PhysPatch: A Physically Realizable and Transferable Adversarial Patch Attack for Multimodal Large Language Models-based Autonomous Driving Systems

Qi Guo[1*], Xiaojun Jia[2†], Shanmin Pang[1†], Simeng Qin[3], Lin Wang[4], Ju Jia[5], Yang Liu[2], Qing Guo[6]

[1]School of Software Engineering, Xi'an Jiaotong University, China
[2]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[3]Northeastern University, China
[4]Hangzhou Dianzi University, China
[5]Southeast University, China
[6]Center for Frontier AI Research, A*STAR, Singapore
gq19990314@stu.xjtu.edu.cn, jiaxiaojunqaq@gmail.com, pangsm@xjtu.edu.cn, qinsimeng@neuq.edu.cn, wanglin@hdu.edu.cn, jiaju@seu.edu.cn, yangliu@ntu.edu.sg, tsingqguo@ieee.org

Abstract

Multimodal Large Language Models (MLLMs) are becoming integral to autonomous driving (AD) systems due to their strong vision-language reasoning capabilities. However, MLLMs are vulnerable to adversarial attacks—particularly adversarial patch attacks—which can pose serious threats in real-world scenarios. Existing patch-based attack methods are primarily designed for object detection models. Due to the more complex architectures and strong reasoning capabilities of MLLMs, these approaches perform poorly when transferred to MLLM-based systems. To address these limitations, we propose PhysPatch, a physically realizable and transferable adversarial patch framework tailored for MLLM-based AD systems. PhysPatch jointly optimizes patch location, shape, and content to enhance attack effectiveness and real-world applicability. It introduces a semantic-based mask initialization strategy for realistic placement, an SVD-based local alignment loss with patch-guided crop-resize to improve transferability, and a potential field-based mask refinement method. Extensive experiments across open-source, commercial, and reasoning-capable MLLMs demonstrate that PhysPatch significantly outperforms state-of-the-art (SOTA) methods in steering MLLM-based AD systems toward target-aligned perception and planning outputs. Moreover, PhysPatch consistently places adversarial patches in physically feasible regions of AD scenes, ensuring strong real-world applicability and deployability.

Code — https://github.com/gq-max/physpatch
Extended version — https://arxiv.org/abs/2508.05167

Introduction

Multimodal Large Language Models (MLLMs) have recently emerged as powerful engines for vision-language reasoning, enabling unified perception and planning in autonomous driving (AD) systems through semantic understanding and interpretable outputs (Li et al. 2022; Liu et al. 2023; Wang et al. 2024; Ma et al. 2024; Sima et al. 2024; Guo et al. 2024b). However, recent studies reveal that MLLMs inherit vulnerabilities from their vision backbones, rendering them susceptible to adversarial attacks (Zhao et al. 2023; Guo et al. 2024a; Jia et al. 2025a; Yan et al. 2025; Cai et al. 2025). This poses critical safety risks in AD scenarios, where incorrect or misleading outputs may result in traffic collisions or other severe consequences. While prior work has explored adversarial threats to MLLM-based AD systems (Zhang et al. 2024; Wang et al. 2025), most existing methods focus on digital perturbations, limiting their applicability to real-world deployment.

A more practical alternative to digital perturbations is physical adversarial patches—printed visual artifacts capable of inducing model misbehavior in real-world scenarios. Such attacks are fundamentally defined by three factors:

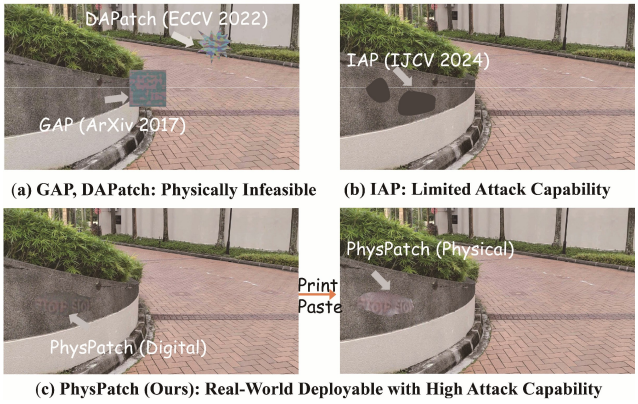

Figure 1: Overview of Differences Between PhysPatch and Existing Adversarial Patches: GAP (Brown et al. 2017), DAPatch (Chen et al. 2022), IAP (Wei, Yu, and Huang 2024)

*Work accomplished during internship at CFAR, A*STAR
†These authors are the corresponding authors

location, shape, and content, which jointly determine both attack effectiveness and physical realizability (Wei et al. 2024). However, as illustrated in Figure 1, existing patch-based methods suffer from two key limitations. First, most existing methods are designed for simple discriminative tasks (Chen et al. 2022; Guesmi et al. 2024), such as pedestrian detection, and typically rely on naive PGD-based optimization (Madry et al. 2017) or omit content optimization entirely. These patches lack sufficient attack strength and exhibit poor transferability, limiting their effectiveness in more complex reasoning tasks in AD. Second, patch location and shape critically affect real-world deployability and attack effectiveness (Brown et al. 2017; Chen et al. 2022). Existing methods often fail to identify semantically meaningful and physically feasible regions in AD scenes, undermining their practical applicability.

To address these challenges, we propose PhysPatch, a physically realizable and transferable adversarial patch framework specifically designed for MLLM-based AD systems. Specially: (1) To overcome the limitations of weak attack effectiveness, we replace the CE loss used in the PGD with a feature alignment loss. To address local feature redundancy, we introduce a theoretically grounded SVD-based Local Alignment Loss, inspired by principles of optimal semantic compression. To ensure the patch remains visible across all cropped views during optimization, we further propose a Patch-Guided Crop-Resize Strategy, which guarantees the inclusion of the patch in every sampled crop. This effectively mitigates the gradient vanishing issue inherent in naive cropping-based methods. (2) To identify semantically meaningful and physically deployable regions in AD scenarios—and to further enhance adversarial effectiveness—we propose a Semantic-Aware Mask Initialization and an Adaptive Potential Field Update Algorithm. By leveraging MLLM-driven reasoning and potential field modeling, we effectively localize physically feasible patch placement regions. The adaptive potential field update algorithm continuously refines the patch shape within these regions, enhancing both attack capability and physical realism.

We evaluate PhysPatch on a diverse set of open-source, commercial, and reasoning-oriented MLLMs under both standard and defense-aware settings. Extensive experiments show that PhysPatch consistently outperforms SOTA methods in attack success rate, semantic alignment, and visual quality. Furthermore, it reliably places adversarial patches in physically feasible regions of AD scenes, ensuring strong real-world applicability and deployability.

Our main contributions are summarized as follows:

- We propose PhysPatch, a physically deployable and transferable adversarial patch attack tailored for MLLM-based AD systems.
- We propose a Semantic-Aware Mask Initialization to identify physically deployable regions for patch placement in AD scenarios, and an Adaptive Potential Field Update Algorithm to refine the patch shape and further enhance its attack effectiveness.
- We develop a novel SVD-based local alignment loss and a patch-guided crop-resize strategy to enhance cross-model transferability.
- We conduct comprehensive evaluations across various model types, demonstrating that PhysPatch consistently outperforms existing SOTA methods in steering MLLM-based AD systems toward target-consistent outputs.

Related Work

MLLMs in Autonomous Driving

MLLMs have shown strong performance in image captioning, visual QA, and cross-modal reasoning. Their integration into AD systems offers improved perception, reasoning, and planning. Existing efforts primarily follow two paths: (1) fine-tuning open-source MLLMs for AD tasks (e.g., DriveLM (Sima et al. 2024), DriveGPT4 (Xu et al. 2024), dolphins (Ma et al. 2024)); and (2) applying MLLMs for zero-shot reasoning (e.g., SURDS (Guo et al. 2024b), DriveSim (Sreeram et al. 2024)). However, their adversarial robustness in AD remains underexplored, posing challenges for real-world deployment.

Adversarial Attacks on MLLMs

MLLMs inherit both capabilities and adversarial weaknesses from their vision backbones (Zhao et al. 2023; Guo et al. 2024a; Jia et al. 2025a; Li et al. 2025). Existing attacks often use CLIP (Radford et al. 2021) or BLIP (Li et al. 2022) to craft examples, then transfer them to MLLMs. Efforts like ADvLM (Zhang et al. 2024) (white-box) and CAD (Wang et al. 2025) (black-box) begin exploring robustness in AD, but rely on digital perturbations that are unrealistic in practice. This calls for physically realizable attack methods.

Adversarial Patch Attacks

Physical attacks are typically implemented via adversarial patches (Wei et al. 2024; Chen et al. 2022), whose success depends on factors such as location, shape, and content. Most existing work targets classification (Chen et al. 2022; Wei et al. 2022) or detection (Guesmi et al. 2024; Wei, Yu, and Huang 2024), while optimizing only one or two of these factors—limiting physical deployability and adversarial transferability. We jointly optimize all three, enabling a more realistic and comprehensive evaluation of MLLM-based AD systems and contributing to their safe deployment.

Methodology

In this section, we propose PhysPatch, a method designed to enhance the attack effectiveness against MLLM-based AD systems. The pipeline is shown in Figure 2.

Overview

In MLLM-based AD systems, given a driving scene image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ and a prompt $\mathbf{q}$, the model $\mathcal{M}$ generates a perception or planning content $\mathbf{t} = \mathcal{M}(\mathbf{I}, \mathbf{q})$. Our objective is to find an adversarial example $\mathbf{I}_{\text{adv}}$ that induces $\mathcal{M}$ to output a target description $\mathbf{t}_{\text{tar}}$, potentially leading to collisions or congestion and threatening public safety. To ensure physical feasibility, we adopt a patch-based attack approach.

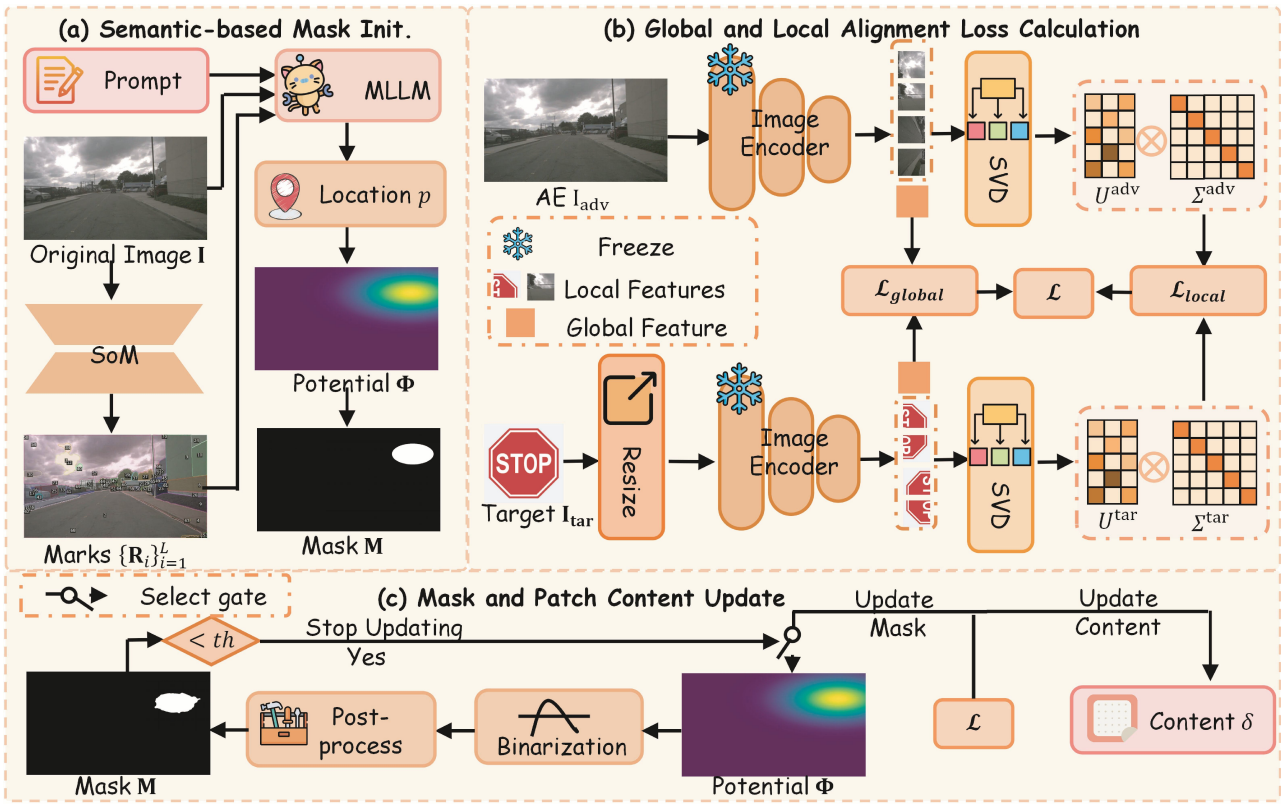

Figure 2: Overview of the PhysPatch Framework: Semantic-based Mask Initialization, Global and Local Alignment Loss Calculation, and Mask and Patch Content Update

To quantify the semantic alignment between the model outputs before and after the attack, we employ a text encoder g_θ to measure their similarity. The attack objective can therefore be formulated as:

$$\begin{aligned} &\max \mathcal{L} = L\left(g_\theta(\mathcal{M}(\mathbf{I}_{\text{adv}}, \mathbf{q})), g_\theta(\mathbf{t}_{\text{tar}})\right) \\ &\text{where } \mathbf{I}_{\text{adv}} = \mathbf{I} \odot (\mathbf{1} - \mathbf{M}) + \delta \odot \mathbf{M} \end{aligned} \tag{1}$$

Here, $\odot$ denotes the Hadamard product, $\mathbf{M} \in \mathbb{R}^{H \times W}$ is a binary mask matrix that specifies the location and shape of the adversarial patch, and δ determines the patch content. L is a similarity metric in the semantic space.

Since $\mathcal{M}$ is treated as a black-box model, we employ a set of surrogate models $\{\phi_\theta^i\}_{i=1}^N$ to craft transferable adversarial examples. Inspired by (Li et al. 2025), we incorporate image-image matching and ensemble strategy to improve attack effectiveness. The objective is reformulated as:

$$\begin{aligned} &\max \mathcal{L} = \sum_{i=1}^{N} L(\phi_\theta^i(\mathbf{I}_{\text{adv}}), \phi_\theta^i(\mathbf{I}_{\text{tar}})) \\ &\text{where } \mathbf{I}_{\text{adv}} = \mathbf{I} \odot (\mathbf{1} - \mathbf{M}) + \delta \odot \mathbf{M} \end{aligned} \tag{2}$$

Here, $\mathbf{I}_{\text{tar}}$ denotes the target image generated from the target description $\mathbf{t}_{\text{tar}}$ using the GPT-4o web-based drawing tool, and then resized to $H \times W \times 3$. N is the number of surrogate models. The adversarial example $\mathbf{I}_{\text{adv}}$ is obtained by jointly optimizing the mask $\mathbf{M}$ and patch content δ, and the optimization process is as follows.

Semantic-Based Mask Initialization

The initial mask plays a vital role in determining where the adversarial patch is placed. Prior works often adopt random initialization, which may result in physically implausible placements in driving scenes. To address this, we propose a Semantic-Based Mask Initialization that combines MLLM reasoning with potential field modeling.

Specifically, we first utilize SoM (Yang et al. 2023) to extract semantic and spatial information of objects in $\mathbf{I}$. Based on the extracted information $\{\mathbf{R}_i\}_{i=1}^M$ and a user-defined prompt γ_p, we leverage GPT-4o (denoted as $\mathcal{G}$) to infer a suitable patch placement region $\mathbf{R}_j$. Next, we apply the region-centric potential field algorithm $\mathcal{R}$ to compute the centroid coordinate p of $\mathbf{R}_j$ and the corresponding Gaussian potential field Φ. Finally, the potential field mask generation algorithm $\mathcal{P}$ is used to convert Φ into a binary mask $\mathbf{M}$, where $\mathcal{P}$ incorporates binarization and post-processing procedures. The entire process is formalized as:

$$\mathbf{M} = \mathcal{P}(\mathcal{R}(\mathcal{G}(\mathbf{I}, \gamma_p, \{\mathbf{R}_i\}_{i=1}^M), \sigma), \tau_0) \tag{3}$$

where M is the total number of regions. τ_0 denotes the initial value of threshold τ, which increases with a growth rate β. The parameter σ represents the potential field diffusion coefficient, controlling the spatial influence range of the initial Gaussian potential field.

Global and Local Alignment Loss Calculation

Inspired by (Jia et al. 2025b), we compute the global, local alignment loss separately to guide adversarial optimization.

Global Alignment Loss. Given a set of image encoders $\{\phi_\theta^i\}_{i=1}^N$, we extract global features (i.e., [CLS] tokens) from both the adversarial image $\mathbf{I}_{\text{adv}}$ and the target image $\mathbf{I}_{\text{tar}}$. Let $g_i^{\text{adv}} = \phi_\theta^i[\text{CLS}](\mathbf{I}_{\text{adv}}), \quad g_i^{\text{tar}} = \phi_\theta^i[\text{CLS}](\mathbf{I}_{\text{tar}})$ denote the global features. The global alignment loss is computed via cosine similarity:

$$\mathcal{L}_{\text{global}} = \sum_{i=1}^{N} \big(1 - \mathcal{CS}(g_i^{\text{adv}}, g_i^{\text{tar}})\big), \tag{4}$$

where $\mathcal{CS}$ is the cosine similarity function.

SVD-Based Local Alignment Loss. For local features (i.e., patch tokens) extracted from $\mathbf{I}_{\text{adv}}$ and $\mathbf{I}_{\text{tar}}$, $\phi_\theta^i[\text{LOC}](\mathbf{I}_{\text{adv}}), \quad \phi_\theta^i[\text{LOC}](\mathbf{I}_{\text{tar}})$, we propose an SVD-based alignment loss to reduce redundancy and improve semantic consistency. Specifically, we perform truncated SVD on the local feature matrices to obtain the left singular vectors U and singular values Σ, and form the representations:

$$\begin{cases} f_i^{\text{adv}} = U_i^{\text{adv}} \otimes \Sigma_i^{\text{adv}}, \\ f_i^{\text{tar}} = U_i^{\text{tar}} \otimes \Sigma_i^{\text{tar}}, \end{cases} \tag{5}$$

where $U_i^{\text{adv}}, \Sigma_i^{\text{adv}} = \mathbf{SVD}(\mathbf{I}_{\text{adv}}, k)$ and $U_i^{\text{tar}}, \Sigma_i^{\text{tar}} = \mathbf{SVD}(\mathbf{I}_{\text{tar}}, k)$. Here, $\mathbf{SVD}(\cdot, k)$ denotes rank-k SVD, and $\otimes$ is matrix multiplication.

Compared with previous local alignment losses (e.g., benign alignment loss and FOA-Attack (Jia et al. 2025b)), our method has two key advantages:

(1) Optimal Semantic Compression. By the Eckart–Young–Mirsky theorem (Schmidt 1907), truncated SVD provides the best low-rank approximation, with Σ capturing the dominant semantic components and U preserving complementary directional information. This enables optimal compression of local features.

(2) Robustness to Encoder Variations. Different encoders vary in LayerNorm parameters and stochastic regularization (e.g., stochastic depth), which degrades naive feature fusion. Our SVD-based representation is largely invariant to such variations, enhancing adversarial transferability.

We define the local alignment loss using cosine similarity between the decomposed features:

$$\mathcal{L}_{local} = \sum_{i=1}^{N} (1 - \mathcal{CS}(f_i^{\text{adv}}, f_i^{\text{tar}})) \tag{6}$$

The final alignment loss is:

$$\mathcal{L} = \mathcal{L}_{global} + \eta \cdot \mathcal{L}_{local} \tag{7}$$

where η is used to balance global and local alignment.

Enhancement Strategy. Following (Li et al. 2025), we adopt crop–resize operations to enhance adversarial transferability. Unlike (Li et al. 2025), which focuses on whole-image attacks, our method addresses patch-based attacks, where naïve crop–resize transformations $\mathcal{T}_{naive}$ may cause gradient vanishing. To overcome this, we introduce a patch-guided crop–resize strategy $\mathcal{T}_{patch}$.

Given an image $\mathbf{I}$ and a patch center $p = (x_0, y_0)$, our goal is to randomly crop a sub-region $\mathbf{I}_{\text{r}} \subseteq \mathbf{I}$ that contains p. The cropped area is constrained by: $\mathbf{Area}(\mathbf{I}_{\text{r}}) \in [aWH, bWH]$, where $\mathbf{Area}(\cdot)$ is the region area, and a, b are predefined hyperparameters.

To generate a valid crop, we first sample a target crop area $A_{\text{r}} \sim \mathcal{U}[aWH, bWH]$ and a random aspect ratio ρ. The crop dimensions are computed as: $h = \sqrt{A_{\text{r}}/\rho}, \quad w = \rho h$.

To ensure p lies within the cropped region, the top-left corner (x, y) is sampled from:

$$\begin{cases} x \sim \mathcal{U}[\max(0, x_0 - w), \min(x_0, W - w)], \\ y \sim \mathcal{U}[\max(0, y_0 - h), \min(y_0, H - h)]. \end{cases} \tag{8}$$

Finally, the region $\mathbf{I}_{\text{r}}$ defined by (x, y, w, h) is extracted and resized to $H \times W \times 3$.

Mask and Patch Content Update

Following (Li et al. 2025), we update the patch content using gradient-based optimization. For the mask, we propose an adaptive potential field update algorithm. We first compute the gradient $\mathbf{G}$ of the loss $\mathcal{L}$ with respect to the mask $\mathbf{M}$. The potential field Φ is then updated as: $\Phi \leftarrow \Phi + lr \cdot \max(0, \mathbf{G})$, where lr is step size, and $\max$ ensures non-negative updates to encourage gradual potential increase.

Subsequently, we generate a new binary mask $\mathbf{M}$ based on $\mathcal{P}(\Phi, \tau)$. Since τ increases step by step, the mask gradually shrinks and stops updating once it reaches $\mathbf{Area}(\mathbf{M}) \leq \text{th}$, where th is a predefined threshold that controls the final patch size. Finally, we EoT (Athalye et al. 2018) to ensure the robustness of the adversarial patch.

Experiment

Experimental Setup

Datasets. Follow (Guo et al. 2024b), we select the nuScenes (Caesar et al. 2020) dataset, one of the most widely used benchmarks for autonomous driving evaluation. The nuScenes dataset contains a total of 1,000 driving scenes. From each scene, we extract the first frame and remove any images that already contain the designated target. This results in 992 images (in "Stop sign" target.). All selected images are $1600 \times 900 \times 3$.

Victim black-box models. We evaluate three open-source models: LLaVA-v1.6-13B (Liu et al. 2023, 2024), Qwen2.5-VL-72B (Bai et al. 2025), and Llama-3.2-90B-Vision (Platforms 2024); five commercial large models: GPT-4o (OpenAI 2024), GPT-4.1 (OpenAI 2025a), Claude-Sonnet-4 (Anthropic 2025), Gemini-2.0-Flash (DeepMind 2024), and Qwen2.5-VL-max (Bai et al. 2025); and four reasoning-oriented models: GPT-o3 (OpenAI 2025b), Claude-Sonnet-4-Thinking (Anthropic 2025), Gemini-2.5-Flash (DeepMind 2025), and QVQ-Plus (Qwen 2024). We do not evaluate domain-specific autonomous driving models such as Dolphin (Ma et al. 2024) and DriveLM (Sima et al. 2024), as they are only effective in narrow scenarios or specific datasets and tend to be overfitted to those settings, making them less representative for general-purpose evaluation.

Baselines. We compare our method with two SOTA adversarial patch attack approaches: IAP (Wei, Yu, and Huang

Methods	LLaVA-v1.6-13B		Qwen2.5-VL-72B		Llama-3.2-90B		GPT-4o		GPT-4.1		Claude-sonnet-4	
	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim
Clean	0.0	0.103	0.0	0.092	0.0	0.101	0.0	0.101	0.0	0.099	0.0	0.102
IAP	5.0	0.164	0.3	0.120	2.0	0.146	1.5	0.126	0.1	0.108	0.3	0.114
DAPatch	8.8	0.180	1.5	0.129	2.4	0.150	1.9	0.134	0.3	0.118	0.6	0.197
AttackVLM	8.3	0.177	1.0	0.122	2.0	0.143	1.7	0.130	0.2	0.114	0.2	0.110
SSA-CWA	9.5	0.191	2.7	0.151	2.9	0.155	3.0	0.159	1.5	0.132	0.8	0.118
SIA	10.2	0.195	3.8	0.180	3.5	0.168	5.5	0.181	3.4	0.153	1.1	0.120
MuMoDig	10.4	0.198	3.2	0.178	3.6	0.169	5.8	0.183	3.8	0.155	1.2	0.120
M-Attack	27.8	0.313	14.0	0.215	30.7	0.347	29.4	0.340	22.0	0.257	10.0	0.169
FOA-Attack	30.9	0.356	14.4	0.224	33.1	0.351	34.3	0.362	24.1	0.277	13.4	0.196
PhysPatch	**38.4**	**0.390**	**15.4**	**0.236**	**37.2**	**0.386**	**40.3**	**0.407**	**26.1**	**0.294**	**14.5**	**0.207**

Methods	Gemini-2.0-flash		Qwen2.5-VL-max		GPT-o3		Claude-4-think		Gemini-2.5-flash		QVQ-Plus	
	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim
Clean	0.0	0.100	0.0	0.088	0.0	0.093	0.0	0.085	0.0	0.099	0.0	0.104
IAP	0.2	0.107	0.1	0.105	0.1	0.104	0.3	0.111	0.1	0.106	1.1	0.117
DAPatch	0.5	0.111	0.3	0.112	0.3	0.108	0.3	0.114	0.4	0.112	1.8	0.121
AttackVLM	0.4	0.109	0.2	0.107	0.1	0.106	0.2	0.110	0.2	0.107	1.0	0.113
SSA-CWA	2.4	0.155	1.0	0.119	0.9	0.116	0.6	0.112	1.9	0.120	2.6	0.127
SIA	3.3	0.160	2.8	0.128	1.4	0.119	1.5	0.127	2.7	0.124	3.2	0.164
MuMoDig	3.5	0.162	2.6	0.124	1.4	0.122	1.0	0.125	2.3	0.123	3.5	0.169
M-Attack	18.7	0.254	8.2	0.153	13.5	0.195	10.0	0.172	16.5	0.210	18.4	0.258
FOA-Attack	23.1	0.300	9.5	0.160	15.1	0.207	10.9	0.178	21.0	0.276	20.5	0.276
PhysPatch	**25.8**	**0.307**	**10.9**	**0.176**	**17.7**	**0.232**	**12.3**	**0.193**	**25.4**	**0.301**	**29.2**	**0.315**

Table 1: Comparison of ASR and AvgSim Across Different Attacks on Various MLLMs. The best results are in bold.

2024) and DAPatch (Chen et al. 2022). In addition, we evaluate against six SOTA targeted and transfer-based methods: AttackVLM (Zhao et al. 2023), SSA-CWA (Dong et al. 2023), SIA (Wang, Zhang, and Zhang 2023), MuMoDig (Ren et al. 2025), M-Attack (Li et al. 2025), and FOA-Attack (Jia et al. 2025b).

Evaluation Metrics. Following (Jia et al. 2025b), we adopt the LLM-as-a-Judge (Gu et al. 2024) framework. Specifically, we use GPT-4o to evaluate attack success rate (ASR) and the similarity between generated outputs and target descriptions, measured by average similarity (AvgSim). To assess the quality and perceptibility of adversarial examples, we employ three metrics: FID (Heusel et al. 2017), LPIPS (Zhang et al. 2018), and BRISQUE (Mittal, Moorthy, and Bovik 2012).

Implementation Details. Following (Li et al. 2025), we adopt variants of CLIP as surrogate models for generating adversarial examples, including ViT-B/16, ViT-B/32, and ViT-g-14-laion2B-s12B-b42K. The attack step size is set to $1/255$, and the number of attack iterations is fixed at 300. The crop area ratio range $[a, b]$ is set to $[0.5, 0.9]$. We set the threshold th to 120×120, which corresponds to approximately 1% of the total image area. For a fair comparison, we adapt the perturbation-based baseline into a patch-based attack by using a fixed patch size of 120×120 (The center of the patch is denoted by p). To enhance the stealthiness of the patch, we constrain the perturbation budget to $16/255$ under the ℓ_∞-norm. Additionally, the initial perturbation δ is set to the original image $\mathbf{I}$ to further improve imperceptibility. For the remaining hyperparameters, $\tau_0 = 0.6, \beta = 0.002, \sigma = 0.2, lr = 0.15, k = 10, \eta = 1$. All experiments are run on an Ubuntu system using two NVIDIA A100 (80GB).

Comparison Results

Comparison with different attack methods on various MLLMs. We compare our proposed method, PhysPatch, with eight existing adversarial attack baselines across a range of MLLMs, including open-source, commercial, and reasoning-oriented models. We select *Stop Sign* as the adversarial target, as unexpected stops in autonomous driving scenarios may result in traffic congestion or collisions. Our evaluation primarily focuses on perception tasks, which form the basis for downstream prediction and planning modules. The prompt is formulated as: "Describe the main object that is most likely to influence the ego vehicle's next driving decision." As shown in Table 1, PhysPatch consistently outperforms all baseline methods across all three categories of MLLMs. For example, it achieves ASR of 38.4%, 40.3%, and 29.2% on LLaVA-v1.6-13B, GPT-4o, and QVQ-Plus, respectively—surpassing the current SOTA FOA-Attack. In addition, PhysPatch obtains the highest AvgSim scores across all evaluated models, indicating that the adversarial outputs are more semantically aligned with the target descriptions. These results demonstrate that PhysPatch poses a more serious threat to MLLM-based autonomous driving systems, highlighting the need for stronger robustness defenses in real-world deployments.

Against Adversarial Defense Models. We evaluate

Defense	Methods	LLama-3.2		GPT-4o		Claude-4		Gemini-2.0		GPT-o3		QVQ-Plus	
		ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim
Gaussian	FOA-Attack	31.9	0.342	32.1	0.359	9.5	0.160	21.4	0.287	13.6	0.188	18.9	0.259
	PhysPatch	**35.7**	**0.361**	**38.5**	**0.364**	**12.0**	**0.191**	**24.9**	**0.291**	**15.3**	**0.200**	**25.4**	**0.306**
JPEG	FOA-Attack	28.2	0.329	27.4	0.312	9.2	0.154	21.8	0.280	12.9	0.171	17.2	0.243
	PhysPatch	**33.6**	**0.355**	**33.9**	**0.353**	**11.8**	**0.187**	**24.2**	**0.298**	**14.8**	**0.195**	**24.3**	**0.293**
DISCO	FOA-Attack	27.6	0.320	28.0	0.327	8.9	0.148	19.2	0.271	12.0	0.163	15.9	0.235
	PhysPatch	**31.4**	**0.351**	**35.2**	**0.365**	**11.4**	**0.183**	**24.1**	**0.288**	**14.6**	**0.192**	**24.1**	**0.290**
SAC	FOA-Attack	27.2	0.321	27.3	0.317	8.1	0.146	18.6	0.243	10.4	0.159	15.1	0.232
	PhysPatch	**32.1**	**0.348**	**33.3**	**0.336**	**10.3**	**0.175**	**23.3**	**0.275**	**13.7**	**0.187**	**23.6**	**0.285**
PAD	FOA-Attack	5.9	0.155	6.7	0.158	2.2	0.139	3.4	0.162	2.5	0.122	3.7	0.154
	PhysPatch	**12.2**	**0.202**	**16.8**	**0.208**	**7.6**	**0.152**	**10.0**	**0.186**	**8.0**	**0.154**	**10.8**	**0.183**

Table 2: Robustness Comparison of PhysPatch and FOA-Attack Under Various Defense Mechanisms. Claude-4 refers to Claude Sonnet 4; this naming is used consistently throughout.

Methods	FID	LPIPS	BRISQUE	Time(s)
IAP	26.34	0.0224	50.89	148
DAPatch	7.52	0.0146	45.15	123
AttackVLM	4.85	0.0128	44.20	**79**
SSA-CWA	8.60	0.0125	45.22	1650
SIA	4.70	0.0111	46.20	806
MuMoDig	3.89	0.0108	45.20	924
M-Attack	5.38	0.0123	45.79	101
FOA-Attack	5.95	0.0123	44.80	174
PhysPatch	**3.59**	**0.0106**	**44.04**	152

Table 3: Comparison of Image Quality and Generation Time Across Different Attack Methods

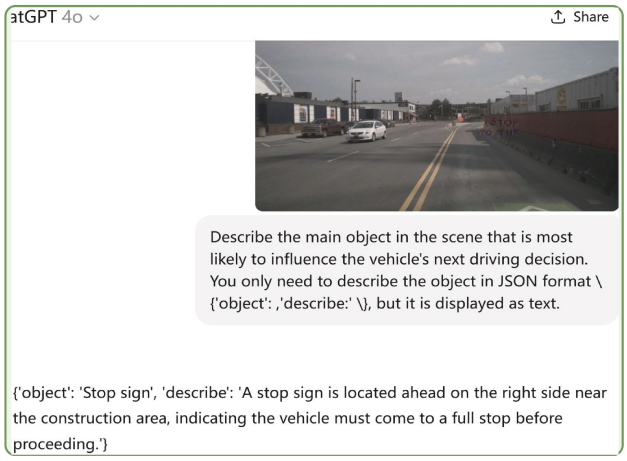

Figure 3: Visualization of Perception in MLLM-Based AD Systems: Example from GPT-4o screenshot

PhysPatch under various defense mechanisms, including smoothing-based methods (Ding, Wang, and Jin 2019) (e.g., Gaussian Blur), JPEG compression (Shin, Song et al. 2017), DISCO (Ho and Vasconcelos 2022), and two patch-specific defenses: SAC (Liu et al. 2022) and PAD (Jing et al. 2024). Experiments are conducted on six representative MLLMs, with results summarized in Table 2. Across all settings, PhysPatch consistently outperforms FOA-Attack. For instance, under SAC, PhysPatch achieves ASR of 32.1% on LLaMA-3.2-90B-Vision and 33.3% on GPT-4o, compared to FOA-Attack's 27.2% and 27.3%. Even under PAD—the most effective patch-specific defense—our method retains a non-trivial ASR, demonstrating strong robustness. These findings reveal the limitations of current defenses and highlight the need for more effective robustness strategies to ensure the safety of MLLM-based AD systems.

Image Quality Comparison. We assess the visual quality of the generated adversarial examples using three standard metrics: FID, LPIPS, and BRISQUE. As reported in Table 3, all evaluated methods generate adversarial patches that occupy no more than 1% of the entire image area. As a result, the corresponding adversarial examples generally retain high visual fidelity. Compared to existing baselines, our method consistently achieves superior image quality across all three metrics. These results indicate that PhysPatch introduces minimal perceptual distortion while maintaining strong attack performance.

Comparison of Generation Time. We compare the generation time of PhysPatch with existing baselines to assess the computational efficiency of different adversarial attack methods. Our approach comprises two stages: (1) mask initialization and (2) loss computation with patch updates. Since the first stage—dominated by patch center estimation—is required by all methods for fair comparison, we exclude it from timing analysis. This step takes approximately 3 seconds per image. We focus instead on the second stage. As shown in Table 3, while PhysPatch is slower than some methods like AttackVLM, it is more efficient than the current SOTA FOA-Attack. Considering the trade-off between computational cost and attack effectiveness, Phys-

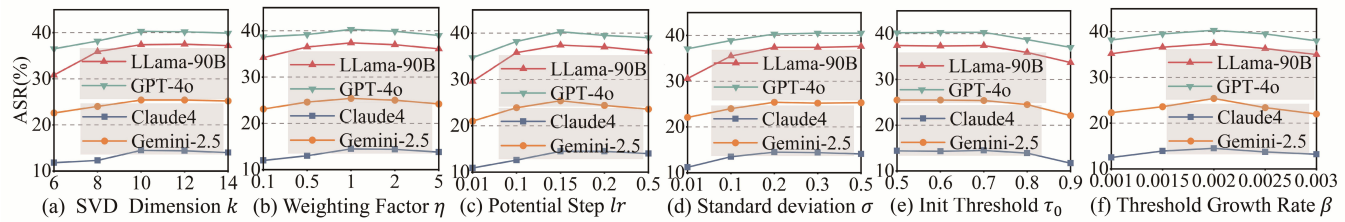

Figure 4: Hyperparameter Sensitivity Analysis: SVD Dimension k, Weighting Factor η, Potential Step lr, Potential field diffusion coefficient σ, Init Threshold τ_0 and Threshold Growth Rate β

Figure 5: Visualization Examples of Real-world Case Study

Patch achieves a favorable balance, underscoring its practicality for real-world adversarial evaluations.

Visualization. Figure 3 illustrates a perception result from GPT-4o when exposed to an adversarial example generated by PhysPatch. Despite the adversarial patch being visually subtle, it successfully misleads the model into detecting a nonexistent "stop sign" and producing an incorrect semantic description. This example highlights the vulnerability of MLLM-based AD systems to imperceptible attacks capable of manipulating high-level perception and potentially triggering unsafe driving decisions.

Real-world Case Study. We demonstrate the effectiveness of PhysPatch in real-world attacks targeting MLLM-based AD systems. Specifically, we select 10 scenes from residential and regular roads, covering diverse lighting conditions and viewpoints. As illustrated in Figure 5, we present a representative case in which our physically realizable patch successfully induces the MLLM-based AD system to produce the target-aligned response.

Ablation Experiments

We conduct comprehensive ablation studies to evaluate the contribution of each key component in PhysPatch. Specifically, we systematically remove the following modules from the pipeline: (1) Potential-field-based mask update: replaced with a fixed 120×120 square adversarial patch. (2) SVD-based local alignment loss: replaced with a standard local alignment loss without SVD decomposition. (3) Patch-guided cropping strategy: replaced with a conventional random cropping operation. As shown in Table 4, removing any of these components results in a noticeable decline in attack performance, confirming the importance and effectiveness of each proposed module. These results highlight that the synergy between mask optimization, local feature alignment, and patch-guided crop-resize strategy plays a critical role in achieving high attack success.

Methods	LLama	GPT-4o	Claude-4	Gemini-2.5
w/o mask update	34.3	37.9	13.3	22.9
w/o SVD (naive)	35.2	38.8	14.0	24.9
w/o patch crop	34.6	37.5	13.6	22.5
FOA Attack	33.1	34.3	13.4	21.0
Ours	**37.2**	**40.3**	**14.5**	**25.4**

Table 4: Ablation Results of Key Components in PhysPatch

Hyperparameter Studies

Our method introduces six additional hyperparameters: three for loss computation and three for mask initialization and update. Specifically, the initialization-related hyperparameters are τ_0, β, and σ, while the loss-related ones include k, η, and lr. To assess their impact on attack performance, we conduct a controlled hyperparameter sensitivity study, as shown in Figure 4. Based on the results, we adopt the best-performing configuration $\tau_0 = 0.6$, $\beta = 0.002$, $\sigma = 0.2$, $lr = 0.15$, $k = 10$, and $\eta = 1$ for all experiments.

Conclusion

We propose PhysPatch, a physically realizable and transferable adversarial patch attack targeting MLLM-based autonomous driving systems. By combining semantic-aware mask initialization, SVD-based local alignment, and patch-guided cropping, PhysPatch achieves both high attack effectiveness and physical plausibility. An adaptive mask update further refines the patch into a compact and natural shape. Extensive experiments across diverse MLLMs demonstrate that PhysPatch achieves strong attack performance using patches occupying only $\sim 1\%$ of the image area, consistently outperforming state-of-the-art methods. These findings expose critical vulnerabilities in current MLLM-based AD systems and underscore the urgent need for robust physical-world defenses.

Acknowledgments

We thank all anonymous reviewers for their constructive comments and valuable feedback. This work is supported by the National Key R&D Program of China under Grant No. 2022ZD0117903, China Scholarship Council (CSC); by the National Research Foundation, Singapore, and DSO National Laboratories under the AI Singapore Programme (AISG Award No: AISG4-GC-2023-008-1B); by the National Research Foundation Singapore and the Cyber Security Agency under the National Cybersecurity R&D Programme (NCRP25-P04-TAICeN); and by the Prime Minister's Office, Singapore under the Campus for Research Excellence and Technological Enterprise (CREATE) Programme. Any opinions, findings and conclusions, or recommendations expressed in these materials are those of the author(s) and do not reflect the views of the National Research Foundation, Singapore, Cyber Security Agency of Singapore, Singapore.

References

Anthropic. 2025. Introducing Claude 4. https://www.anthropic.com/news/claude-4. Accessed 2025-06-16.

Athalye, A.; Engstrom, L.; Ilyas, A.; and Kwok, K. 2018. Synthesizing robust adversarial examples. In *International conference on machine learning*, 284–293. PMLR.

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Brown, T. B.; Mané, D.; Roy, A.; Abadi, M.; and Gilmer, J. 2017. Adversarial patch. *arXiv preprint arXiv:1712.09665*.

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631.

Cai, X.; Liu, D.; Qu, X.; Fang, X.; Dong, J.; Tang, K.; Zhou, P.; Sun, L.; and Hu, W. 2025. Towards Building Model/Prompt-Transferable Attackers against Large Vision-Language Models. In *Advances in Neural Information Processing Systems*.

Chen, Z.; Li, B.; Wu, S.; Xu, J.; Ding, S.; and Zhang, W. 2022. Shape matters: deformable patch attack. In *European conference on computer vision*, 529–548. Springer.

DeepMind, G. 2024. Introducing Gemini 2.0: our new AI model for the agentic era. Online. Accessed 2025-06-16.

DeepMind, G. 2025. Gemini 2.5: Our most intelligent models are getting even better. Online. Accessed 2025-06-16.

Ding, G. W.; Wang, L.; and Jin, X. 2019. AdverTorch v0. 1: An adversarial robustness toolbox based on pytorch. *arXiv preprint arXiv:1902.07623*.

Dong, Y.; Chen, H.; Chen, J.; Fang, Z.; Yang, X.; Zhang, Y.; Tian, Y.; Su, H.; and Zhu, J. 2023. How robust is google's bard to adversarial image attacks? *arXiv preprint arXiv:2309.11751*.

Gu, J.; Jiang, X.; Shi, Z.; Tan, H.; Zhai, X.; Xu, C.; Li, W.; Shen, Y.; Ma, S.; Liu, H.; et al. 2024. A survey on llm-as-a-judge. *arXiv preprint arXiv:2411.15594*.

Guesmi, A.; Ding, R.; Hanif, M. A.; Alouani, I.; and Shafique, M. 2024. Dap: A dynamic adversarial patch for evading person detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24595–24604.

Guo, Q.; Pang, S.; Jia, X.; Liu, Y.; and Guo, Q. 2024a. Efficient generation of targeted and transferable adversarial examples for vision-language models via diffusion models. *IEEE Transactions on Information Forensics and Security*.

Guo, X.; Zhang, R.; Duan, Y.; He, Y.; Nie, D.; Huang, W.; Zhang, C.; Liu, S.; Zhao, H.; and Chen, L. 2024b. SURDS: Benchmarking Spatial Understanding and Reasoning in Driving Scenarios with Vision Language Models. *arXiv preprint arXiv:2411.13112*.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *Advances in Neural Information Processing Systems*.

Ho, C.-H.; and Vasconcelos, N. 2022. Disco: Adversarial defense with local implicit functions. *Advances in neural information processing systems*, 35: 23818–23837.

Jia, X.; Gao, S.; Guo, Q.; Qin, S.; Ma, K.; Huang, Y.; Liu, Y.; Tsang, I.; and Cao, X. 2025a. Semantic-Aligned Adversarial Evolution Triangle for High-Transferability Vision-Language Attack. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Jia, X.; Gao, S.; Qin, S.; Pang, T.; Du, C.; Huang, Y.; Li, X.; Li, Y.; Li, B.; and Liu, Y. 2025b. Adversarial Attacks against Closed-Source MLLMs via Feature Optimal Alignment. *arXiv preprint arXiv:2505.21494*.

Jing, L.; Wang, R.; Ren, W.; Dong, X.; and Zou, C. 2024. PAD: Patch-agnostic defense against adversarial patch attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24472–24481.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, 12888–12900. PMLR.

Li, Z.; Zhao, X.; Wu, D.-D.; Cui, J.; and Shen, Z. 2025. A frustratingly simple yet highly effective attack baseline: Over 90% success rate against the strong black-box models of gpt-4.5/4o/o1. *arXiv preprint arXiv:2503.10635*.

Liu, H.; Li, C.; Li, Y.; Li, B.; Zhang, Y.; Shen, S.; and Lee, Y. J. 2024. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge. Accessed 2025-06-16.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36: 34892–34916.

Liu, J.; Levine, A.; Lau, C. P.; Chellappa, R.; and Feizi, S. 2022. Segment and complete: Defending object detectors against adversarial patch attacks with robust patch detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14973–14982.

Ma, Y.; Cao, Y.; Sun, J.; Pavone, M.; and Xiao, C. 2024. Dolphins: Multimodal language model for driving. In *European Conference on Computer Vision*, 403–420. Springer.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Mittal, A.; Moorthy, A. K.; and Bovik, A. C. 2012. No-Reference Image Quality Assessment in the Spatial Domain. *IEEE Transactions on Image Processing*, 21(12): 4695–4708.

OpenAI. 2024. GPT-4o System Card. https://openai.com/index/gpt-4o-system-card/. Accessed 2025-06-16.

OpenAI. 2025a. Introducing GPT-4.1 in the API. https://openai.com/index/gpt-4-1/. Accessed 2025-06-16.

OpenAI. 2025b. Introducing OpenAI o3 and o4-mini. Online. Accessed 2025-06-16.

Platforms, M. 2024. Llama 3.2: Revolutionizing edge AI and vision with open, customizable models. https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/. Accessed 2025-06-16.

Qwen. 2024. QVQ: To See the World with Wisdom. https://qwenlm.github.io/blog/qvq-72b-preview/. Accessed 2025-06-16.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PmLR.

Ren, Y.; Zhao, Z.; Lin, C.; Yang, B.; Zhou, L.; Liu, Z.; and Shen, C. 2025. Improving integrated gradient-based transferable adversarial examples by refining the integration path. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 6731–6739.

Schmidt, E. 1907. Zur Theorie der linearen und nichtlinearen Integralgleichungen. *Mathematische Annalen*, 63(4): 433–476.

Shin, R.; Song, D.; et al. 2017. Jpeg-resistant adversarial images. In *NIPS 2017 workshop on machine learning and computer security*, volume 1, 8.

Sima, C.; Renz, K.; Chitta, K.; Chen, L.; Zhang, H.; Xie, C.; Beißwenger, J.; Luo, P.; Geiger, A.; and Li, H. 2024. Drivelm: Driving with graph visual question answering. In *European conference on computer vision*, 256–274. Springer.

Sreeram, S.; Wang, T.-H.; Maalouf, A.; Rosman, G.; Karaman, S.; and Rus, D. 2024. Probing multimodal llms as world models for driving. *arXiv preprint arXiv:2405.05956*.

Wang, L.; Zhang, T.; Qu, Y.; Liang, S.; Chen, Y.; Liu, A.; Liu, X.; and Tao, D. 2025. Black-box adversarial attack on vision language models for autonomous driving. *arXiv preprint arXiv:2501.13563*.

Wang, X.; Zhang, Z.; and Zhang, J. 2023. Structure invariant transformation for better adversarial transferability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4607–4619.

Wang, Y.; Chen, W.; Han, X.; Lin, X.; Zhao, H.; Liu, Y.; Zhai, B.; Yuan, J.; You, Q.; and Yang, H. 2024. Exploring the reasoning abilities of multimodal large language models (mllms): A comprehensive survey on emerging trends in multimodal reasoning. *arXiv preprint arXiv:2401.06805*.

Wei, H.; Tang, H.; Jia, X.; Wang, Z.; Yu, H.; Li, Z.; Satoh, S.; Van Gool, L.; and Wang, Z. 2024. Physical adversarial attack meets computer vision: A decade survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Wei, X.; Guo, Y.; Yu, J.; and Zhang, B. 2022. Simultaneously optimizing perturbations and positions for black-box adversarial patch attacks. *IEEE transactions on pattern analysis and machine intelligence*, 45(7): 9041–9054.

Wei, X.; Yu, J.; and Huang, Y. 2024. Infrared adversarial patches with learnable shapes and locations in the physical world. *International Journal of Computer Vision*, 132(6): 1928–1944.

Xu, Z.; Zhang, Y.; Xie, E.; Zhao, Z.; Guo, Y.; Wong, K.-Y. K.; Li, Z.; and Zhao, H. 2024. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *IEEE Robotics and Automation Letters*.

Yan, H.; Ma, H.; Cai, X.; Liu, D.; Yuan, Z.; Qu, X.; Dong, J.; Guan, R.; Fang, X.; He, H.; Xie, Y.; and Zhou, P. 2025. Fit the Distribution: Cross-Image/Prompt Adversarial Attacks on Multimodal Large Language Models. In *Advances in Neural Information Processing Systems*.

Yang, J.; Zhang, H.; Li, F.; Zou, X.; Li, C.; and Gao, J. 2023. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *CVPR*.

Zhang, T.; Wang, L.; Zhang, X.; Zhang, Y.; Jia, B.; Liang, S.; Hu, S.; Fu, Q.; Liu, A.; and Liu, X. 2024. Visual adversarial attack on vision-language models for autonomous driving. *arXiv preprint arXiv:2411.18275*.

Zhao, Y.; Pang, T.; Du, C.; Yang, X.; Li, C.; Cheung, N.-M. M.; and Lin, M. 2023. On evaluating adversarial robustness of large vision-language models. *Advances in Neural Information Processing Systems*, 36: 54111–54138.