VLASCD: A Visual Language Action Model for Simultaneous Chatting and Decision Making

Zuojin Tang[1,2], Bin Hu[3], Chenyang Zhao[4], De Ma[1], Gang Pan[1], Bin Liu[2]*
[1]College of Computer Science and Technology, Zhejiang University
[2]Home Robotics Lab, E-surfing Digital Life Technology Co., Ltd., China Telecom
[3]Zhejiang Lab, [4]Trinity College Dublin
{zuojintang, made, gpan}@zju.edu.cn
hubin@zhejianglab.com; zhaoc4@tcd.ie; bins@ieee.org

Abstract

Recent large pretrained models such as LLMs (e.g., GPT series) and VLAs (e.g., OpenVLA) have achieved notable progress on multimodal tasks, yet they are built upon a multi-input single-output (MISO) paradigm. We show that this paradigm fundamentally limits performance in multi-input multi-output (MIMO) scenarios, where parallel task execution is required. In MISO architectures, tasks compete for a shared output channel, creating mutual exclusion effects that cause unbalanced optimization and degraded performance. To address this gap, we introduce MIMO-VLA (VLASCD), a unified training framework that enables concurrent multi-task outputs, exemplified by simultaneous dialogue generation and decision-making. Inspired by human cognition, MIMO-VLA eliminates interference between tasks and supports efficient parallel processing. Experiments on the CARLA autonomous driving platform demonstrate that MIMO-VLA substantially outperforms state-of-the-art MISO-based LLMs, reinforcement learning models, and VLAs in MIMO settings, establishing a new direction for multimodal and multitask learning. Our code is available at: *https://github.com/Mark-zjtang/MIMO-VLA*

1 Introduction

Since the emergence of ChatGPT, large language models (LLMs) have become the prototypical large-scale pretrained models. Trained on vast corpora of text and code, they capture rich world knowledge and exhibit strong generalization, including in-context learning and reasoning (e.g., chain-of-thought (Wei et al., 2022)). A growing trend is the extension of LLMs beyond language tasks such as dialogue and text generation toward decision-making in open physical environments.

Currently, three main paradigms exist for building large-scale pretrained models for decision-making in open physical environments. (1) Sequence modeling: methods such as Decision Transformers (Chen et al., 2021) serialize the decision-making process, treating actions as tokens analogous to language; however, they rely heavily on large, high-quality decision-making datasets. (2) Hierarchical modular systems: LLMs perform high-level planning (Chen et al., 2024; Carta et al., 2023; Hu et al., 2024; Zhou et al., 2024), decomposing tasks and coordinating specialized modules or tools. (3) End-to-end Vision-Language-Action (VLA) models: approaches like (Padalkar et al., 2023; Kim et al., 2024) bypass modularity altogether, directly mapping multimodal inputs to action outputs.

In recent years, pretrained large models have achieved remarkable progress on multimodal tasks, with LLMs (e.g., ChatGPT) and VLAs (e.g., OpenVLA) as representative examples. These models typically follow a multi-input single-output (MISO) architecture, where multiple inputs produce a single output, and have shown strong performance in text generation and visual understanding.

However, our investigation reveals fundamental limitations of MISO architectures in multi-input multi-output (MIMO) scenarios, such as parallel multi-task execution. Existing MISO LLMs (Chen et al., 2024; Liu et al., 2023) often fail to generate effective actions without compromising dialogue, while MISO VLAs (Kim et al., 2024; Liu et al., 2024) struggle to produce coherent dialogue at all. This stems from task interference: competing tasks contend for shared output channels, leading to imbalanced optimization and degraded performance. By contrast, human cognition inherently supports non-interfering concurrent task execution (e.g., dialogue and decision-making).

Motivated by this gap, we propose MIMO-VLA (VLASCD), a unified MIMO training architecture with parallel multi-task output capabilities, instantiated as a visual-language-action model for simulta-

*Corresponding Author

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 9212–9232
November 4-9, 2025

neous dialogue and decision-making. Evaluated on the CARLA 0.9.10 autonomous driving platform (Dosovitskiy et al., 2017), MIMO-VLA demonstrates that task-adaptive distributed output mapping not only enables efficient multimodal collaboration but also resolves the interference bottleneck of existing MISO models.

The main contributions of this work are:

- We provide the first evidence that existing MISO models (e.g., LLMs and VLAs) fundamentally fail to handle MIMO tasks.
- We propose MIMO-VLA, a unified MIMO training architecture with parallel multi-task output capabilities. The design integrates several validated techniques: (1) a computational module and loss term for generating continuous action values; (2) an image reconstruction loss to exploit rich visual information during text generation and decision-making; (3) a label smoothing strategy to preserve dialogue capabilities while enhancing decision accuracy.
- Extensive experiments demonstrate that MIMO-VLA achieves more accurate real-time action decisions than state-of-the-art (SOTA) baselines, while fully retaining real-time dialogue functionality.

2 Related Work

2.1 LLMs for decision-making

Since (Brown et al., 2020), GPT has established itself as the dominant paradigm for LLMs. Successive models such as GPT-3.5 and GPT-4 (OpenAI, 2023a) have demonstrated strong zero-shot generalization and reasoning. The open-source LLaMA series (Touvron et al., 2023a,b) further accelerated progress, while advances such as chain-of-thought reasoning (Wei et al., 2022) and ReAct (Yao et al., 2022) improved reasoning and action generation. More recently, LLMs have been integrated into hierarchical modular decision-making agents, where they typically perform high-level planning rather than directly generating decisions (Ahn et al., 2022; Fu et al., 2023; Carta et al., 2023; Chen et al., 2024; Xu et al., 2024; Sha et al., 2023; Hu et al., 2024; Zhou et al., 2024).

In contrast, the proposed MIMO-VLA can be viewed as a multimodal GPT variant fine-tuned for downstream application scenarios, distinguished by its ability to simultaneously output action decisions and natural-language dialogue.

2.2 VLA model for decison-making

Vision-Language-Action (VLA) models integrate multimodal inputs for embodied decision-making. Unlike conversational LLMs such as ChatGPT, VLAs generate control signals for physical agents (e.g., robots) interacting with the environment. By combining language understanding, visual perception, and action generation (Huang et al., 2023; Li et al., 2023b; Zhen et al., 2024; Dorka et al.), they excel at instruction-following tasks. Compared to deep reinforcement learning (RL) approaches, VLAs demonstrate superior versatility, flexibility, and generality in complex environments (Padalkar et al., 2023; Brohan et al., 2023; et al, 2024; Team et al., 2024; Li et al., 2023c; Bai et al., 2023; Li et al., 2022, 2023a; Liu et al., 2024; Tan and Bansal, 2019).

However, leading VLA models such as RT-X (Padalkar et al., 2023) and OpenVLA (Kim et al., 2024) typically discretize continuous action spaces into fixed intervals. This discretization imposes a critical limitation, preventing them from handling the fine-grained continuous actions required for nuanced operations in complex tasks.

2.3 LLMs for MIMO settings

Existing multitasking approaches often rely on task-specific designs, which hinder collaborative optimization and increase computational cost (Geng et al., 2022; Liu et al., 2023; Ouyang et al., 2022; Driess et al., 2023). Recent efforts toward unified frameworks—such as multitask fine-tuning with CGC-LoRA for LLMs (Song et al., 2024) or generalized policies for multi-task learning (Driess et al., 2023; Alayrac et al., 2022; Reed et al., 2022; Jiang et al., 2022; Ahn et al., 2022)—remain constrained to MISO architectures and thus cannot produce MIMO outputs. The most recent Simlingo framework (Renz et al., 2025) shares certain conceptual similarities, but our work differs in three key aspects: (1) we identify and analyze multi-task interference in MISO models; (2) we introduce end-to-end continuous action mapping; (3) we design a dynamic composite loss integrating language, action, and image objectives to enable more efficient multimodal learning.

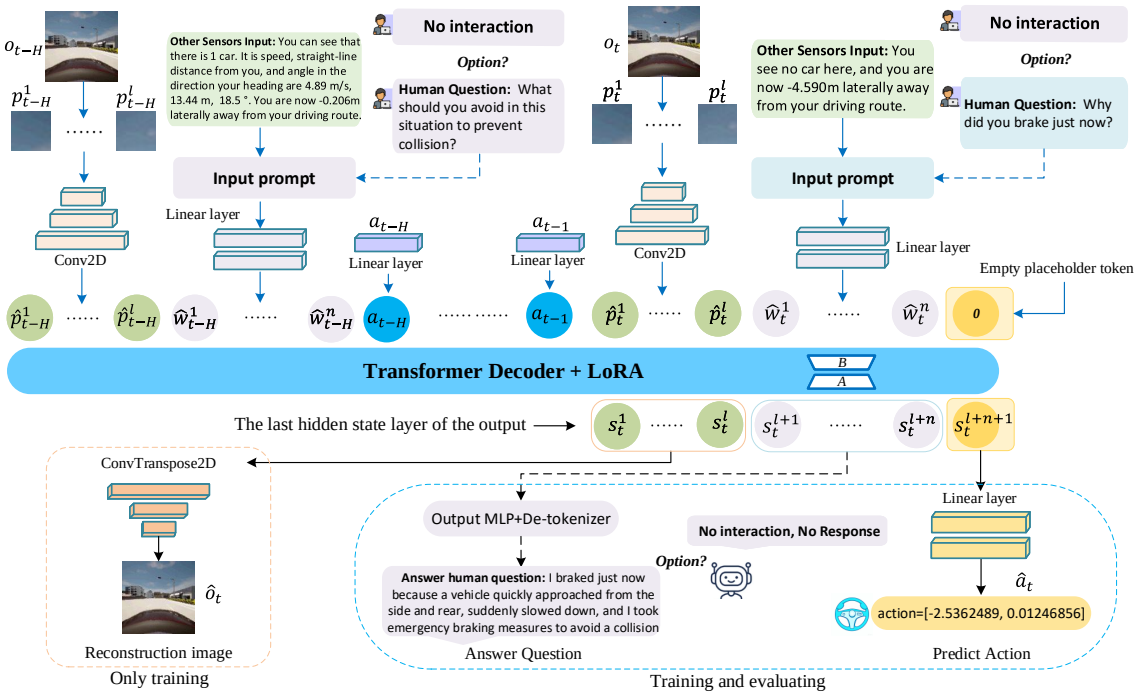


Figure 1: Overview of the MIMO-VLA framework. Expert dataset images, text descriptions, and action values are first mapped into feature representations through linear encoders. These representations are concatenated in a fixed order and fed into a transformer backbone. In the final layer of the LoRA-tuned model, we (i) reconstruct sensor outputs (training only), (ii) generate query responses, and (iii) map continuous actions (during both training and evaluation).

3 Methodology

In this section, we present how to build MIMO-VLA in detail, including the model architecture and the training procedure, with a focus on the loss designs in the last output hiddens layer. An overview of MIMO-VLA is illustrated in Figure 1. To begin with, we present the problem setting of our concern.

3.1 Problem Setting

We consider a multimodal setting similar to (Xiao et al., 2020), where at each time step t, the agent executes an action a_t and the environment returns visual and textual observations, denoted $\{o_t, \hat{w}_t\}$. Our goal is to learn a generative model $\pi(\hat{a}_t, \hat{w*}_t | o_{t-H}, \hat{w}_{t-H}, a_{t-H}, ..., o_t, \hat{w}_t)$ that produces both high-quality action decisions $\hat{a}_t$ and textual responses $\hat{w*}_t$ conditioned on a sequence of past trajectories. Here, H denotes the context length.

3.2 Model Architecture

Our model supports three input modalities: text, image, and numeric vectors. We adopt LLaMA-7B (Touvron et al., 2023b) as the backbone and encode textual inputs using its pretrained embedding layers. Visual inputs are encoded following standard practices in visual-language models (VLMs) (Liu et al., 2024) and VLAs (Kim et al., 2024): each image o_t is first divided into L patches $p_l, l = 1, \ldots, L$, which are then mapped into the vector space via a trainable 2D convolutional network. For numeric action inputs, a multi-layer perceptron (MLP) encodes action values into the same vector space. Finally, embeddings from all three modalities are concatenated to form the trajectory embedding sequence τ at time t, defined as follows:

$$\tau_t = \{(\hat{p}^1_{t-H}, \ldots, \hat{p}^L_{t-H}), (\hat{w}^1_{t-H}, \ldots, \hat{w}^n_{t-H}), a_{t-H}, \ldots, (\hat{p}^1_t, \ldots, \hat{p}^l_t), (\hat{w}^1_t, \ldots, \hat{w}^n_t)\} \quad (1)$$

where $\hat{p}^i_t$ and $\hat{w}^j_t$ denote the embeddings of the i-th visual patch and the j-th textual token at time step t, respectively.

During inference, the transformer backbone of MIMO-VLA generates hidden embeddings $s^{l+1}_t, \cdots, s^{l+n+1}_t$ (Figure 1), which are subsequently decoded into outputs for different modalities. MIMO-VLA supports two output modalities: textual responses for dialogue and numeric vectors for action-level decision-making. For di-

alogue, we leverage the pretrained output MLP layers and tokenizer of LLaMA-7B to generate text. For action decisions, the model generates an additional embedding vector following the "< EOS >"placeholder token. Unlike prior approaches such as OpenVLA (Kim et al., 2024) and RT-X (Brohan et al., 2023), which treat action prediction as a token generation task by discretizing the action space into bins, we employ an action head comprising multiple MLP modules that directly map embeddings to continuous action values. We empirically find that this design achieves superior performance compared to discretization-based methods.

3.3 Training Procedure

We fine-tune the transformer backbone using LoRA (Hu et al., 2021) and jointly train the image, text, and action encoding modules, as well as the decoding modules, on an offline dataset D_{expert} containing demonstrated driving trajectories paired with question-answer (QA) annotations. The model learns to predict control actions and respond to driving-related queries, e.g., "*Summarize the current driving scenario*." An auxiliary image reconstruction task is incorporated, where a transposed convolution layer reconstructs input image patches from the output embeddings $s_t^1, \cdots, s_t^l$ to enhance feature learning. The overall training objective comprises three loss terms—text generation, action prediction, and image reconstruction—where ϕ denotes decoder parameters and θ represents all other trainable parameters.

Text Generation. In our experiments, we observed that simply replacing specific numerical values in the translation template (Chen et al., 2024) produces minimal representational differences due to the sequential nature of the data. Consequently, using conventional cross-entropy loss for text generation can easily lead to model overfitting (see Appendix A.8 for details). To address this, we adopt label smoothing (Szegedy et al., 2016) to regularize training. Specifically, the hard label for token w_i is softened by distributing a small portion of the probability mass to incorrect classes:

$$q_i^k = \begin{cases} 1-\epsilon & \text{if } k = y_i, \\ \frac{\epsilon}{K-1} & \text{otherwise,} \end{cases} \tag{2}$$

where ϵ is the smoothing factor and K is the number of total classes, i.e., the vocabulary size. Accordingly, the final loss term used for text generation is:

$$\mathcal{L}_{\text{language}}(\theta) = \frac{1}{N} \sum_i \sum_k q_i^k \log p(k|\tau^{:i-1}, \theta), \tag{3}$$

where $\tau^{:i-1}$ denotes the input token sequence preceding position i, used to predict token i, and N represents the maximum padding length for input text normalization.

Action Prediction. To directly predict continuous action values instead of discretized action bins, we train the model using a mean squared error (MSE) loss between the ground-truth action a_t and the predicted action $\hat{a}_t$, defined as follows:

$$\mathcal{L}_{\text{action}}(\theta) = \frac{1}{T} \sum_{t=1}^{T} \frac{1}{D} \sum_{d=1}^{D} [(a_t^d - \pi_\theta(\hat{a}_t^d|\tau_t))^2] \tag{4}$$

where T denotes the sequence length per training sample. Where D denotes the dimensionality of the action space. In our experiments, $D = 2$, corresponding to vehicle acceleration and steering.

Image Reconstruction. To better exploit the rich environmental information in visual inputs while mitigating information loss under limited data, we introduce an auxiliary image reconstruction task. This provides additional supervision for the visual modality by using a 2D transposed convolution layer f_ϕ to reconstruct image patches from their embeddings. The reconstruction loss is defined as the pixel-wise Euclidean distance between the original and reconstructed patches:

$$\mathcal{L}_{\text{image}}(\theta, \phi) = \frac{1}{L} \sum_{l=1}^{L} \text{mse}(o_t, f_\phi(\pi_\theta(g_\theta(\tau_t^{:p_t^l})))) \tag{5}$$

where o_t denotes the input image, $\tau_t^{:p_t^l}$ represents the input sequence up to patch p_t^l, and g_θ is a trainable 2D convolutional network that maps image patches $p_t^1, \cdots, p_t^l$ directly into the language embedding space $\hat{p}_t^1, \cdots, \hat{p}_t^l$.

Loss Function for Model Training In summary, the overall training loss for our model is defined as:

$$\mathcal{L} = \alpha_1 \mathcal{L}_{\text{language}} + \alpha_2 \mathcal{L}_{\text{action}} + \lambda \mathcal{L}_{\text{image}} \tag{6}$$

where $\alpha_1, \alpha_2, \lambda$ are the weighting hyperparameters for the three loss components. Importantly, we explicitly implement a gradient-space isolation mechanism: $\mathcal{L}_{\text{language}}$ affects only text token positions, $\mathcal{L}_{\text{action}}$ affects only action token positions, and $\mathcal{L}_{\text{image}}$ affects only image patch positions. Ablation studies (Table 4) systematically validate the independent contribution of each component, demonstrating that this design effectively

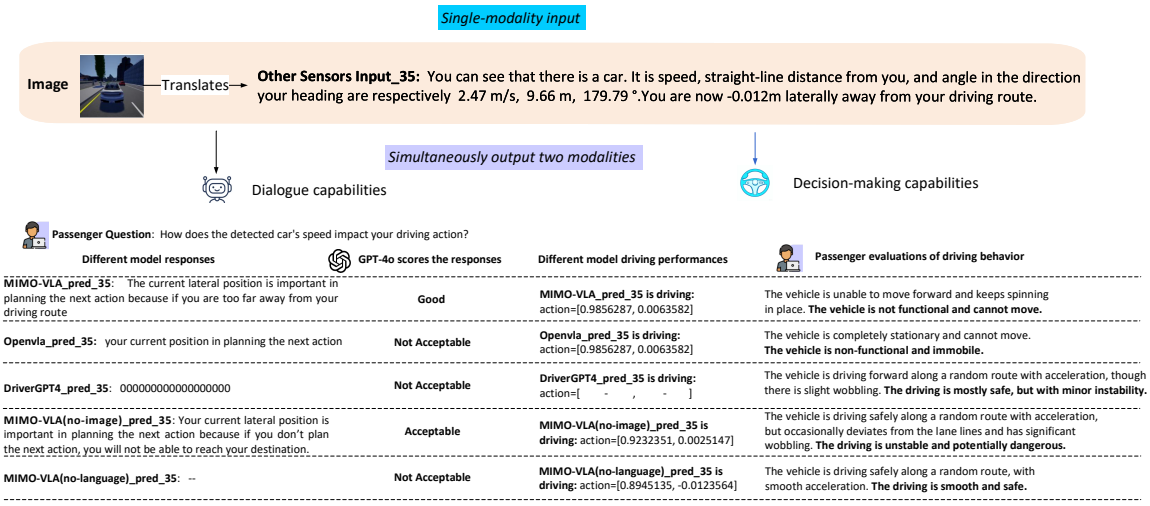


Figure 2: Random examples of different MIMO models engaging in fluent human–machine dialogue while simultaneously making real-time driving decisions.

encourages the formation of modality-specific representations at distinct positional encodings within the model.

4 Experiments

In this section, we validate MIMO-VLA on the CARLA autonomous driving simulation platform, demonstrating its ability to make fine-grained action decisions while maintaining dialogue capabilities. Our experiments focus on: (1) the impact of individual loss components on model performance, and (2) the influence of textual data quality on driving decisions.

4.1 Experimental setting

Our experiments were conducted in gym-carla (Chen, 2020), an OpenAI Gym-compatible environment built on CARLA 0.9.10. For LoRA fine-tuning, we selectively updated only the Q and V projection modules, comprising 0.06% of LLaMA-7B's parameters. Additional implementation details—including MIMO-VLA hyperparameters, linear mapping layers, and gym-carla configurations—are provided in Appendix A.1.

4.2 Comparison methods

Behavior Cloning (BC) implemented in gym-carla was used as a baseline. Comparative methods include the RL approaches Dreamer (Hafner et al., 2019) and Forbes (Chen et al., 2022), Decision Transformer (DT) (Chen et al., 2021), and VLA models OpenVLA (Kim et al., 2024) and DriverGPT4 (Xu et al., 2024).

4.3 Training datasets

The training dataset D_{expert} was collected using the EGADS framework (Tang et al., 2024), which designs RL- and IL-based agents with safety constraints and demonstrates strong performance in CARLA. We use this agent as our expert, driving vehicles in town03 of CARLA to collect the dataset. D_{expert} totals 5.69GB and contains 13,761 frames. For each frame, one question from a set of 50 was randomly selected based on the textual description of the current observation, along with its corresponding answer. The 50-question set is detailed in Appendix A.11, and further descriptions of D_{expert} and the map are provided in Appendix A.3 and Appendix A.2, respectively. Figure 3(b) illustrates the layout of town03 used for training.

During data collection and online evaluation, vehicles randomly select directions at intersections, follow randomly generated routes, slow down for preceding vehicles, and stop at red lights. Actions are defined as action = [accel, steering], with acceleration ranging from [-3, 3] ([0, 3] for acceleration, [-3, 0] for deceleration) and steering range in [-0.2, 0.2]. Additional implementation details are provided in Appendix A.1.

Following Chen et al. (2024), we design a template-based parser that converts sensor data (e.g., position and distance information, excluding vision and LiDAR) into natural language descriptions, shown as "other sensors input" in Figures 1 and 2. This input excludes any action-related information from MIMO-VLA, such as speed and heading, allowing us to evaluate whether the model

can leverage textual information to improve action decision quality. Details on the templates are provided in Appendix A.7.

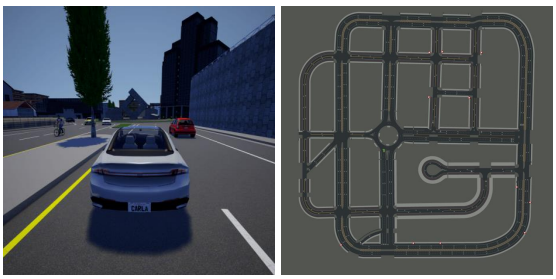
4.4 Performance metrics

Performance Metrics for Evaluating Chatting Ability The study by Wang et al. (2023) demonstrates that ChatGPT achieves high alignment with human judgments. Building on this finding, we employed GPT-4o (OpenAI, 2023a) to systematically compare the answer quality of MIMO-VLA with baseline models. Our evaluation procedure consisted of three steps: (1) selecting 50 random driving environment–question pairs; (2) generating responses from the baseline models for each pair; and (3) scoring the responses on a 0–10 scale using GPT-4o, with the following categories: Not Acceptable (< 3), Acceptable ($3 \leq$ score < 6), and Good (≥ 6). The full evaluation prompt is provided in Appendix A.7.

To further examine the contributions of the language and image components to dialogue capabilities, we evaluated two simplified variants of MIMO-VLA: MIMO-VLA (no-language) and MIMO-VLA (no-image). These variants were trained by removing the loss terms corresponding to text generation and image reconstruction, respectively, resulting in $\mathcal{L}$action + $\mathcal{L}$image and $\mathcal{L}$action + $\mathcal{L}$language.

Performance metrics for evaluating the decision-making ability We deployed our trained model on a vehicle for autonomous urban navigation and evaluated its performance using established metrics: Collision Rate (CR), Off-road Rate (OR), Episode Completion Rate (ER), Average Safe Driving Distance (ASD), Average Reward (AR), and Driving Score (DS). DS, a composite metric reflecting overall performance, is defined as DS = ER $\times$ AR, following the methodology of the CARLA Leaderboard. For AR, we adopted the reward function from Chen et al. (2019), which accounts for driving dynamics such as yaw, collisions, speeding, and lateral velocity. Model selection prioritized checkpoints that optimized both DS and AR. The remaining metrics (ER, OR, ASD) were implemented in accordance with Gao et al. (2024). Additional details on reward computation and metric calculations are provided in Appendices A.5 and A.6.

As shown in Figures 2 and 4, MIMO-VLA significantly outperforms the other models in terms of chatting ability. In contrast, OpenVLA performs poorly in question-answering, as it focuses solely on optimizing the action loss. DriverGPT4 also struggles: since both tasks share the same decoder, the model tends to interpret all inputs as intended for action prediction, making it difficult to generate complete text. Although DriverGPT4 has two independent loss terms, it fails to effectively balance them.



(a) (b)

Figure 3: (a) a sample view of the simulation environment; (b) a bird-eye view of our task scenario.

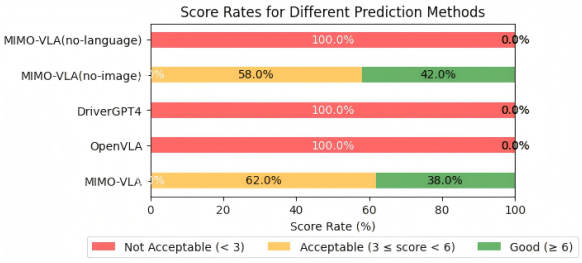


Figure 4: GPT-4o scores the answers from five methods for randomly generated inputs and question

Ablation results further highlight the role of different components. MIMO-VLA (no-language) exhibits a substantial drop in conversational performance compared to the full MIMO-VLA, whereas MIMO-VLA (no-image) performs comparably to MIMO-VLA. This indicates that the language loss component is crucial for enhancing chatting capabilities, while the image loss contributes less directly to dialogue performance.

4.5 Results on Decision-Making Ability Evaluation

In Tables 1–3, the "-" entries indicate system failure cases where no complete action value is generated within the 50-second threshold, representing the maximum allowable stationary wait time before triggering system intervention. Here, H denotes the length of the context.

As shown in Table 1, MIMO-VLA significantly outperforms BC and OpenVLA in DS, AR, and

Table 1: Evaluation results for different methods in town03 (random), *H=1*

Method	Input	DS ↑	AR (f) ↑	ASD(m) ↑	ER(%) ↑	OR(%) ↓	CR(%) ↓
BC	image	20.21 ± 7.46	175.34 ± 72.86	54.21 ± 6.41	9.08 ± 0.56	54.86 ± 20.04	60.00 ± 11.23
DriverGPT4	image, text	-	-	-	-	-	-
Openvla	image, text	-13.02 ± 4.02	-199.16 ± 38.73	24.34 ± 5.02	5.25 ± 0.39	24.36 ± 4.17	95.00 ± 0.00
MIMO-VLA	image, text	92.78 ± 23.75	466.80 ± 91.66	71.77 ± 9.40	16.35 ± 1.56	15.33 ± 4.36	55.00 ± 11.41

Table 2: Evaluation results for different methods in town03 (random), *H=4*

Method	Input	DS ↑	AR (f) ↑	ASD(m) ↑	ER(%) ↑	OR(%) ↓	CR(%) ↓
BC	image	36.39 ± 13.37	314.66 ± 86.02	64.08 ± 10.48	9.04 ± 0.62	37.56 ± 16.44	45.00 ± 11.41
Dreamer	image	-0.03 ± 0.01	-14.96 ± 0.09	0.02 ± 0.01	0.22 ± 0.01	0.00 ± 0.00	0.00 ± 0.00
Forbes	image	0.98 ± 1.43	21.63 ± 21.72	22.84 ± 1.00	6.30 ± 0.31	18.78 ± 1.03	56.67 ± 9.20
DT	image	7.68 ± 3.24	51.97 ± 29.33	23.74 ± 2.47	9.92 ± 0.71	10.31 ± 2.32	65.00 ± 10.94
DriverGPT4	image, text	-	-	-	-	-	-
Openvla	image, text	-7.84 ± 0.67	-160.37 ± 7.85	18.03 ± 1.92	4.76 ± 0.19	20.77 ± 3.36	100.00 ± 0.00
MIMO-VLA	image, text	105.25 ± 14.03	349.52 ± 49.75	59.76 ± 5.04	25.02 ± 2.57	19.93 ± 2.11	30.00 ± 10.51

Table 3: Evaluation the generalization for different methods in town04 (random), *H=4*

Method	Input	DS ↑	AR (f) ↑	ASD(m) ↑	ER(%) ↑	OR(%) ↓	CR(%) ↓
BC	image	39.22 ± 11.64	358.79 ± 79.59	63.08 ± 9.37	8.69 ± 0.56	5.64 ± 1.26	60.00 ± 11.23
Dreamer	image	-0.03 ± 0.01	-15.03 ± 0.07	0.02 ± 0.01	0.01 ± 0.21	0.01 ± 0.00	0.00 ± 0.00
Forbes	image	-2.63 ± 2.75	-17.37 ± 22.98	19.79 ± 1.20	6.24 ± 0.69	15.80 ± 2.74	66.70 ± 8.75
DT	image	10.66 ± 3.26	85.58 ± 27.04	24.94 ± 2.92	10.55 ± 0.58	11.38 ± 2.15	55.00 ± 11.41
DriverGPT4	image, text	-	-	-	-	-	-
Openvla	image, text	-6.74 ± 0.88	-153.35 ± 10.26	13.62 ± 1.86	4.26 ± 0.17	15.70 ± 2.71	100.00 ± 0.00
MIMO-VLA	image, text	94.26 ± 15.26	384.52 ± 51.72	56.93 ± 4.03	21.49 ± 1.86	12.75 ± 2.28	45.00 ± 11.41

ASD at a single time step, while DriverGPT4 fails to produce precise action values. Across multiple time steps (Table 2), MIMO-VLA continues to show substantial improvements over other methods, indicating sustained performance over longer durations.

To evaluate generalization, models were trained on the town03 dataset and tested online in town04. Table 3 demonstrates that MIMO-VLA achieves a markedly higher DS than the other methods, highlighting its strong generalization capability. By contrast, DriverGPT4 struggles to generate precise action values for real-time control commands, reflecting the challenges of producing accurate outputs using a detokenizer. OpenVLA can generate precise values in experiments but often produces identical action commands, causing the vehicle to spin or wander in place, incurring significant penalties.

Additionally, Tables 2 and 3 show that MIMO-VLA significantly outperforms DT, Dreamer, and Forbes in both decision-making performance and generalization. Finally, Figure 2 illustrates that MIMO-VLA can seamlessly engage in conversation with a human while simultaneously making real-time driving decisions.

4.6 Ablation Studies on Loss Function Design

As shown in Equation (6), our loss function consists of three components: action loss $\mathcal{L}$action, language loss $\mathcal{L}$language, and image loss $\mathcal{L}_{\text{image}}$. We conducted ablation studies to investigate the contribution of each loss to MIMO-VLA's performance. The results are presented in Table 4.

For comparison, we include the action-bins loss $\mathcal{L}$action-bins used by OpenVLA and RT2, which handle continuous-valued actions via value discretization. We also evaluate two simplified variants of MIMO-VLA: MIMO-VLA (no-language) and MIMO-VLA (no-image). These were trained using $\mathcal{L}$action + $\mathcal{L}$image and $\mathcal{L}$action + $\mathcal{L}_{\text{language}}$, respectively, to isolate the effects of the language and image loss components.

On the effect of $\mathcal{L}_{\text{action}}$ As shown in Table 4, comparing the performance of $\mathcal{L}$image + $\mathcal{L}$language + $\mathcal{L}$action-bins with $\mathcal{L}$image + $\mathcal{L}$language + $\mathcal{L}$action clearly demonstrates the advantage of our action loss $\mathcal{L}$action. This explains why MIMO-VLA outperforms VLA models using action losses similar to $\mathcal{L}$action-bins, as observed in Tables 1–3.

Our experiments indicate that action discretization and tokenization, as used in current VLA models, often yield low training loss but poor inference

Table 4: Ablation Study on MIMO-VLA Loss Functions in Town03 (Random), with $H = 4$

Loss function	Input	DS ↑	AR (f) ↑	ASD(m) ↑	ER(%) ↑	OR(%) ↓	CR(%) ↓
$\mathcal{L}_{\text{image}}$+$\mathcal{L}_{\text{language}}$+$\mathcal{L}_{\text{action-bins}}$	image, text	11.57 ± 0.00	142.83 ± 0.01	22.71 ± 0.01	8.10 ± 0.05	30.87 ± 0.10	100.00 ± 0.00
$\mathcal{L}_{\text{image}}$+$\mathcal{L}_{\text{action}}$	$image, text$	45.08 ± 10.88	234.36 ± 52.21	39.64 ± 4.03	14.13 ± 1.71	16.68 ± 3.15	30.00 ± 10.51
$\mathcal{L}_{\text{language}}$+$\mathcal{L}_{\text{action}}$	image, text	74.85 ± 10.97	331.78 ± 49.88	50.63 ± 4.73	18.62 ± 1.95	15.96 ± 2.45	25.00 ± 9.93
$\mathcal{L}_{\text{image}}$+$\mathcal{L}_{\text{language}}$+$\mathcal{L}_{\text{action}}(our)$	image, text	105.25 ± 14.03	349.52 ± 49.75	59.76 ± 5.04	25.02 ± 2.57	19.93 ± 2.11	30.00 ± 10.51

Table 5: Impact of Sensor Input and QA Content Noise on MIMO-VLA's Decision-Making Performance in Town03 (Random)

Input	Sensor input / QA (noise ratio)	DS ↑	AR (f) ↑	ASD(m) ↑	ER(%) ↑	OR(%) ↓	CR(%) ↓
image, text	0%/100%	74.32 ± 24.44	288.54 ± 74.62	62.42 ± 7.71	25.76 ± 1.54	11.05 ± 1.62	50.0 ± 0.51
image, text	0%/0%	93.89 ± 29.73	336.11 ± 86.72	45.42 ± 9.53	16.68 ± 2.50	19.05 ± 4.96	5.00 ± 5.00
image, text	100%/0%	-0.01 ± 1.12	-5.10 ± 0.00	0.00 ± 0.00	0.30 ± 0.00	0.00 ± 0.00	0.00 ± 0.00

performance. This occurs because adjacent action intervals are represented by consecutive token IDs that are close in token space, causing the model to repeatedly output the same token during inference, even though the corresponding action values may differ substantially. For example, in OpenVLA, discretizing throttle control (0–100%) into 256 bins produces intervals of 0.39%, allowing only fixed values such as 0%, 0.39%, 0.78%, etc. Token IDs like 31830 and 31831 may be treated by the model as equivalent, preventing fine-grained distinctions.

In contrast, our approach uses an MLP to map actions to continuous control values, enabling direct outputs with arbitrary precision.

On the effect of $\mathcal{L}_{\text{language}}$ As shown in Table 4, comparing $\mathcal{L}$image + $\mathcal{L}$action (MIMO-VLA (no-language)) with $\mathcal{L}$image + $\mathcal{L}$language + $\mathcal{L}$action (MIMO-VLA) reveals that including $\mathcal{L}$language in the loss function significantly improves decision-making quality. Figures 2 and 4 further show that MIMO-VLA (no-language) exhibits markedly weaker dialogue capabilities than MIMO-VLA, whereas MIMO-VLA (no-image) performs comparably to MIMO-VLA. These results highlight that $\mathcal{L}$language is crucial for maintaining strong dialogue performance. In summary, incorporating $\mathcal{L}$language benefits both dialogue and decision-making capabilities.

On the effect of $\mathcal{L}_{\text{image}}$ As shown in Table 4, adding $\mathcal{L}$image to the loss function (corresponding to $\mathcal{L}$image + $\mathcal{L}$language + $\mathcal{L}$action) improves all decision-making performance metrics compared to $\mathcal{L}$language + $\mathcal{L}$action. This demonstrates that $\mathcal{L}_{\text{image}}$ provides significant benefits for enhancing decision-making. We attribute this improvement to the fact that high-quality image reconstruction allows the model to better extract and leverage rich scene information from the image modality, thereby supporting more informed decision-making.

4.7 Resolving Conflicts in Simultaneous Multi-Task Output

In DriveGPT4, the text generation and action generation tasks lack independently designed loss functions, leading to conflicts between them. This is especially problematic in complex environments, where the model struggles to generate both effective text and precise action instructions simultaneously. As shown in Figure 2, DriveGPT4 cannot reliably produce fine-grained action commands at every moment, which compromises both decision accuracy and dialogue quality. In contrast, MIMO-VLA employs separate objective functions for text and action generation, enabling efficient parallel processing of each task and effectively preventing conflicts between them.

4.8 Impact of Textual Data Quality on Decision-Making Capability

Consider a driver conversing with a passenger while operating a car. If the passenger's remarks are irrelevant to the driving context, they may interfere with the driver's decisions, although humans naturally exhibit some tolerance to such noise. To investigate whether our model demonstrates similar human-like decision-making behavior, we designed a set of experiments.

As shown in Table 5, introducing noise unrelated to driving into the sensor inputs causes a rapid decline in the model's decision-making performance. In contrast, adding noise solely to the QA content while keeping sensor inputs clean results in a less pronounced performance drop. This suggests that the model remains robust as long as the sensor inputs are relevant to the driving scenario, even when the QA content contains noise. These findings indicate that our model's decision-making behavior

closely mirrors that of human drivers.

5 Conclusion

In this paper, we investigated how to develop a multimodal pre-training framework to address the inherent task conflicts in MISO architectures under MIMO scenarios, where multiple tasks share output channels. Such conflicts often lead to imbalanced optimization and significant performance degradation in individual tasks. To tackle this, we proposed a unified MIMO training architecture with parallel multi-task output capabilities—MIMO-VLA. Experiments demonstrate that MIMO-VLA outperforms SOTA VLA models, RL approaches, and decision transformers in decision-making while maintaining fluent dialogue, thanks to our continuous-action handling, loss function design, and label smoothing techniques.

The shift from modular systems composed of discrete subcomponents to unified end-to-end models represents a major trend in AI research. In the context of MIMO scenarios, we believe this work constitutes a meaningful first step toward developing a unified generative model capable of simultaneously handling both dialogue and action generation in an end-to-end manner.

Limitations

This study has several limitations that guide future research. First, although MIMO-VLA is a general-purpose unified MIMO framework, our validation is limited to autonomous driving, leaving its generalizability to other domains, such as robotics or human-computer interaction, untested. Second, joint optimization of dialogue and action generation still requires improvement, particularly in multi-task coordination and scalability. Our dialogue evaluation also lacks scenario categorization and linguistic diversity, which future studies could address to enable more detailed assessment of conversational capabilities. Finally, long text prompts or large image inputs can create synchronization delays, underscoring the need for more efficient token processing and resource allocation to support real-time performance.

References

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, and 1 others. 2022. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, and 1 others. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, and 1 others. 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Thomas Carta, Clément Romac, Thomas Wolf, Sylvain Lamprier, Olivier Sigaud, and Pierre-Yves Oudeyer. 2023. Grounding large language models in interactive environments with online reinforcement learning. In *International Conference on Machine Learning*, pages 3676–3713. PMLR.

Jianyu Chen. 2020. An openai gym third party environment for carla simulator. https://github.com/cjy1992/gym-carla?tab=readme-ov-file.

Jianyu Chen, Bodi Yuan, and Masayoshi Tomizuka. 2019. Model-free deep reinforcement learning for urban autonomous driving. In *2019 IEEE intelligent transportation systems conference (ITSC)*, pages 2765–2771. IEEE.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. 2021. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097.

Long Chen, Oleg Sinavski, Jan Hünermann, Alice Karnsund, Andrew James Willmott, Danny Birch, Daniel Maund, and Jamie Shotton. 2024. Driving with llms: Fusing object-level vector modality for explainable autonomous driving. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 14093–14100. IEEE.

Xiaoyu Chen, Yao Mark Mu, Ping Luo, Shengbo Li, and Jianyu Chen. 2022. Flow-based recurrent belief state

learning for pomdps. In *International Conference on Machine Learning*, pages 3444–3468. PMLR.

Nicolai Dorka, Chenguang Huang, Tim Welschehold, and Wolfram Burgard. What matters in employing vision language models for tokenizing actions in robot control? In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*.

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

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

A. S. et al. 2024. Introducing rfm-1: Giving robots human-like reasoning capabilitie. Introducingrfm-1: Givingrobotshuman-likereasoningcapabilitie.

Justin Fu, Kelvin Zhang, Utkarsh Sanyal, Lantao Yu, Collin Moses, Fan Yang, Stefano Ermon, and Zhibin Zhao. 2023. Driving with reasoning: Reinforcement learning with generalist language models for interpretable policies. *arXiv preprint arXiv:2303.00745*.

Zeyu Gao, Yao Mu, Chen Chen, Jingliang Duan, Ping Luo, Yanfeng Lu, and Shengbo Eben Li. 2024. Enhance sample efficiency and robustness of end-to-end urban autonomous driving via semantic masked world model. *IEEE Transactions on Intelligent Transportation Systems*.

Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2022. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM Conference on Recommender Systems*, pages 299–315.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. 2019. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*.

Bin Hu, Chenyang Zhao, Pu Zhang, Zihao Zhou, Yuanhang Yang, Zenglin Xu, and Bin Liu. 2024. Enabling intelligent interactions between an agent and an llm: A reinforcement learning approach. *Reinforcement Learning Conference (RLC)*.

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

Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. 2023. An embodied generalist agent in 3d world. *arXiv preprint arXiv:2311.12871*.

Chia-Chun Hung, Timothy Lillicrap, Josh Abramson, Yan Wu, Mehdi Mirza, Federico Carnevale, Arun Ahuja, and Greg Wayne. 2019. Optimizing agent behavior over long time scales by transporting value. *Nature communications*, 10(1):5223.

Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. 2022. Vima: General robot manipulation with multimodal prompts. *arXiv preprint arXiv:2210.03094*, 2(3):6.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, and 1 others. 2024. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR.

Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, and 1 others. 2023b. Vision-language foundation models as effective robot imitators. *arXiv preprint arXiv:2311.01378*.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023c. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Junling Liu, Chao Liu, Peilin Zhou, Renjie Lv, Kang Zhou, and Yan Zhang. 2023. Is chatgpt a good recommender? a preliminary study. *arXiv preprint arXiv:2304.10149*.

OpenAI. 2023a. Gpt-4 technical report.

OpenAI. 2023b. Gpt-4: Technical report. https://www.openai.com/research/gpt-4.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Abhishek Padalkar, Acorn Pooley, Ajinkya Jain, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anikait Singh, Anthony Brohan, and 1 others. 2023. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*.

Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, and 1 others. 2022. A generalist agent. *arXiv preprint arXiv:2205.06175*.

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

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

Chao Song, Zhihao Ye, Qiqiang Lin, Qiuying Peng, and Jun Wang. 2024. A framework to implement 1+ n multi-task fine-tuning pattern in llms using the cgc-lora algorithm. *arXiv preprint arXiv:2402.01684*.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*.

Zuojin Tang, Xiaoyu Chen, YongQiang Li, and Jianyu Chen. 2024. Efficient and generalized end-to-end autonomous driving system with latent deep reinforcement learning and demonstrations. *arXiv preprint arXiv:2401.11792*.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, and 1 others. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faysal Azhar, and 1 others. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Pierre-Emmanuel Albert, Amjad Almahairi, Yasmine Babaei, Dmytro Bashlykov, Subhojit Batra, Anurag Bhargava, Shruti Bhosale, and 1 others. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jiaan Wang, Yunlong Liang, Fandong Meng, Zengkui Sun, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. 2023. Is chatgpt a good nlg evaluator? a preliminary study. *arXiv preprint arXiv:2303.04048*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Yi Xiao, Felipe Codevilla, Akhil Gurram, Onay Urfalioglu, and Antonio M López. 2020. Multimodal end-to-end autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 23(1):537–547.

Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li, and Hengshuang Zhao. 2024. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *IEEE Robotics and Automation Letters*.

Shunyu Yao, Jeffrey Wu, Daisy Zhe Liu, Dale Schuurmans, Quoc V Le, Denny Zhou, Yuan Cao, and Andrew Dai. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 2024. 3d-vla: A 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*.

Zihao Zhou, Bin Hu, Pu Zhang, Chenyang Zhao, and Bin Liu. 2024. Large language model as a policy teacher for training reinforcement learning agents. *International Joint Conference on Artificial Intelligence (IJCAI)*.

A Appendix

A.1 Hyperparameter settings

In this section, we present the model parameters of MIMO-VLA, the parameters of the custom linear layers, and the settings for Gym-CARLA and evaluation, as summarized in Tables 6, 7, and 8. The models were trained using Python 3.8, Transformers 4.30.0, and an NVIDIA Tesla V100 GPU. Training times ranged from 5 to 13 hours, depending on the input modality and trajectory length. We also conducted experiments on the three loss function hyperparameters, as detailed in Appendix A.12, and in our experiments we set $\alpha_1 = 0.1$, $\alpha_2 = 10$, and $\lambda = 0.5$.

A.2 CARLA maps

To comprehensively evaluate the performance of MIMO-VLA, we utilized five maps in CARLA, including town03 and town04, as shown in Figure 6. Town03 is a complex urban map closely resembling real-world road environments, featuring tunnels, intersections, roundabouts, curves, and multi-turns. It covers an area of 400m $\times$ 400m, with a total road length of approximately 6km. In contrast, Town04 is a smaller town set against a backdrop of snow-capped mountains and conifers, with a multi-lane road forming a "figure of 8" around the town.

A.3 Training datasets

We trained all comparison methods using an expert dataset D_{expert}, which is 5.69GB in size and contains 13,761 frames. Of this, 90% was used for training and the remaining 10% for testing. The comparison methods were evaluated online in CARLA town03 under random mode. Following DT (Chen et al., 2021), we assessed sequence fusion performance for both single and multiple time steps. We set the context length $H = 1$, resulting in a fusion sequence length of 489 tokens, which includes 64 tokens for the 128$\times$128 image and padded text sequences of length 424 (including an empty placeholder token).

Due to computational constraints, we explored trajectory sequences with a maximum length of $489 \times 4 = 1956$ to assess performance over longer contexts. Further analysis of MIMO-VLA's decision-making ability with longer trajectory contexts is provided in Appendix A.4. We also evaluated performance across different modalities and generalization capabilities in town04. Detailed information on the CARLA maps can be found in Appendix A.2.

All comparison methods were tested online in the CARLA simulator over 20 episodes, each consisting of 1,000 steps and 200 vehicles, with driving routes and traffic scenarios generated in random mode.

A.4 Effect of Longer Trajectory Context on Decision-Making Ability

As shown in Table 9, increasing the context length H of input trajectories provides some improvement in MIMO-VLA's overall DS and AR, but the gains are modest. The improvement mainly stems from higher route completion rates and lower collision rates associated with longer time steps. For example, when $H = 4$, the sequence length extends to 1,956 tokens—a fourfold increase—yet the improvements in DS and AR remain limited. In fact, for metrics such as AR and ASD, the performance for $H = 4$ can even be worse than that for $H = 1$, suggesting that excessively long trajectories may introduce redundant information that can negatively impact decision-making.

This result highlights several key points. While longer contexts provide the model with more historical information, too much data can hinder its ability to filter and extract useful decision signals, leading to redundancy. Redundant information not only increases computational complexity but can also distract the model, reducing its capacity to capture critical features and negatively affecting overall decision-making. Consequently, shorter context sequences offer more concise and precise inputs, enabling quicker and more accurate judgments. These findings suggest that the current sequence fusion method has limited benefits from longer contexts. Prior work (Chen et al., 2021; Hung et al., 2019) indicates that longer contexts can improve decision control, motivating future research on compressing historical information and efficiently fusing it to enhance decision-making.

A.5 Reward function

We use the default reward function of the Gym-Carla benchmark (Chen et al., 2019) to evaluate all experimental methods, as follows:

$$f = 200r_c + v_{lon} + 10r_f + r_o \\ -5\alpha^2 + 0.2r_{lat} - 0.1 \tag{7}$$

where r_c is the collision-related reward, set to -1 if the ego vehicle collides and 0 otherwise. v_{lon}

Table 6: Hyperparameters

Parameter	Value
batch_size	64
micro_batch_size	8
num_epochs	3
learning_rate	3e-4
cutoff_len	424
val_set_size	0.1
save_step	25
lora_r	8
lora_alpha	16
lora_dropout	0.05
lora_target_modules	{q_proj, k_proj}
Other Sensors Input_types	{obs, text}
lambda_action	10
lambda_smooth	0.1
lambda_img	0.5
horizon	1
regular_action_loss	False
img_patch_size	16

Table 7: Model Parameters and Layers

Parameter/Layer	Details
num_patches	64
tokenizer_vocab_size	32000
split_obs_proj	Conv2d(3, 4096, kernel_size=16, stride=16)
inverse_split_obs_proj	ConvTranspose2d(4096, 3, kernel_size=16, stride=16)
split_obs_position_embedding	Parameter(torch.randn(1, 64, 4096))
text_embedding	nn.Embedding(32000, 4096)
custom_lm_head	Linear(4096, 32000, bias=False)
actor_linear1	Linear(4096, 2048)
actor_linear2	Linear(2048, 1024)
actor_linear3	Linear(1024, 512)
actor_linear4	Linear(512, 256)
actor_linear5	Linear(256, 128)
actor_linear6	Linear(128, 64)
actor_linear7	Linear(64, 2)
reconstruction_layer	Linear(4096, micro_batch_size*3*128*128)
action_linear	Linear(2, 4096)

denotes the longitudinal speed of the ego vehicle. r_f is the reward for exceeding the desired speed ($8m/s$ in this case), set to -1 if the speed limit is exceeded and 0 otherwise. r_o is set to -1 if the ego vehicle leaves its lane, and 0 otherwise. α represents the steering angle of the ego vehicle in radians. The lateral acceleration reward r_{lat} is computed as $r_{lat} = -|\alpha| \cdot v_{lon}^2$. Finally, a constant term is added to prevent the ego vehicle from remaining stationary.

A.6 Performance metrics of autonomous driving models

We use several key metrics to evaluate the performance of autonomous driving models across various driving scenarios:

Table 8: gym-carla and evaluation Environment Parameters

Parameter	**Value**
Number of Vehicles	200
Number of Walkers	0
Random Seed	1
Other Sensors Input_names	lidar_noground
Display Size	400
Max Past Step	1
Time Step (dt)	0.1
Discrete Control	False
Continuous Acceleration Range	[-3.0, 3.0]
Continuous Steering Range	[-0.2, 0.2]
Ego Vehicle Filter	vehicle.lincoln*
Traffic Manager Port	Random integer (2000 to 9000)
Town Map	town03 or town04
Task Mode	Random
Max Time per Episode	2000
Max Waypoints	12
Observation Range	32
LiDAR Bin Size	0.25
Distance Behind Ego Vehicle	12
Lane Threshold	2.0
Desired Speed	8
Max Ego Vehicle Spawn Times	200
Display Route	True
PIXOR Grid Size	64
PIXOR Mode	False
Predict Speed	True

Table 9: Evaluation of MIMO-VLA with Longer Contexts for Multimodal Input in Town03 (Random)

Input	$\mathcal{L}_{\text{image}}$	H	DS ↑	AR (f) ↑	ASD(m) ↑	ER(%) ↑	OR(%) ↓	CR(%) ↓
image	×	1	29.55 ± 6.17	226.91 ± 42.24	54.24 ± 4.30	11.85 ± 0.68	20.22 ± 5.57	70.00 ± 10.5
image	×	4	22.38 ± 4.96	155.79 ± 31.87	32.45 ± 1.74	14.41 ± 0.59	15.93 ± 2.65	40.00 ± 11.23
text	×	1	37.44 ± 10.11	248.89 ± 52.91	47.37 ± 5.43	15.63 ± 1.98	17.02 ± 2.71	40.00 ± 11.24
text	×	4	44.16 ± 7.39	252.10 ± 38.94	46.96 ± 3.23	15.66 ± 1.06	12.86 ± 2.45	60.00 ± 11.23
image, text	×	1	68.10 ± 13.20	417.24 ± 57.41	58.81 ± 6.55	13.71 ± 1.26	11.39 ± 2.41	40.00 ± 11.24
image, text	×	4	74.85 ± 10.97	331.78 ± 49.88	50.63 ± 4.73	18.62 ± 1.95	15.96 ± 2.45	25.00 ± 9.93
image, text	✓	1	92.78 ± 23.75	466.80 ± 91.66	71.77 ± 9.40	16.35 ± 1.56	15.33 ± 4.36	55.00 ± 11.41
image, text	✓	4	105.25 ± 14.03	349.52 ± 49.75	59.76 ± 5.04	25.02 ± 2.57	19.93 ± 2.11	30.00 ± 10.51

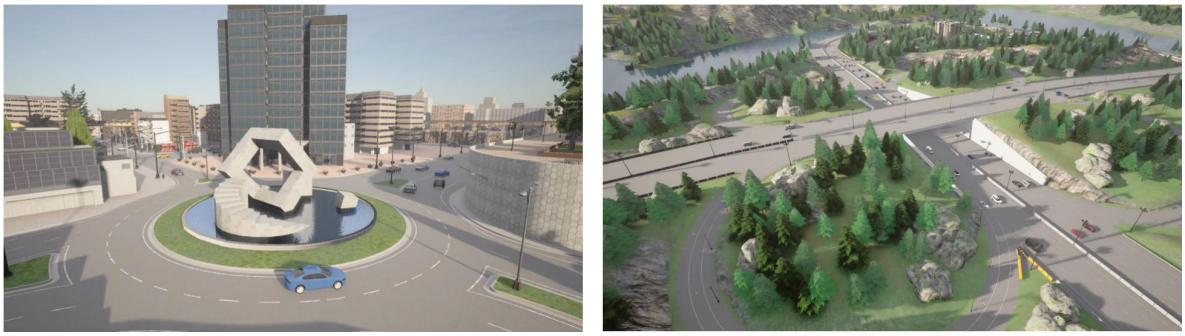
- Collision Rate (CR): The frequency at which the vehicle collides with obstacles or other vehicles, critical for assessing driving safety.

- Outlane Rate (OR): The rate at which the vehicle deviates from its designated lane, reflecting the model's lane-keeping ability.

- Episode Completion Rate (ER): The percentage of driving tasks or episodes successfully completed, with higher values indicating better task performance.

- Average Safe Driving Distance (ASD): The average distance driven without incidents, such as collisions or off-road events, highlighting safe driving capability over extended periods.

- Average Return (AR): The cumulative reward collected during driving tasks, reflecting both task performance and adherence to safety guidelines.



(a) Town03 (b) Town04

Figure 5: CARLA maps

- Driving Score (DS): A comprehensive metric capturing overall performance in terms of safety, efficiency, and traffic compliance, following the CARLA Leaderboard methodology.

For AR, we adopt the reward function f from Chen et al. (2019), which evaluates driving dynamics including yaw, collisions, speeding, and lateral velocity. Model selection prioritizes checkpoints optimizing both DS and AR. The remaining metrics (ER, OR, ASD) are implemented following Gao et al. (2024).

$$CR = \frac{N_{\text{collisions}}}{N_{\text{total\_episodes}}}, OR = \frac{N_{\text{off\_road\_events}}}{N_{\text{total\_episodes}}} \quad (8)$$

$$ER = \frac{N_{\text{completed\_steps}}}{N_{\text{total\_steps}}}, ASD = \frac{\sum_{i=1}^{N_{\text{episodes}}} \text{distance}_i}{N_{\text{total\_episodes}}} \quad (9)$$

$$AR = \frac{\sum_{i=1}^{N_{\text{episodes}}} \text{rewards}_i}{N_{\text{total\_episodes}}}, DS = ER \times AR \quad (10)$$

where $N_{\text{collisions}}$ denotes the number of collisions during the episode, and $N_{\text{total\_episodes}}$ is the total number of episodes in the test. $N_{\text{off\_road\_events}}$ represents the number of times the vehicle went off-road, while $N_{\text{total\_steps}}$ is the total number of steps across all episodes. distance_i is the distance driven during the i-th safe driving episode, and $N_{\text{safe\_episodes}}$ is the number of episodes completed without incidents, such as collisions or off-road events. $N_{\text{completed\_steps}}$ is the number of successfully completed steps, and $N_{\text{total\_steps}}$ is the total number of steps in the episode. Finally,AR denotes the average reward f collected during the episode.

A.7 The natural language template for text input

We collected information from the CARLA environment using additional sensors (e.g., speed and position sensors), excluding the ego vehicle's acceleration and steering actions. This information is then converted into a natural language template interpretable by the VLA, as illustrated below:
<lateral_dis, delta_yaw, speed, vehicles_info> = <observation_vehicle_state>
<vehicles_num> = <len(vehicles_info)>
<multi_dis += str(vehicles_info[i][0])+"", multi_yaw += str(vehicles_info[i][1])+"", multi_speed += str(vehicles_info[i][2])+"">
*<**if vehicles_num**=1:>*
*<**new_input**="You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively **{multi_speed}** m/s, **{multi_dis}** m, **{multi_yaw}**°." "You are now **{lateral_dis}**m laterally away from your driving route. ">*
*<**elif vehicles_num**>1:>*
*<**new_input**="You can see that there are **vehicles_num** cars. Their speed, straight-line distance from you, and angle in the direction you're heading are respectively **{multi_speed}** m/s, **{multi_dis}** m, **{multi_yaw}**°." "You are now **{lateral_dis}**m laterally away from your driving route. ">*
*<**elif vehicles_num**=0:>*
*<**new_input**="You see no car here, and you are now **{lateral_dis}**m laterally away from your driving route.">*

We followed Wang et al. (2023) *"Is ChatGPT a Good NLG Evaluator?"* approach. The complete evaluation prompt template for using GPT-4o (OpenAI, 2023b) is as follows:"*The document contains 50 similar examples as described above. For each example, based on the given Input_0: and Question_0:, please evaluate and score the responses generated by the five methods (MIMO-VLA_pred_0, Openvla_pred_0,*

DriverGPT4_pred_0, MIMO-VLA_image_pred_0, and MIMO-VLA_language_pred_0) using a 10-point scale with the following criteria:Not Acceptable (< 3), Acceptable ($3 \leq score < 6$), and Good (≥ 6). Please output the individual scores for each example. After evaluating all 50 examples, calculate the average rates for: Not Acceptable, Acceptable, Good, and Excellent performance for each method."

A.8 The benefits of cross-entropy loss and label smoothing loss for MIMO-VLA

We observed that simply replacing specific numerical values in the translation template (Chen et al., 2024) produces minimal representational differences due to the sequential nature of the data. This makes conventional cross-entropy loss prone to overfitting in text generation tasks. As shown in Table 10, evaluations on both town03 and town04 revealed a decline in the model's decision-making performance when using standard cross-entropy loss. In contrast, cross-entropy loss with smoothed labels yielded better performance. Consequently, we adopted cross-entropy loss with smoothed labels for text generation in MIMO-VLA in our experiments.

A.9 Impact of Training Data on Model Decision Performance

In the multimodal ablation experiments on MIMO-VLA, summarized in Table 11, we systematically removed or replaced individual modalities to evaluate their contributions to decision-making. The results show that models using both image and text inputs significantly outperform those using only a single modality in terms of decision accuracy and stability. This indicates that the text modality provides higher-level semantic information that complements visual inputs, thereby enhancing overall decision-making.

Furthermore, as shown in Table 11, models with text-only input outperform those with image-only input. This highlights that the text modality in our dataset—particularly the "other sensors input" shown in Figure 6—offers highly informative cues that substantially improve the model's decision-making capability.

A.10 The noise datasets

The noise consisted of information entirely unrelated to the current driving scenario, as follows: {"A playful puppy brings joy and laughter to our days", "The whisper of the wind carries secrets of the universe", "A hidden garden blooms with the magic of nature's colors", "The aroma of fresh coffee awakens the senses each morning", "A handwritten letter feels like a warm hug from afar", "The glimmer of fireflies creates a magical summer night", "A spontaneous adventure can lead to unforgettable memories", "The serenity of a quiet lake reflects the beauty of the world", "A gentle touch can convey love without a single word", "The laughter of friends is the sweetest melody of all", "A warm hug is a universal language of comfort", "The dance of leaves in the breeze tells stories of change", "A cozy fire invites stories and shared moments", "The beauty of art inspires creativity and self-expression", "A day spent volunteering fills the heart with purpose", "The excitement of a new book is like embarking on a journey", "A delicious meal shared brings people closer together", "The sound of laughter can brighten even the gloomiest day", "A fleeting moment can hold the weight of a thousand memories", "The charm of small towns lies in their simple beauty", "A gentle rain nurtures the earth and inspires growth", "A colorful painting captures the essence of joy", "The peace of a mountain retreat refreshes the soul", "A favorite mug holds warmth and comfort on a chilly day", "The rustle of leaves underfoot reminds us of nature's rhythm", "A well-crafted story has the power to transport us anywhere", "The thrill of discovery keeps our spirits young and curious", "A cherished photograph holds a lifetime of memories", "The beauty of winter blankets the world in quiet calm", "A moment of kindness can change the trajectory of a day", "The aroma of spices fills the kitchen with warmth and love", "A shared joke creates bonds that laughter alone cannot", "The glow of a sunrise fills the heart with hope", "A melody can linger in the mind long after it fades", "The colors of autumn leaves create a vibrant tapestry", "A soft pillow cradles the head and invites sweet dreams", "The laughter of children brings joy and light to our lives", "A surprise visit from a friend can brighten any day", "The beauty of a flower garden is a celebration of life", "A good book can be a loyal companion on lonely nights", "The embrace of nature can heal and rejuvenate the spirit", "A treasure hunt ignites the spirit of adventure", "The warmth of homemade cookies fills the home with love", "A playful kitten brings joy and mischief to our lives", "The scent of pine trees evokes

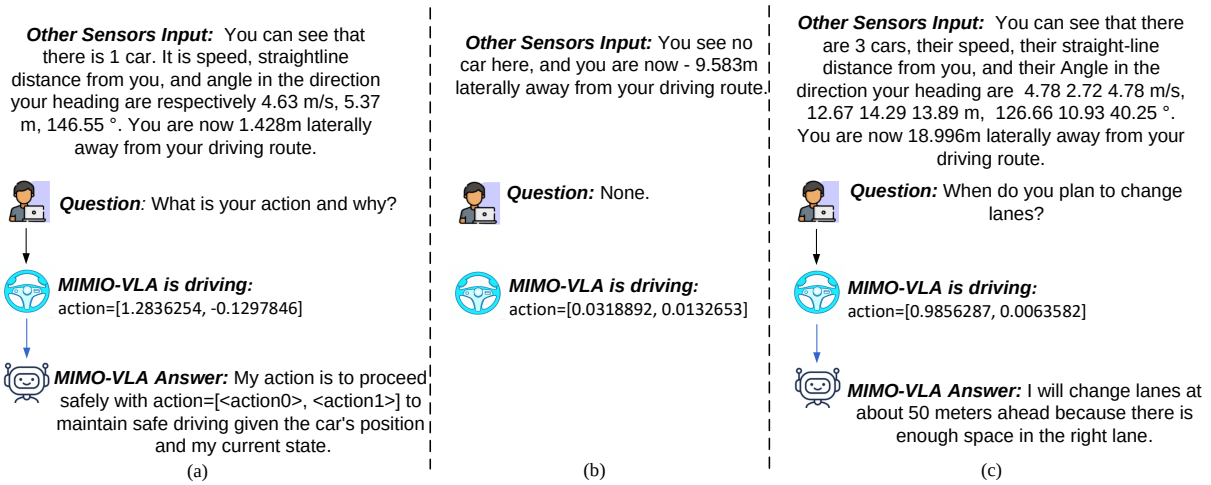


Figure 6: An example illustrating how MIMO-VLA seamlessly engages in conversation with a human while simultaneously making real-time driving decisions.

Table 10: Evaluation of MIMO-VLA Performance Using Smoothed Label Loss and Cross-Entropy Loss (*H=4*)

$\mathcal{L}_{\text{language}}$	Town	DS ↑	AR (f) ↑	ASD(m) ↑	ER(%) ↑	OR(%) ↓	CR(%) ↓
Cross Entropy	town03	48.97 ± 7.60	296.53 ± 40.72	47.10 ± 4.87	15.37 ± 0.85	12.41 ± 2.73	35.00 ± 10.94
Smooth Label	town03	105.25 ± 14.03	349.52 ± 49.75	59.76 ± 5.04	25.02 ± 2.57	19.93 ± 2.11	30.00 ± 10.51
Cross Entropy	town04	66.69 ± 16.97	358.11 ± 61.10	52.72 ± 5.44	15.43 ± 1.11	9.63 ± 1.42	55.00 ± 11.41
Smooth Label	town04	94.26 ± 15.26	384.52 ± 51.72	56.93 ± 4.03	21.49 ± 1.86	12.75 ± 2.28	45.00 ± 11.41

Table 11: Impact of Different Modal Inputs on MIMO-VLA Decision-Making in Town03 (Random) with *H=4*

Input	$\mathcal{L}_{\text{image}}$	DS ↑	AR (f) ↑	ASD(m) ↑	ER(%) ↑	OR(%) ↓	CR(%) ↓
image	×	22.38 ± 4.96	155.79 ± 31.87	32.45 ± 1.74	14.41 ± 0.59	15.93 ± 2.65	40.00 ± 11.23
text	×	44.16 ± 7.39	252.10 ± 38.94	46.96 ± 3.23	15.66 ± 1.06	12.86 ± 2.45	60.00 ± 11.23
image, text	×	74.85 ± 10.97	331.78 ± 49.88	50.63 ± 4.73	18.62 ± 1.95	15.96 ± 2.45	25.00 ± 9.93
image, text	✓	105.25 ± 14.03	349.52 ± 49.75	59.76 ± 5.04	25.02 ± 2.57	19.93 ± 2.11	30.00 ± 10.51

memories of the forest",]}. Additionally, we generated a dataset of over 500 question-answer pairs using GPT to introduce noise into the QA content. A random subset is shown below: { "question": "What is the national flower of Egypt?", "answer": "The national flower of Egypt is the lotus." }, { "question": "How do I make a vegetable soup?", "answer": "To make a vegetable soup, chop vegetables, simmer them in broth, and season with herbs and spices." }, { "question": "What is the largest gulf in the world?", "answer": "The largest gulf in the world is the Gulf of Mexico." }, { "question": "How do I clean my humidifier?", "answer": "To clean your humidifier, empty it, wash the tank with soap and water, and rinse thoroughly." }, { "question": "What is the national animal of Egypt?", "answer": "The national animal of Egypt is the steppe eagle." }, { "question": "How do I make a cup of lavender tea?", "answer": "To make a cup of lavender tea, steep dried lavender flowers in hot water for 5-7 minutes." }, { "question": "What is the largest palace in the world?", "answer": "The largest palace in the world is the Palace of the Parliament in Romania." }, { "question": "How do I check my computer's hard drive health?", "answer": "To check your computer's hard drive health, use a diagnostic tool like CrystalDiskInfo or HD Tune." }, { "question": "What is the national flower of Greece?", "answer": "The national flower of Greece is the violet." }, { "question": "How do I make a chicken soup?", "answer": "To make a chicken soup, simmer chicken, vegetables, and broth together, and season with herbs and spices." }, { "question": "What is the largest strait in the world?", "answer": "The largest strait in the world is the Strait of Malacca." }, { "question": "How do I clean my dehumidifier?", "answer": "To clean your dehumidifier, empty it, wash the tank with soap and water, and rinse thoroughly." }, { "question": "What is the national animal of Greece?", "answer": "The national animal of Greece is the dolphin." }, { "question": "How do I make a cup

Other Sensors Input_0: You can see that there is a car. It is speed, straight-line distance from you, and angle in the direction your heading are respectively 0.0 m/s, 12.4 m, 180.0 °.You are now -0.000m laterally away from your driving route.
Question_0: What should be your next steering action?
MIMO-VLA_pred_0: Your next steering action should be to turn the steering wheel to the left.
***Openvla_pred_0*:** your steering action, your steering action, your steering action, your steering action, your steering action, your steering action, your steering action, your steering action, your steering action, your steering
DriverGPT4_pred_0: 0000000000000000000000000000000000
MIMO-VLA(no-image)_pred_0: right
MIMO-VLA(no-language)_pred_0: -- ,- ,- ,- ,- ,- ,- ,- ,- ,- ,- ,- ,- ,- ,-

---

Other Sensors Input_1: You can see that there are 2 cars. Their speed, straight-line distance from you, and angle in the direction your heading are respectively 5.07 2.4 m/s, 14.89 9.64 m, 14.18 179.72 °.You are now -0.029m laterally away from your driving route.
Question_1: How does the detected car's speed impact your driving action?
MIMO-VLA_pred_1: The detected car's speed impacts your driving action because if the detected car's speed is 0.000
Openvla_pred_1: driving speed and your speed and your speed and your speed and your speed and your speed and
DriverGPT4_pred_1: 0000000000000000000000000000000000
MIMO-VLA(no-image)_pred_1: The car's speed does not impact my driving action.
MIMO-VLA(no-language)_pred_1: -- ,- ,- ,- ,- ,- ,- ,- ,- ,- ,- ,- ,- ,- ,-

Figure 7: An example illustrating how MIMO-VLA seamlessly engages in conversation with a human while simultaneously making real-time driving decisions.

of rose tea?", "answer": "To make a cup of rose tea, steep dried rose petals in hot water for 5-7 minutes." }, { "question": "What is the largest castle in the world?", "answer": "The largest castle in the world is Prague Castle in the Czech Republic." }, { "question": "How do I check my computer's network speed?", "answer": "To check your computer's network speed, use an online speed test tool and follow the instructions." }, { "question": "What is the national flower of Turkey?", "answer": "The national flower of Turkey is the tulip." }, { "question": "How do I make a beef stew?", "answer": "To make a beef stew, simmer beef, vegetables, and broth together, and season with herbs and spices." }, { "question": "What is the largest canal in the world?", "answer": "The largest canal in the world is the Grand Canal in China." }, { "question": "How do I clean my air purifier?", "answer": "To clean your air purifier, remove and wash the filters, and wipe the exterior with a damp cloth." }, { "question": "What is the national animal of Turkey?", "answer": "The national animal of Turkey is the gray wolf." },

{ "question": "How do I make a cup of turmeric tea?", "answer": "To make a cup of turmeric tea, steep fresh turmeric slices in hot water for 5-7 minutes." }

A.11 Definition of the 50-Question Set

Questions are randomly selected from the following set of 50 questions: {"What are you seeing/observing?", "What are you paying attention to and why?", "Are there any traffic lights? What's the color of the traffic light?", "What's your current speed and steering angle?", "What is your action and why?", "Summarize the current driving scenario at a high level.", "How are you going to drive in this situation and why?", "What's the straight-line distance to the nearest car?", "What is the angle of the nearest car relative to your heading?", "Is there any lateral deviation from your driving route?", "What should be your next steering action?", "What should be your next acceleration command?", "Is there any moving object around you?", "Describe the position of the car relative to your heading.", "What is your current lateral position relative to your route?", "What would be a safe driving action given the detected car's details?", "What is the speed of the detected car?", "How far is the detected car from you?", "What angle should you adjust your steering to avoid collision?", "Why is it important to note the angle of the detected car?", "Is the detected car in motion?", "What should you be cautious of given the car's position?", "What action should be taken to maintain alignment on your driving route?", What should you avoid in this situation to prevent collision?, "What considerations are necessary for the detected car's speed?", "What's the importance of your current lateral position in planning the next action?", Why did you brake just now?, "What factors are influencing your next driving decision?", "Is there any obstacle directly ahead?", "How should you interpret the car's angle for your steering decision?", "What immediate adjustments are necessary for safe driving?", "How does the detected

car's speed impact your driving action?", "What should be your focus given the detected car's proximity and angle?", "What safe action is suggested based on the current scenario?", "What should you avoid in this situation to prevent collision?", "Is there a need for a speed adjustment?", "How will your steering angle change based on the detected car's angle?", "What should you consider for maintaining a safe path?", "How would you describe the current traffic conditions?", "What immediate action is necessary given your current lateral position?", "What factors need to be monitored to ensure safe navigation?", "Is the detected car influencing your path directly?", "What is the priority in adjusting your speed and direction?", " "}

A.12 Loss weights

Our loss weighting analysis yielded several key insights. First, increasing the image reconstruction weight (from 0.5 to 2) degraded decision-making performance while leaving language generation largely unaffected. Second, increasing the language loss weight (from 0.1 to 10) significantly enhanced conversational quality but substantially impaired driving performance. Third, raising the action prediction weight (from 0.1 to 10) markedly improved driving performance with minimal impact on language outputs. Based on these observations, we identified the optimal weighting scheme for autonomous driving applications as: language loss weight = 0.1, action loss weight = 10, and image reconstruction weight = 0.5. This configuration provides the best balance between driving performance and auxiliary task capabilities.

A.13 Further results on MIMO-VLA in answering questions

Other Sensors Input_33: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.45 m/s, 9.73 m, 179.82 °.You are now -0.009m laterally away from your driving route.
Question_33: What factors need to be monitored to ensure safe navigation?
MIMO-VLA_pred_33: You need to monitor the speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.45 m/s, 9.73 m, 179.82 °.
Openvla_pred_33: monitored to safe to ensure safe and to ensure safe monitored to safe
DriverGPT4_pred_33:
00000000000000000000000000000000000000
MIMO-VLA_image_pred_33: speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.45 m/s, 9.73 m, 179.82 °.You are now -0.009m laterally away from your driving route.
MIMO-VLA_language_pred_33: -

Other Sensors Input_34: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.49 m/s, 9.69 m, 179.81 °.You are now -0.011m laterally away from your driving route.
Question_34: What should be your next steering action?
MIMO-VLA_pred_34: Your next steering action should be to turn the steering wheel to the left.
Openvla_pred_34: your steering action and your next steering action and your steering action and your steering action and your steering action and
DriverGPT4_pred_34:
00000000000000000000000000000000000000
MIMO-VLA_image_pred_34: right
MIMO-VLA_language_pred_34: -

Other Sensors Input_35: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.47 m/s, 9.66 m, 179.79 °.You are now -0.012m laterally away from your driving route.
Question_35: What's the importance of your current lateral position in planning the next action?
MIMO-VLA_pred_35: The current lateral position is important in planning the next action because if you are too far away from your driving route.
Openvla_pred_35: your current position in planning the next action
DriverGPT4_pred_35:
00000000000000000000000000000000000000
MIMO-VLA_image_pred_35: Your current lateral position is important in planning the next action because if you don't plan the next action, you will not be able to reach your destination.
MIMO-VLA_language_pred_35: -

Other Sensors Input_36: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.41 m/s, 9.64 m, 179.78 °.You are now -0.014m laterally away from your driving

route.
Question_36: What safe action is suggested based on the current scenario?
MIMO-VLA_pred_36: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively.
Openvla_pred_36: suggested based on the current safe action on the current
DriverGPT4_pred_36:
0000000000000000000000000000000000000
MIMO-VLA_image_pred_36: The car is speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.41 m/s, 9.64 m, 179.78 °.You are now -0.014m laterally away from your driving route.
MIMO-VLA_language_pred_36: - -

Other Sensors Input_37: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.37 m/s, 9.63 m, 179.77 °.You are now -0.016m laterally away from your driving route.
Question_37: Is there any lateral deviation from your driving route?
MIMO-VLA_pred_37: There is no lateral deviation from your driving route.
Openvla_pred_37: driving route from your route and your route from your driving route from your route from your
DriverGPT4_pred_37:
0000000000000000000000000000000000000
MIMO-VLA_image_pred_37: Yes, there is lateral deviation from your driving route.
MIMO-VLA_language_pred_37: -

Other Sensors Input_38: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.37 m/s, 9.63 m, 179.76 °.You are now -0.018m laterally away from your driving route.
Question_38: How will your steering angle change based on the detected car's angle?
MIMO-VLA_pred_38: Your steering angle will change based on the detected car's angle.
Openvla_pred_38: your steering on the detected car's steering
DriverGPT4_pred_38:
0000000000000000000000000000000000000
MIMO-VLA_image_pred_38: The steering angle will change based on the detected car's angle.

Other Sensors Input_39: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.41 m/s, 9.63 m, 179.75 °.You are now -0.020m laterally away from your driving route.
Question_39: Is there any moving object around you?
MIMO-VLA_pred_39: Yes, there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.
Openvla_pred_39: moving object of you and your work with your work?
DriverGPT4_pred_39:
0000000000000000000000000000000000000
MIMO-VLA_image_pred_39: Yes, there is a car.
MIMO-VLA_language_pred_39: - -

Other Sensors Input_40: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.45 m/s, 9.63 m, 179.74 °.You are now -0.022m laterally away from your driving route.
Question_40: Is there any moving object around you?
MIMO-VLA_pred_40: Yes, there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.
Openvla_pred_40: moving object of you and your work with your work?
DriverGPT4_pred_40:
0000000000000000000000000000000000000
MIMO-VLA_image_pred_40: Yes, there is a car.
MIMO-VLA_language_pred_40: - -

Other Sensors Input_41: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.46 m/s, 9.63 m, 179.73 °.You are now -0.024m laterally away from your driving route.
Question_41: How does the detected car's speed impact your driving action?
MIMO-VLA_pred_41: The detected car's speed impacts your driving action because if the detected car's speed is faster than your driving speed
Openvla_pred_41: speed with your speed and your speed with your speed speed with your speed with your speed with your speed with your speed with

DriverGPT4_pred_41:
0000000000000000000000000000000000000000
MIMO-VLA_image_pred_41: The car's speed impacts my driving action because the car's speed is increasing, which means that the car's speed is getting faster
MIMO-VLA_language_pred_41: - -

Other Sensors Input_42: You can see that there is a car. Its speed, straight-line distance from you, and angle in the direction you're heading are respectively 2.43 m/s, 9.63 m, 179.73 °.You are now -0.027m laterally away from your driving route.
Question_42: What is your action and why?
MIMO-VLA_pred_42: My action is to turn left. I am turning left because I want to turn left.
Openvla_pred_42: " your action" and your action and your " your action" and your action and your
DriverGPT4_pred_42:
0000000000000000000000000000000000000000
MIMO-VLA_image_pred_42: My action is straight-line distance from you, and why is because I am going straight.
My action is straight-line distance from you, and why is because I am going straight. My action is straight-line distance from you, and why is because I am going straight.
MIMO-VLA_language_pred_42: - -

Other Sensors Input_43: You can see that there are 2 cars. Their speed, straight-line distance from you, and angle in the direction you're heading are respectively 5.07 2.4 m/s, 14.89 9.64 m, 14.18 179.72 °.You are now -0.029m laterally away from your driving route.
Question_43: What safe action is suggested based on the current scenario?
MIMO-VLA_pred_43: What safe action is suggested based on the current scenario? What safe action is suggested based on the current scenario? What safe action is suggested based
Openvla_pred_43: What safe action is based on the current? What safe action is based on the current
DriverGPT4_pred_43:
0000000000000000000000000000000000000000
MIMO-VLA_image_pred_43: straight-line distance from you, and angle in the direction you're heading are respectively 5.07 2.4 m/s, 14.89 9.64 m, 14.18 179.72 °.
MIMO-VLA_language_pred_43: - - - ,- - ,-

Other Sensors Input_44: You can see that there are 2 cars. Their speed, straight-line distance from you, and angle in the direction you're heading are respectively 4.98 2.38 m/s, 14.15 9.66 m, 14.95 179.71 °.You are now -0.032m laterally away from your driving route.
Question_44: Is the detected car in motion?
MIMO-VLA_pred_44: Yes answer: No answer: Yes answer: No answer
Openvla_pred_44: motion in the detected car in motion in the detected car in motion in the detected car in motion in the detected car in motion in the detected car in motion in the detected car in motion in the detected car in motion in the detected car in motion in the detected car in motion in the detected car in motion in the detected car in motion in the
DriverGPT4_pred_44:
0000000000000000000000000000000000000000
MIMO-VLA_image_pred_44: Yes, the detected car is in motion. """""""""""""""""""""""""""
MIMO-VLA_language_pred_44: - -
