text{i=1}}^{96\times 96}$. Therefore, each continuous location in BEV can be represented by a location token of the cell it falls in. The traffic light is represented by a location token and a state token, i.e., "<red>,<yellow>,<green>". A delimiter token is applied to separate each object. The

Privileged Agent Prompt

Human: Predict ten future locations in 2.5 seconds if the current speed <speed_start>3.2</speed_end>, the future goal is <goal_start>(18.79, -37.26)</goal_end> and the command is to follow the lane, given current front camera view:
<im_start><im_patch><im_patch> … <im_patch><im_end>
and the information about surrounding objects with their predicted movements, traffic lights with their states, and the planned route:
Vehicles: <veh_start><loc7399><delimiter><loc126><delimiter>…<delimiter><loc293></veh_end>
Pedestrians: <wlk_start><loc8462><delimiter><loc667></wlk_end>
Traffic lights: <tl_start><loc7524><delimiter><red></tl_end>
Planned route: <rl_start><loc6122><loc409>…<loc836></rl_end>

Agent: Sure, here are the future waypoints <waypoints_start><w1><w2><w3><w4><w5><w6><w7><w8><w9><w10></waypoints_end>

Sensorimotor Agent Prompt

Human: Predict ten future locations in 2.5 seconds if the current speed <speed_start>3.2</speed_end>, the future goal is <goal_start>(18.79, -37.26)</goal_end> and the command is to follow the lane, given current front camera view:
<im_start><im_patch><im_patch> … <im_patch><im_end>

Agent: Sure, here are the future waypoints <waypoints_start><w1><w2><w3><w4><w5><w6><w7><w8><w9><w10></waypoints_end>

Sensorimotor Agent Prompt with Feedback

Human: given current front camera view:
<im_start><im_patch><im_patch> … <im_patch><im_end>
If ego vehicle's current speed is <speed_start>3.2</speed_end>, the future goal is <goal_start>(18.79, -37.26)</goal_end> and the command is to follow the lane, please evaluate the predicted future locations of ego vehicle
<waypoint_proposal_start><loc9168><delimiter><loc8592>…<delimiter><loc5280></waypoint_proposal_end>

Agent: Sure, given the predicted future locations of ego vehicle, the following instances would occur:
<feedback_start>Collision with vehicle at 1.5 seconds in the future, with vehicle at (-0.04, -7.77).
Collision with pedestrian at 1.5 seconds in the future, with pedestrian at (0.21, -3.34).
Traffic light violation.
Large deviation with planned route at 2.5 seconds in the future, with an error of 3.99 meters.
Large deviation with expert waypoints at 2.5 seconds in the future, with an error of 1.58 meters.</feedback_end>
And the corrected future locations should be:
<waypoints_start><w1><w2><w3><w4><w5><w6><w7><w8><w9><w10></waypoints_end>

Figure 3. **Examples of Input Prompts of Sensorimotor Agent, Privileged Agent and Feedback Reasoning.**

planned route is represented by a set of location tokens of points sampled uniformly along the centerline of the lane. With ground-truth environmental information, the privileged agent features can provide rich supervision through distillation, discussed next.

Sensorimotor Agent Loss: For initial training of the sensorimotor agent, we leverage a loss comprising two terms. First, we incorporate a waypoints prediction task, computed by leveraging an $\mathcal{L}_1$ loss computed between the prediction and the expert demonstration

$$\mathcal{L}_{wpts} = \|f^s_{\boldsymbol{\theta}}(\mathbf{x}^s) - \mathbf{y}\|_1 \tag{4}$$

Secondly, following distillation-based approaches [89, 101] we apply a feature distillation $\mathcal{L}_2$ loss between the features $\mathbf{H}^s$ and $\mathbf{H}^p$ in the sensorimotor and privileged agent

$$\mathcal{L}_{feat} = \|\mathbf{H}^s - \mathbf{H}^p\|_2 \tag{5}$$

We leverage distillation and do not add additional BEV-based auxiliary tasks (e.g., [14]), to avoid drastically increasing the language space size and resulting in prohibitive computational overhead.

Therefore, our optimization objective for training the sensorimotor agent is a weighted sum over the feature distillation and waypoints prediction loss

$$\mathcal{L} = \mathcal{L}_{wpts} + \mathcal{L}_{feat} \tag{6}$$

3.3. Feedback-Guided Fine-tuning

We propose to incorporate feedback fine-tuning by leveraging fine-grained textual feedback regarding waypoint prediction errors. This enables the sensorimotor agent to effectively learn from experience, including failure which can provide a highly informative supervision signal. In FeD, we guide the waypoint predictions with structured critique and reasoning as language prompts. Given the ground-truth surrounding object states and the original waypoint predictions, we define a rich taxonomy over five failure cases and generate a corresponding feedback prompt for each failure case. Examples of this process are shown in Fig. 3.

Collision with Vehicles: To enhance the model's ability to reason over dynamic scenes and future states of surrounding objects, our feedback emphasizes both current and potential future collisions. To achieve this, we extrapolate the future states of surrounding vehicles using a kinematic bicycle model $\mathcal{T}_{veh}$ [11, 14], which predicts the next location (x_{t+1}, y_{t+1}), orientation α_{t+1}, and speed v_{t+1} of the vehicle, given its current location (x_t, y_t), orientation α_t, speed v_t, and the applied control a_t

$$x_{t+1}, y_{t+1}, \alpha_{t+1}, v_{t+1} = \mathcal{T}_{veh}(x_t, y_t, \alpha_t, v_t, a_t) \tag{7}$$

We then compute the potential collision Col^j_t with the j-th

vehicle at time step t,

$$\text{Col}_t^j = \mathbb{1}(\text{IoU}(B_t^s, O_t^j) > \varepsilon_C) \tag{8}$$

where B_t^s is the ego vehicle bounding box in BEV at the predicted waypoint location $\mathbf{y}_t^s$, O_t^j is the bounding box of the j-th vehicle given its bicycle model output $(x_t^j, y_t^j, \alpha_t^j)$ at time t, and ε_C is the collision threshold. We use the intersection over union (IoU) between two bounding boxes to determine if collisions occur.

Collision with Pedestrians: To check for collision with pedestrians, we predict future pedestrian locations using a kinematic model $\mathcal{T}_{ped}$ which anticipates the next location (x_{t+1}, y_{t+1}) given its current location (x_t, y_t) and speed v_t assuming a constant speed

$$x_{t+1}, y_{t+1} = \mathcal{T}_{ped}(x_t, y_t, v_t) \tag{9}$$

We identify potential collisions with pedestrians by calculating IoU between ego vehicle and pedestrian bounding boxes similar to Eq. 8. This collision model, though simplified, prioritizes safety by preventing policy overconfidence in less likely scenarios, e.g., assuming a pedestrian will jaywalk unless slowing down.

Traffic Light Violations: When the ego vehicle enters the range of the traffic light bounding box, we check the traffic light violation by calculating the movement of the ego vehicle when the traffic light state is red or yellow

$$\text{TL\_Violation}_t = \mathbb{1}(\sum_{t=0}^{K-1} \|\mathbf{y}_{t+1}^s - \mathbf{y}_t^s\|_2 > \varepsilon_{TL}) \tag{10}$$

A traffic light violation occurs if the accumulated predicted waypoints movement is larger than threshold ε_{TL}.

Deviation from Expert Demonstration: We calculate the deviation of the predicted waypoints $\mathbf{y}_t^s$ from the expert demonstration y_t at each time step t

$$\text{Dev\_Expert}_t = \mathbb{1}(\|\mathbf{y}_t^s - \mathbf{y}_t\|_2 > \varepsilon_E) \tag{11}$$

We identify deviations by checking if the distance between the predicted waypoint and the expert demonstration is larger than the threshold ε_E.

Deviation from Planned Route: We further compute the deviation of the predicted waypoints $\mathbf{y}_t^s$ from N_c points along the centerline of the planned route,

$$\text{Dev\_Plan}_t = \mathbb{1}(\min_{i \in N_c}(\|\mathbf{y}_t^s - \mathbf{y}_i^p\|_2) > \varepsilon_P) \tag{12}$$

where $\mathbf{y}_i^p$ is the i-th point along the center line. We examine deviations by verifying whether the distance between the predicted waypoint and its nearest point along the centerline of the planned route is larger than the threshold ε_P,

Model Fine-tuning with Feedback: To ensure our agent's ability to both generate informative failure feedback and rectify mispredicted waypoints, we supervise the model learning by applying a Cross-Entropy (CE) loss of the LLaMA outputs over the generated language feedback,

$$\mathcal{L}_{CE} = \exp\Big[-\frac{1}{t}\sum_{n}^{t} \log P(s_n | s_1, \ldots, s_{n-1})\Big] \tag{13}$$

where t is the feedback length. We additionally compute a $\mathcal{L}_1$ loss over the corrected waypoints similar to Eq. 4. The optimization objective for our proposed feedback fine-tuning procedure is hence a weighted sum over waypoint predictions and language CE loss

$$\mathcal{L}_{feedback} = \mathcal{L}_{wpts} + \mathcal{L}_{CE} \tag{14}$$

We show that our proposed approach further improves the driving performance in all benchmarks in Sec. 4.

3.4. Training Procedure

We initialize our vision encoder and language encoder network with the pre-trained LLaVA-7B model, which are tuned via a LORA [32] adapter for efficiency. Training the sensorimotor agent involves two stages. First, the agent is trained with feature distillation based on Eq. 6. In the second stage, we fine-tune the model with the proposed feedback reasoning and the objective in Eq. 14. The two-stage training procedure is designed for the sake of speed and memory efficiency, which is a critical current bottleneck with MLLMS. Given that the language prompt with feedback reasoning can be lengthy, leading to a significant increase in memory usage and training time, we opt to train the model initially without feedback and a larger batch size (of six samples per GPU). This enables the model to rapidly learn the waypoint prediction task. We train for 10 epochs on four NVIDIA A6000 GPUs. We then fine-tune with the introduced feedback reasoning for 10 epochs using a smaller batch size of one. We use AdamW [56] and cosine annealing [55] scheduler with weight decay 1×10^{-6} and an initial learning rate of 5×10^{-4}.

4. Experiments

In this section, we demonstrate the effectiveness of the proposed FeD framework through extensive closed-loop evaluation in CARLA simulator [21] and open-loop evaluation on nuScenes dataset [5]. In CARLA evaluation, to measure the generalization of the methods to new towns, we use the LAV [9] benchmark. It contains four routes in Town 02 and Town 05 with four weathers (16 in total), which are withheld in training. We follow the standard CARLA leaderboard [1] metrics and report Driving Score (DS), Route Completion (RC), and Infraction Score (IS). DS is our main

Table 1. **Quantitative Evaluation in CARLA.** Driving Score (DS - main metric), Route Completion (RC), and Infraction Score (IS) are reported. We additionally show detailed infraction metrics including Pedestrian Collisions (Ped), Vehicle Collisions (Veh), Layout Collisions (LC), Red Light Violations (Red), Route deviation (Dev), Route Timeouts (TO), and Agent Blocked (Block). Results are taken from the corresponding publication when available. * denotes our evaluation using the publicly available code.

Method	DS ↑	RC ↑	IS ↑	Ped ↓	Veh ↓	LC ↓	Red ↓	Dev ↓	TO ↓	Block ↓
NEAT* [13]	14.66	**100.0**	0.15	0.00	0.03	0.00	0.87	0.00	0.00	0.00
LAV [9]	45.20	91.55	0.49	0.00	0.92	0.33	0.28	-	-	-
TransFuser [14]	39.00	84.00	0.46	0.00	0.74	1.04	0.20	0.00	0.23	0.21
InterFuser* [11]	44.27	59.22	0.67	0.00	0.57	0.06	0.07	0.00	0.47	0.18
TCP [89]	58.00	85.00	0.67	0.00	0.35	0.16	0.01	0.00	0.19	0.19
CaT [99]	66.70	92.14	0.73	-	-	-	-	-	-	-
TransFuser++ [37]	70.00	99.00	0.70	0.01	0.63	0.01	0.04	0.00	0.05	0.00
FeD (Privileged)	80.51	89.20	**0.92**	0.00	0.01	0.00	0.01	0.00	0.01	0.02
FeD (w/o Feedback, Distillation)	69.14	86.64	0.83	0.00	0.01	0.00	0.01	0.00	0.03	0.00
FeD (w/o Feedback)	72.45	86.78	0.78	0.00	0.02	0.00	0.03	0.00	0.00	0.02
FeD	**81.60**	97.60	0.84	0.00	0.01	0.00	0.00	0.00	0.01	0.00
TinyFeD	74.65	84.98	0.90	0.00	0.01	0.00	0.00	0.00	0.00	0.03
Expert	88.32	95.82	0.93	0.00	0.03	0.00	0.00	0.00	0.00	0.02

metric which is computed from RC and IS. To provide more insight into our evaluation, more detailed metrics are provided in Table 1. We further analyze FeD in real-world driving settings using the challenging nuScenes [5] benchmark. Following previous works [35, 40, 58, 99], we show the L_2 displacement error and the collision rate in one, two, and three seconds to evaluate the model performance.

4.1. Comparison With State-Of-The-Art

Closed-Loop Results: As shown in Table 1, with feature distillation training from the privileged agent and the proposed feedback reasoning fine-tuning, FeD obtains a DS of 81.60%, achieving state-of-the-art performance. We note that FeD, which *only takes as input a single front camera image*, outperforms the best LiDAR-based method, TransFuser++ by 16.6% in DS, from 70.00 to 81.60, and 20.0% in IS while achieving comparable performance to the rule-based expert. Given various memory and inference constraints of LLMs, we also include results with a lighter-weight model based on TinyLLaVA [105]. The resulting TinyFeD is one-seventh its original model size and achieves a 74.65 DS, 84.98 RC, and 0.90 IS scores. Specifically, TinyFeD outperforms Transfuser++, while the two have similar inference speeds, 10.3 FPS vs. 11.5 FPS for TinyFeD and Transfuser++, respectively.

Open-Loop Results: To further demonstrate the benefits of FeD in real-world driving scenarios, Table 2 shows the proposed method obtains state-of-the-art performance on the official nuScenes validation split in both displacement error and collision rate. Specifically, FeD achieves an average L_2 displacement error of 0.34m, improving by 8.1% over the previous best none-LLM-based method VAD [40]. Moreover, our proposed method surpasses the LLM-based GPT-Driver [58], reducing the average L_2 displacement error from 0.44m to 0.34m, and the average collision rate from 0.17% to 0.06% under ST-P3 metrics. We note that GPT-Driver depends on a separate vision perception model which is not trained in an end-to-end fashion. Moreover, FeD is efficient, while GPT-Driver involves complicated chain-of-thought reasoning and slow recursive language generation making real-time closed-loop evaluation difficult. Under UniAD metrics [35], FeD consistently outperforms prior works by a significant margin. Additional analysis can be found in the supplementary.

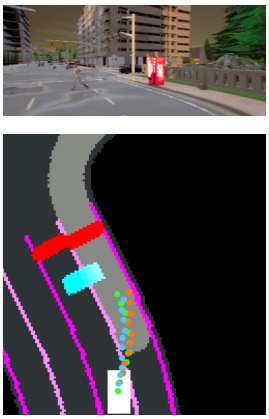
Table 2. **Open-Loop Evaluation on nuScenes.** FeD achieves state-of-the-art open-loop evaluation performance on nuScenes [5] validation set compared with both none-LLM based methods and LLM-based GPT-Driver [58]. We evaluate FeD on two different measures of metrics for fair comparison[1].

Metrics	Method	L2 (m) ↓				Collision (%) ↓			
		1s	2s	3s	Avg.	1s	2s	3s	Avg.
ST-P3	ST-P3 [34]	1.33	2.11	2.90	2.11	0.23	0.62	1.27	0.71
	VAD [40]	**0.17**	0.34	0.60	0.37	0.07	0.10	0.24	0.14
	GPT-Driver [58]	0.20	0.40	0.70	0.44	0.04	0.12	0.36	0.17
	FeD	0.21	**0.33**	**0.49**	**0.34**	**0.00**	**0.03**	**0.15**	**0.06**
UniAD	NMP [94]	-	-	2.31	-	-	-	1.92	-
	SA-NMP [94]	-	-	2.05	-	-	-	1.59	-
	FF [33]	0.55	1.20	2.54	1.43	0.06	0.17	1.07	0.43
	EO [43]	0.67	1.36	2.78	1.60	0.04	0.09	0.88	0.33
	UniAD [35]	0.48	0.96	1.65	1.03	0.05	0.17	0.71	0.31
	GPT-Driver [58]	**0.27**	0.74	1.52	0.84	0.07	0.15	1.10	0.44
	FeD	**0.27**	**0.53**	**0.94**	**0.58**	**0.00**	**0.04**	**0.52**	**0.19**

4.2. Ablation Studies

Impact of Privileged Prompting and Distillation: Table 1 depicts incorporating the privileged environmental information about the surrounding objects, traffic lights and

[1]The issue has been publicly discussed on Github: https://github.com/hustvl/VAD/issues/33



GT Feedback

Given the predicted future locations of ego vehicle, the following instances would occur:

Large deviation with planned route at 1.25 seconds in the future, with an error of 0.72 meters.
Large deviation with planned route at 2.0 seconds in the future, with an error of 0.57 meters.
Large deviation with expert route at 2.25 seconds in the future, with an error of 0.80 meters.
Large deviation with expert route at 2.5 seconds in the future, with an error of 1.20 meters.

Predicted Feedback

Given the predicted future locations of ego vehicle, the following instances would occur:

Large deviation with planned route at 1.5 seconds in the future, with an error of 0.58 meters.
Large deviation with planned route at 1.75 seconds in the future, with an error of 0.54 meters.
Large deviation with expert route at 1.75 seconds in the future, with an error of 0.54 meters.
Large deviation with expert route at 2.0 seconds in the future, with an error of 1.75 meters.
Large deviation with expert route at 2.5 seconds in the future, with an error of 0.54 meters.

Figure 4. **Example Post-Feedback Prediction.** We present an example of our model waypoint and feedback reasoning predictions. Top-left: Front camera RGB image. Bottom-left: BEV visualization of surrounding environment [101] along with ground truth waypoints (green), initial waypoint predictions before feedback fine-tuning (orange), and corrected waypoint predictions after feedback fine-tuning (blue). Top-right: Ground-truth feedback reasoning. Bottom-right: Feedback predicted by FeD.

the planned path into the language prompt improves the DS from 69.14 to 80.51 and achieves the best IS of 0.92. We note that providing extensive supervision from the privileged agent over the internal features to the sensorimotor agent results in a 4.8% improvement in DS.

Impact of Feedback Reasoning: Table 1 demonstrates that fine-tuning the sensorimotor agent with the proposed feedback reasoning mechanism significantly boosts the driving performance by 12.6% in DS, 12.5% in RC and 7.7% in IS. We also observe that the resulting FeD sensorimotor agent even achieves a higher DS than the privileged agent as the agent learns to reason over and rectify errors based on its own past experiences. More ablations regarding distillation and feedback reasoning are shown in the supplementary.

4.3. Qualitative Results

Fig. 4 depicts the feedback reasoning qualitative results generated by FeD model on CARLA validation set. We visualize a complete scenario: the front camera image, BEV representation [101] of the scene with the originally predicted waypoints, ground truth waypoints, corrected waypoints prediction, the ground truth feedback reasoning, and the predicted feedback reasoning given the originally predicted waypoints. We find that our FeD model is able to generate reasonable feedback about the original prediction errors and predict corrected waypoints accordingly. This validates our hypothesis that providing extensive corrective language feedback benefits sensorimotor driving learning. For example, the initial model struggled to imitate the expert demonstration while changing the lane to the right, leading to a potential off-road deviation as shown in Fig. 4. Our feedback ground truth offers comprehensive corrective guidance for all inaccurately predicted waypoints. After the feedback fine-tuning, the FeD model demonstrates the ability to not only give corrective feedback about the initial wrong predictions but also predict the safe waypoints. More qualitative examples are provided in the supplementary.

5. Conclusion

We present FeD, the first sensorimotor end-to-end LLM-based autonomous driving model. FeD enables efficient closed-loop evaluation compared with the existing LLM-based methods, which often leverage slow and costly inference. In addition to the efficient architecture, FeD is trained from rich supervision, learning via distillation from a privileged agent with BEV information of the scene. We further utilize detailed corrective feedback regarding the reasons behind driving prediction failures, thus allowing the sensorimotor agent to better learn from its own mistakes. FeD obtains state-of-the-art results in both closed-loop simulation and open-loop real-world evaluation. Given the computational overhead of LLMs, future work includes overcoming limitations for real-time applications, e.g., through more effective distillation strategies. Leveraging various types of feedback, e.g., more coarse high-level feedback, can further increase the usability of the proposed feedback-based mechanism in the future. Finally, improved sample efficiency could be pursued by future work, as current fine-tuning requirements for fine-tuning involve hundreds of iterations to effectively leverage feedback.

Acknowledgments: We thank the Red Hat Collaboratory (award #2024-01-RH02), the Rafik B. Hariri Institute for Computing and Computational Science and Engineering (award #2023-07-001), and the National Science Foundation (IIS-2152077) for supporting this research.

References

[1] Carla autonomous driving leaderboard. *https://leaderboard.carla.org/*, 2022. 3, 6

[2] Mayank Bansal, Alex Krizhevsky, and Abhijit Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. In *RSS*, 2019. 1, 2, 3

[3] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016. 1

[4] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. *CoRL*, 2023. 2

[5] 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 *CVPR*, 2020. 6, 7

[6] Minette Alexandra Sy Chan, Yunn Chyi Chao, and Gloria Isabel Miller. How to teach an adult to ride a bicycle. 2001. 1

[7] Raphael Chekroun, Marin Toromanoff, Sascha Hornauer, and Fabien Moutarde. Gri: General reinforced imitation and its application to vision-based autonomous driving. *Robotics*, 2023. 2

[8] Chenyi Chen, Ari Seff, Alain L. Kornhauser, and Jianxiong Xiao. DeepDriving: Learning affordance for direct perception in autonomous driving. In *ICCV*, 2015. 2

[9] Dian Chen and Philipp Krähenbühl. Learning from all vehicles. In *CVPR*, 2022. 2, 3, 6, 7

[10] Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by cheating. In *CoRL*, 2020. 2, 3

[11] Dian Chen, Vladlen Koltun, and Philipp Krähenbühl. Learning to drive from a world on rails. In *ICCV*, 2021. 2, 5, 7

[12] Long Chen, Oleg Sinavski, Jan Hünermann, Alice Karnsund, Andrew James Willmott, Danny Birch, Daniel Maund, and Jamie Shotton. Driving with LLMs: Fusing object-level vector modality for explainable autonomous driving. *arXiv preprint arXiv:2310.01957*, 2023. 2, 3

[13] Kashyap Chitta, Aditya Prakash, and Andreas Geiger. Neat: Neural attention fields for end-to-end autonomous driving. In *ICCV*, 2021. 7

[14] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *PAMI*, 2022. 3, 5, 7

[15] Felipe Codevilla, Eder Santana, Antonio M López, and Adrien Gaidon. Exploring the limitations of behavior cloning for autonomous driving. In *ICCV*, 2019. 1

[16] Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, and Ziran Wang. Drive as you speak: Enabling human-like interaction with large language models in autonomous vehicles. *arXiv preprint arXiv:2309.10228*, 2023. 2

[17] Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, and Ziran Wang. Receive, reason, and react: Drive as you say with large language models in autonomous vehicles. *arXiv preprint arXiv:2310.08034*, 2023. 2, 3

[18] Daniel Dauner, Marcel Hallgarten, Andreas Geiger, and Kashyap Chitta. Parting with misconceptions about learning-based vehicle motion planning. *CoRL*, 2023. 2

[19] Pim de Haan, Dinesh Jayaraman, and Sergey Levine. Causal confusion in imitation learning. In *NeurIPS*, 2019. 1, 2

[20] Vikrant Dewangan, Tushar Choudhary, Shivam Chandhok, Shubham Priyadarshan, Anushka Jain, Arun K Singh, Siddharth Srivastava, Krishna Murthy Jatavallabhula, and K Madhava Krishna. Talk2bev: Language-enhanced bird's-eye view maps for autonomous driving. *arXiv preprint arXiv:2310.02251*, 2023. 2

[21] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *CoRL*, 2017. 2, 3, 6

[22] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023. 1, 2

[23] Daocheng Fu, Xin Li, Licheng Wen, Min Dou, Pinlong Cai, Botian Shi, and Yu Qiao. Drive like a human: Rethinking autonomous driving with large language models. *arXiv preprint arXiv:2307.07162*, 2023. 2, 3

[24] Ruiyuan Gao, Kai Chen, Enze Xie, Lanqing Hong, Zhenguo Li, Dit-Yan Yeung, and Qiang Xu. Magicdrive: Street view generation with diverse 3d geometry control. *arXiv preprint arXiv:2310.02601*, 2023. 2

[25] Shalini Ghosh, Giedrius Burachas, Arijit Ray, and Avi Ziskind. Generating natural language explanations for visual question answering using scene graphs and visual attention. *arXiv preprint arXiv:1902.05715*, 2019. 3

[26] James G Greeno, Allan M Collins, Lauren B Resnick, et al. Cognition and learning. *Handbook of educational psychology*, 1996. 1

[27] Saurabh Gupta, James Davidson, Sergey Levine, Rahul Sukthankar, and Jitendra Malik. Cognitive mapping and planning for visual navigation. In *CVPR*, 2017. 2

[28] 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 *ICRA*, 2020. 1

[29] Lisa Anne Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, and Trevor Darrell. Generating visual explanations. In *ECCV*, 2016. 3

[30] Anthony Hu, Gianluca Corrado, Nicolas Griffiths, Zachary Murez, Corina Gurau, Hudson Yeo, Alex Kendall, Roberto Cipolla, and Jamie Shotton. Model-based imitation learning for urban driving. *NeurIPS*, 2022. 1

[31] Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023. 2

[32] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 6

[33] Peiyun Hu, Aaron Huang, John Dolan, David Held, and Deva Ramanan. Safe local motion planning with self-supervised freespace forecasting. In *CVPR*, 2021. 7

[34] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *ECCV*, 2022. 7

[35] 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 *CVPR*, 2023. 1, 7

[36] Zanming Huang, Zhongkai Shangguan, Jimuyang Zhang, Gilad Bar, Matthew Boyd, and Eshed Ohn-Bar. ASSISTER: Assistive navigation via conditional instruction generation. In *ECCV*, 2022.

[37] Bernhard Jaeger, Kashyap Chitta, and Andreas Geiger. Hidden biases of end-to-end driving models. *arXiv preprint arXiv:2306.07957*, 2023. 1, 2, 7

[38] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *ICML*, 2021. 2

[39] Xiaosong Jia, Yulu Gao, Li Chen, Junchi Yan, Patrick Langechuan Liu, and Hongyang Li. Driveadapter: Breaking the coupling barrier of perception and planning in end-to-end autonomous driving. In *ICCV*, 2023. 1

[40] 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 *ICCV*, 2023. 7

[41] Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. Vima: General robot manipulation with multimodal prompts. In *ICML*, 2023. 2

[42] Parag Khanna, Elmira Yadollahi, Mårten Björkman, Iolanda Leite, and Christian Smith. User study exploring the role of explanation of failures by robots in human robot collaboration tasks. *arXiv preprint arXiv:2303.16010*, 2023. 3

[43] Tarasha Khurana, Peiyun Hu, Achal Dave, Jason Ziglar, David Held, and Deva Ramanan. Differentiable raycasting for self-supervised occupancy forecasting. In *ECCV*, 2022. 7

[44] Lei Lai, Zhongkai Shangguan, Jimuyang Zhang, and Eshed Ohn-Bar. Xvo: Generalized visual odometry via cross-modal self-training. In *ICCV*, 2023. 2

[45] Zhiling Lan, Ziming Zheng, and Yawei Li. Toward automated anomaly identification in large-scale systems. *TPDS*, 2009. 2

[46] Hyang-Jung Lee, Heeseung Lee, Chae Young Lim, Issac Rhim, and Sang-Hun Lee. Corrective feedback guides human perceptual decision-making by informing about the world state rather than rewarding its choice. *PLoS biology*, 2023. 1

[47] Edouard Leurent. An environment for autonomous driving decision-making. https://github.com/eleurent/highway-env, 2018. 2, 3

[48] Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A multi-modal model with in-context instruction tuning. *arXiv preprint arXiv:2305.03726*, 2023. 2

[49] Guohao Li, Matthias Mueller, Vincent Casser, Neil Smith, Dominik L Michels, and Bernard Ghanem. Oil: Observational imitation learning. In *RSS*, 2019. 2

[50] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 1, 2

[51] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *ICRA*, 2023. 2

[52] Kevin Lin, Christopher Agia, Toki Migimatsu, Marco Pavone, and Jeannette Bohg. Text2motion: From natural language instructions to feasible plans. *arXiv preprint arXiv:2303.12153*, 2023. 2

[53] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 1, 2, 3

[54] Zeyi Liu, Arpit Bahety, and Shuran Song. Reflect: Summarizing robot experiences for failure explanation and correction. *arXiv preprint arXiv:2306.15724*, 2023. 3

[55] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 6

[56] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[57] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019. 2, 3

[58] Jiageng Mao, Yuxi Qian, Hang Zhao, and Yue Wang. Gpt-driver: Learning to drive with gpt. *arXiv preprint arXiv:2310.01415*, 2023. 1, 2, 3, 7

[59] Janet Metcalfe. Learning from errors. *Annual review of psychology*, 2017. 1

[60] Matthew H Olson and Julio J Ramírez. *An introduction to theories of learning*. 2020. 1

[61] OpenAI. Gpt-4 technical report, 2023. 1, 2

[62] Takayuki Osa, Joni Pajarinen, Gerhard Neumann, J Andrew Bagnell, Pieter Abbeel, Jan Peters, et al. An algorithmic perspective on imitation learning. *Foundations and Trends® in Robotics*, 2018. 2

[63] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023. 2

[64] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018. 3

[65] Dean A Pomerleau. ALVINN: An autonomous land vehicle in a neural network. In *NeurIPS*, 1989. 2

[66] Tianwen Qian, Jingjing Chen, Linhai Zhuo, Yang Jiao, and Yu-Gang Jiang. Nuscenes-qa: A multi-modal visual question answering benchmark for autonomous driving scenario. *arXiv preprint arXiv:2305.14836*, 2023. 2

[67] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 3

[68] Arijit Ray, Yi Yao, Rakesh Kumar, Ajay Divakaran, and Giedrius Burachas. Can you explain that? lucid explanations help human-ai collaborative image retrieval. In *HCOMP*, 2019. 3

[69] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, et al. A generalist agent. *arXiv preprint arXiv:2205.06175*, 2022. 2

[70] Katrin Renz, Kashyap Chitta, Otniel-Bogdan Mercea, A. Sophia Koepke, Zeynep Akata, and Andreas Geiger. Plant: Explainable planning transformers via object-level representations. In *CoRL*, 2022. 2

[71] Matthew Schmittle, Sanjiban Choudhury, and Siddhartha S Srinivasa. Learning online from corrective feedback: A meta-algorithm for robotics. *arXiv preprint arXiv:2104.01021*, 2021. 1

[72] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. *IJCV*, 2019. 3

[73] Hao Sha, Yao Mu, Yuxuan Jiang, Li Chen, Chenfeng Xu, Ping Luo, Shengbo Eben Li, Masayoshi Tomizuka, Wei Zhan, and Mingyu Ding. Languagempc: Large language models as decision makers for autonomous driving. *arXiv preprint arXiv:2310.03026*, 2023. 2

[74] Dhruv Shah, Błażej Osiński, Sergey Levine, et al. Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *CoRL*, 2023. 2

[75] Dhruv Shah, Błażej Osiński, Sergey Levine, et al. Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *CoRL*, 2023. 2

[76] Shai Shalev-Shwartz and Amnon Shashua. On the sample complexity of end-to-end training vs. semantic abstraction training. *arXiv preprint arXiv:1604.06915*, 2016. 2

[77] Hao Shao, Letian Wang, Ruobing Chen, Hongsheng Li, and Yu Liu. Safety-enhanced autonomous driving using interpretable sensor fusion transformer. In *CoRL*, 2023. 2

[78] Hao Shao, Letian Wang, Ruobing Chen, Steven L Waslander, Hongsheng Li, and Yu Liu. Reasonnet: End-to-end driving with temporal and global reasoning. In *CVPR*, 2023. 2

[79] Chan Hee Song, Jiaman Wu, Clayton Washington, Brian M Sadler, Wei-Lun Chao, and Yu Su. Llm-planner: Few-shot grounded planning for embodied agents with large language models. In *ICCV*, 2023. 1, 2, 3

[80] Jonathan Spencer, Sanjiban Choudhury, Arun Venkatraman, Brian Ziebart, and J Andrew Bagnell. Feedback in imitation learning: The three regimes of covariate shift. *arXiv preprint arXiv:2102.02872*, 2021. 1

[81] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019. 2

[82] Shuhan Tan, Boris Ivanovic, Xinshuo Weng, Marco Pavone, and Philipp Kraehenbuehl. Language conditioned traffic generation. In *CoRL*, 2023. 2

[83] Marin Toromanoff, Emilie Wirbel, and Fabien Moutarde. End-to-end model-free reinforcement learning for urban driving using implicit affordances. In *CVPR*, 2020. 2

[84] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1, 2, 4

[85] Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, and Jiwen Lu. Drivedreamer: Towards real-world-driven world models for autonomous driving. *arXiv preprint arXiv:2309.09777*, 2023. 2

[86] Chuan Wen, Jierui Lin, Trevor Darrell, Dinesh Jayaraman, and Yang Gao. Fighting copycat agents in behavioral cloning from observation histories. In *NeurIPS*, 2020. 1, 2

[87] Licheng Wen, Daocheng Fu, Xin Li, Xinyu Cai, Tao Ma, Pinlong Cai, Min Dou, Botian Shi, Liang He, and Yu Qiao. Dilu: A knowledge-driven approach to autonomous driving with large language models. *arXiv preprint arXiv:2309.16292*, 2023. 2, 3

[88] Dongming Wu, Wencheng Han, Tiancai Wang, Yingfei Liu, Xiangyu Zhang, and Jianbing Shen. Language prompt for autonomous driving. *arXiv preprint arXiv:2309.04379*, 2023. 2

[89] 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. *arXiv preprint arXiv:2206.08129*, 2022. 2, 5, 7

[90] Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kenneth KY Wong, Zhenguo Li, and Hengshuang Zhao. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *arXiv preprint arXiv:2310.01412*, 2023. 1, 2, 3

[91] Sean Ye, Glen Neville, Mariah Schrum, Matthew Gombolay, Sonia Chernova, and Ayanna Howard. Human trust after robot mistakes: Study of the effects of different forms of robot communication. In *RO-MAN*, 2019. 3

[92] Naoki Harrison Yokoyama, Sehoon Ha, Dhruv Batra, Jiuguang Wang, and Bernadette Bucher. Vlfm: Vision-language frontier maps for zero-shot semantic navigation. In *2nd Workshop on Language and Robot Learning: Language as Grounding*, 2023. 2

[93] Wenhao Yu, Nimrod Gileadi, Chuyuan Fu, Sean Kirmani, Kuang-Huei Lee, Montse Gonzalez Arenas, Hao-Tien Lewis Chiang, Tom Erez, Leonard Hasenclever, Jan Humplik, et al. Language to rewards for robotic skill synthesis. *arXiv preprint arXiv:2306.08647*, 2023. 2

[94] Wenyuan Zeng, Wenjie Luo, Simon Suo, Abbas Sadat, Bin Yang, Sergio Casas, and Raquel Urtasun. End-to-end interpretable neural motion planner. In *CVPR*, 2019. 7

[95] Chris Zhang, Runsheng Guo, Wenyuan Zeng, Yuwen Xiong, Binbin Dai, Rui Hu, Mengye Ren, and Raquel Urtasun. Rethinking closed-loop training for autonomous driving. In *ECCV*, 2022. 1

[96] Jimuyang Zhang and Eshed Ohn-Bar. Learning by watching. In *CVPR*, 2021. 2

[97] Jimuyang Zhang, Minglan Zheng, Matthew Boyd, and Eshed Ohn-Bar. X-world: Accessibility, vision, and autonomy meet. In *ICCV*, 2021. 1

[98] Jimuyang Zhang, Ruizhao Zhu, and Eshed Ohn-Bar. Selfd: self-learning large-scale driving policies from the web. In *CVPR*, 2022. 2

[99] Jimuyang Zhang, Zanming Huang, and Eshed Ohn-Bar. Coaching a teachable student. In *CVPR*, 2023. 1, 2, 3, 7

[100] Lunjun Zhang, Yuwen Xiong, Ze Yang, Sergio Casas, Rui Hu, and Raquel Urtasun. Learning unsupervised world models for autonomous driving via discrete diffusion. *arXiv preprint arXiv:2311.01017*, 2023. 2

[101] Zhejun Zhang, Alexander Liniger, Dengxin Dai, Fisher Yu, and Luc Van Gool. End-to-end urban driving by imitating a reinforcement learning coach. In *ICCV*, 2021. 2, 5, 8

[102] Zhejun Zhang, Alexander Liniger, Dengxin Dai, Fisher Yu, and Luc Van Gool. Trafficbots: Towards world models for autonomous driving simulation and motion prediction. *arXiv preprint arXiv:2303.04116*, 2023. 2

[103] Albert Zhao, Tong He, Yitao Liang, Haibin Huang, Guy Van den Broeck, and Stefano Soatto. Lates: Latent space distillation for teacher-student driving policy learning. *arXiv preprint arXiv:1912.02973*, 2019. 2

[104] Brady Zhou, Philipp Krähenbühl, and Vladlen Koltun. Does computer vision matter for action? *Science Robotics*, 4(30), 2019. 2

[105] Baichuan Zhou, Ying Hu, Xi Weng, Junlong Jia, Jie Luo, Xien Liu, Ji Wu, and Lei Huang. Tinyllava: A framework of small-scale large multimodal models. *arXiv preprint arXiv:2402.14289*, 2024. 7

[106] Hattie Zhou, Azade Nova, Hugo Larochelle, Aaron Courville, Behnam Neyshabur, and Hanie Sedghi. Teaching algorithmic reasoning via in-context learning. *arXiv preprint arXiv:2211.09066*, 2022. 2

[107] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 1, 2

[108] Ruizhao Zhu, Peng Huang, Eshed Ohn-Bar, and Venkatesh Saligrama. Learning to drive anywhere. In *CoRL*, 2023. 1, 2