Comparative Analysis of Patch Attack on VLM-Based Autonomous Driving Architectures

David Fernandez, Pedram MohajerAnsari, Amir Salarpour, Long Cheng, Abolfazl Razi, Mert D. Pesé

***Abstract*—Vision-language models are emerging for autonomous driving, yet their robustness to physical adversarial attacks remains unexplored. This paper presents a systematic framework for comparative adversarial evaluation across three VLM architectures: Dolphins, OmniDrive (Omni-L), and LeapVAD. Using black-box optimization with semantic homogenization for fair comparison, we evaluate physically realizable patch attacks in CARLA simulation. Results reveal severe vulnerabilities across all architectures, sustained multi-frame failures, and critical object detection degradation. Our analysis exposes distinct architectural vulnerability patterns, demonstrating that current VLM designs inadequately address adversarial threats in safety-critical autonomous driving applications.**

***Index Terms*—Vision-Language Models, Autonomous Driving, Physical Adversarial Patches, Black-Box Attacks**

I. Introduction

Vision-language models (VLMs) are emerging in autonomous driving (AD), integrating visual perception with language-based reasoning to create interpretable, end-to-end decision-making systems [30]. Unlike traditional modular pipelines that separate perception, prediction, and planning, VLMs leverage large language models (LLMs), enabling them to handle complex driving scenarios through natural language (NL) understanding. Recent systems [15], [25], [29] show that VLMs can generate human-interpretable driving decisions and generalize to previously unseen scenarios.

Despite their promise, the robustness of VLM-based driving systems to physical adversarial attacks remains unclear. Physical adversarial patches pose a significant threat to safety-critical systems [5]. While attacks on traditional vision models are well-studied [2], how they affect end-to-end VLM driving models presents a unique problem. The connection between vision and language creates a complex target, as attacks can corrupt visual data, distort scene understanding, and ultimately cause unsafe driving actions. This robustness question is further complicated by the proliferation of VLM architectures for AD. Recent surveys [18] identify over 20 distinct architectures with varying sensor modalities, vision encoders, language models, and fusion mechanisms. Many systems lack publicly available implementations (e.g., DriveGPT4 [29], VLP [19], DriVLMe [10]), preventing reproducible analysis. Others incorporate additional sensors such as LiDAR (e.g., LMDrive [25]) that fundamentally alter the attack surface, making fair architectural comparisons impractical. Furthermore, these architectures generate a wide range of outputs. As a result, no framework exists to evaluate how well different VLM architectures hold up to adversarial attacks or to compare them systematically, which limits our ability to understand which design choices improve robustness.

This paper introduces a systematic framework (Figure 1) for a comparative evaluation of adversarial robustness in VLM-based autonomous driving systems using the CARLA simulator [3]. We address the challenge of model diversity by establishing rigorous selection criteria for comparable VLM architectures, designing a unified evaluation protocol that accounts for architectural differences while enabling a fair comparison, and developing a semantic-level attack and evaluation methodology that works across diverse output formats. Our framework employs black-box Natural Evolution Strategies (NES) [11] optimization with semantic similarity loss to generate physically realizable adversarial patches, ensuring attacks are architecture-agnostic and deployment-realistic.

We evaluate three representative VLM architectures: **Dolphins** [15], **OmniDrive** (Omni-L) [27], and **LeapVAD** [16]. These systems span the design space of vision-language integration while remaining comparable through a shared modality (camera-only), a shared task (closed-loop driving), and available open-source implementations. To ensure fairness across these architectures, which have different output formats, we introduce a semantic homogenization layer. This enables an architecture-agnostic comparison of attack effectiveness, perceptual degradation, and scene understanding corruption.

We ran closed-loop experiments in the CARLA simulator using two scenarios: one targeting crosswalk pedestrian detection suppression and another focused on highway steering manipulation. Our black-box NES [28] optimizer generated physically constrained adversarial patches. These patches were placed on realistic advertising infrastructure (bus shelters and billboards) and used Expectation over Transformation (EoT) [1] to ensure they remained robust across various viewing conditions. For each VLM, we measured the attack success rate at different spatial distances, its temporal consistency (multi-frame persistence), its impact on object detection (failure to see critical objects), and the degradation of overall scene understanding (using BLEU-4 and semantic similarity).

David Fernandez, Pedram MohajerAnsari, Amir Salarpour, Long Cheng, Abolfazl Razi, and Mert D. Pesé are with the School of Computing, Clemson University, Clemson, SC, USA {dferna3, pmohaje, asalarp, lcheng2, arazi, mpese}@clemson.edu

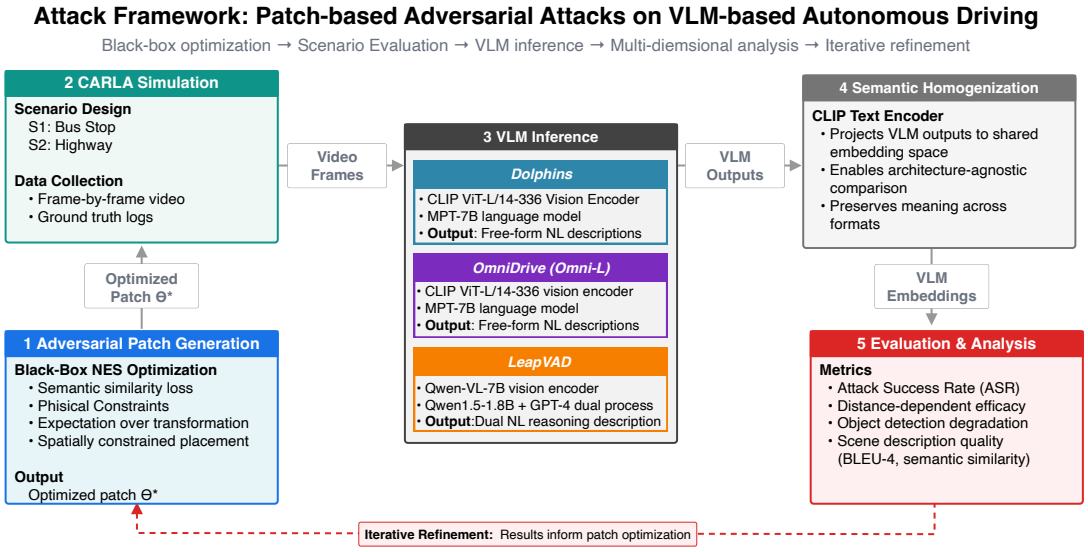

Fig. 1. Overview of Attack Framework: (1) Black-box NES optimization generates adversarial patches; (2) CARLA scenarios capture frame video sequences; (3) VLM architectures process scenes: **Dolphins**, **OmniDrive** (Omni-L), and **LeapVAD**; (4) Semantic homogenization layer projects all VLM outputs into a unified embedding space; (5) Multi-dimensional evaluation.

Our evaluation reveals vulnerabilities across all tested architectures. The comparative evaluation also reveals distinct architectural vulnerability patterns. The adversarial patches achieved high overall attack success rates (ASRs) ranging from 73.5% to 76.0%, a 12-20x increase over baseline inappropriate action rates. These attacks proved both effective at critical distances (10-25 meters) and highly persistent, causing sustained failures for 6.2 to 7.8 consecutive frames. Finally, the patches caused worse scene understanding, with descriptions showing low BLEU-4 (0.18-0.31) and semantic similarity (0.49-0.67) scores when compared against their benign counterparts. This paper makes the following contributions:

- A framework for architecture-agnostic adversarial evaluation which introduces a semantic homogenization layer to project heterogeneous VLM outputs into a unified embedding space to enable a fair, black-box comparison;
- A comprehensive evaluation methodology that uses multi-dimensional metrics to measure an attack's impact on action recommendations, object detection, scene understanding, spatial robustness, and temporal persistence;
- An empirical demonstration showing that adversarial patches can compromise VLM-based driving systems.

II. Related Work

VLMs for AD. LLMs have recently been used for driving scenario analysis [6], showing promise for more understandable explanations of driving scenes. Building on this, recent advances of LLMs have led to the development of VLMs for end-to-end AD. These architectures use diverse integration strategies: Dolphins employs cross-attention mechanisms, OmniDrive explores two architectural variants. Omni-Q uses Q-Former-based 3D perception alignment, while Omni-L leverages MLP projection. LeapVAD introduces a dual-process architecture that distinguishes between a fast Heuristic Process and a slow Analytic Process. While these systems show promising performance in normal scenarios [8], their robustness to adversarial attacks, especially how their different architectures affect vulnerability patterns, remains unexplored. Recent work has also explored lightweight LLMs for real-world deployment [4], [7], though the security implications of such edge deployments remain understudied.

Adversarial Attacks on Computer Vision. Physical adversarial patches have proven to be a significant vulnerability for computer vision systems. Brown et al. [2] showed that printed patterns could reliably fool image classifiers in the real world. Later research extended these attacks to object detectors [5] and segmentation models [9], revealing that safety-critical perception components can be fooled by these carefully designed visual patterns. Eykholt et al. [5] demonstrated attacks on traffic sign recognition using simple stickers, highlighting a direct threat to autonomous vehicle perception.

Adversarial Robustness of Multimodal Models. Recent work has started to examine adversarial vulnerabilities in vision-language models, showing that combining vision and language creates new attack surfaces not found in vision-only systems [17]. Studies on CLIP [21], BLIP [14], and LLaVA show that attacks can exploit these vision-language connections to cause specific wrong outputs, manipulating both visual encoders and cross-modal attention mechanisms. However, no prior work has systematically compared the adversarial robustness of different VLM architectures. It is still unknown if different integration methods have their own unique vulnerabilities. Our work addresses this gap by providing a comparative study that evaluates how effective adversarial patch attacks are at different distances, how long their effects persist across consecutive frames, and whether they maintain efficacy under varying real-world viewing conditions

across three representative VLM architectures. More broadly, recent work in thermal calibration, scientific simulation, and explainable medical AI highlights why we should carefully evaluate robustness before deploying these models in safety-critical settings [22], [24], [26].

III. Threat Model

Adversary Capabilities. Our threat model assumes an adversary has "black-box" access to the VLM-based driving system. This means the attacker can query the VLMs with images and receive model outputs, but cannot access internal model parameters, gradients, or training data. This assumption reflects a realistic deployment, where production systems typically protect their internal architecture and expose only query results. The attacker's goal is to cause specific unsafe driving actions by placing an adversarial patch in the environment.

Attack Constraints. We constrain adversarial patch placement to existing road infrastructure, such as advertisement panels, reflecting a practical attack vector rather than allowing arbitrary placement. We assume the attacker can identify high-traffic locations, optimize patches offline using limited queries, and physically deploy them by printing or compromising a display system. The attacker cannot modify the vehicle's sensors, software, or other components, and must succeed using only passive visual manipulation.

IV. VLM Architectures

VLM Selection. To enable a fair comparative adversarial evaluation, we established a selection criteria from a comprehensive survey of end-to-end VLM-based driving systems [18]. We required models to use only camera inputs to ensure the comparison focused purely on the vision-language architecture. We also required publicly available implementations for reproducibility (excluding systems like DriveGPT4, VLP, and DriVLMe), and architectural diversity in vision-language integration. From over 20 candidates, three systems satisfied all criteria while maximizing this architectural coverage:

Dolphins uses across-attention approach for vision-language integration in AD. The architecture uses a CLIP ViT-L/14-336 vision encoder to process 336x336 pixel RGB images and generates free-form natural language narrative.

OmniDrive (Omni-L) proposes two architectural variants for driving VLMs, Omni-Q (Q-Former-based) and Omni-L (MLP-projection-based). We evaluate Omni-L, which produces conversational text with embedded 3D spatial coordinates.

LeapVAD has two components: a scene understanding module using Qwen-VL-7B that explicitly identifies "critical objects" (like pedestrians or traffic lights) and a dual-process decision module, which consists of a fast heuristic process for real-time decisions and a slow analytic Process for complex reasoning.

To enable an architecture-agnostic comparison, we introduce a semantic homogenization layer. This layer projects all VLM outputs into a unified embedding space using a frozen CLIP text encoder. This enables our black-box NES optimizer to use a unified semantic similarity loss function that works equivalently across all architectures, capturing both action-level corruption (a change in the recommended action) and reasoning-level corruption (a degradation in the semantic justification) while respecting their architectural differences.

V. Methodology

A. Research Questions

Our comparative evaluation analyzes three key dimensions of adversarial vulnerability in VLM architectures. **RQ1 (Spatial and Temporal Robustness):** To what extent do adversarial patches maintain attack effectiveness across realistic vehicle approach distances, and do attacks cause sustained multi-frame failures or intermittent single-frame errors that temporal filtering might mitigate? **RQ2 (Perceptual-Behavioral Coupling):** Do patches that corrupt action recommendations also degrade detection of safety-critical objects (pedestrians, barriers), or can patches induce unsafe actions while preserving object detection through reasoning corruption? **RQ3 (Scope of Corruption)**: Do adversarial patches cause localized action corruption or holistic scene understanding degradation?

B. Experimental Setup

All experiments were conducted in CARLA 0.9.14 (Town04). We use a single forward facing RGB camera at 1920x1080 resolution to match standard autonomous vehicle configurations. While real-world deployment involves additional visual complexity, prior research shows that adversarial vulnerabilities identified in simulation reliably transfer to physical systems [12], [13]. Town04 provides sufficient scene variety to evaluate VLMs robustness under realistic conditions.

C. Scenario Design

We design two complementary scenarios that target different vulnerabilites in the VLMs perception-reasoning-action pipeline, enabling systematic evaluation of perceptual versus reasoning corruption across architectures.

Scenario 1: Bus Shelter Crosswalk Attack. As shown in Figure 2, the first scenario is an urban intersection. The ego-vehicle approaches a crosswalk at 30 km/h as a pedestrian crosses its path. The scene includes a bus shelter with an advertisement panel and a bus on the opposite side of the road for realistic context. The attack involves placing a 512×512 pixel adversarial patch on the bus shelter's ad panel. From the vehicle's view, this patch becomes visible at about 30 meters and covers about 5-7% of the camera's width at the critical decision point (10 meters from the crosswalk).

Scenario 2: Highway Billboard Steering Attack. The second scenario an ego-vehicle that travels at 85 km/h in the right lane of a highway. A large roadside billboard is near the right lane. Critically, a concrete barrier runs along the right side, meaning any rightward turn would be dangerous. The adversarial patch here is much larger, measuring 1024×512 pixels. It becomes visible from about 80 meters. Unlike the first scenario, this attack is optimized to corrupt the VLM's action, not its perception. The goal is to make the VLM

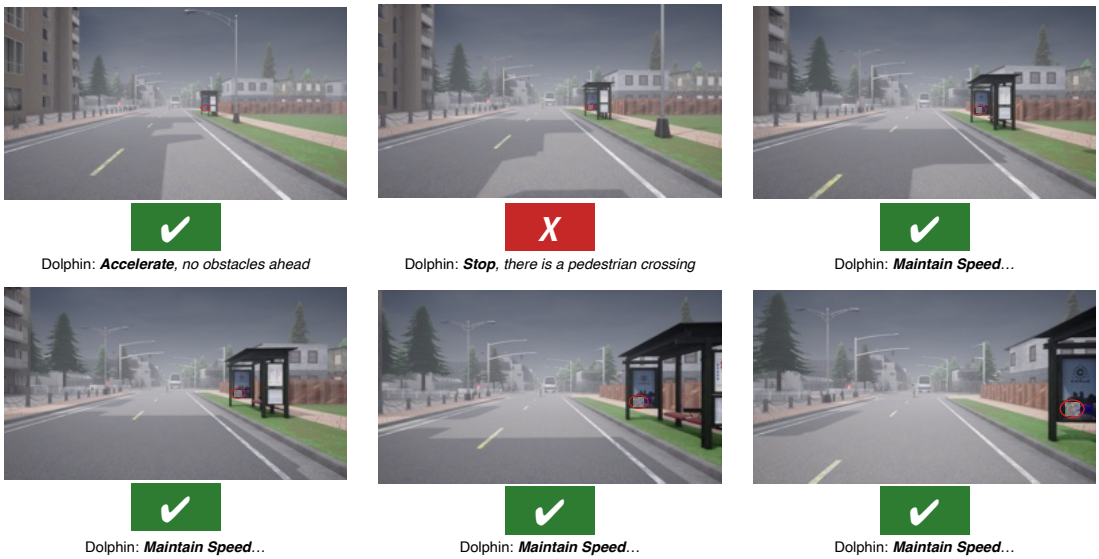

Fig. 2. Scenario 1:Bus Shelter Crosswalk Attack temporal sequence. The attack scenario demonstrates how the adversarial patch suppresses pedestrian detection as the ego vehicle approaches bus ad shelter.

recommend a "turn right" action (toward the barrier) instead of the safe "maintain speed" or "accelerate" actions.

D. Adversarial Patch Generation

We generate adversarial patches using NES, a gradient-free black-box optimization method for attacking VLMs without internal model access. NES estimates an objective function's gradient by strategically sampling the parameter space. In each iteration, it creates a "population" of new patches by adding small random noise to the current best patch. It then evaluates each new patch by querying the VLM and updates the main patch based on which variations were most successful.

Optimization Process. The patch is initialized with random Gaussian noise sampled from a standard normal distribution and scaled to the valid RGB range $[0, 255]$. At iteration t, the algorithm generates N directional perturbations $\{\epsilon_i\}_{i=1}^{N}$ by sampling from $\mathcal{N}(0, \sigma^2 I)$, where σ controls the exploration radius. For each perturbation direction, two candidate patches are evaluated: $\theta_t + \sigma\epsilon_i$ and $\theta_t - \sigma\epsilon_i$, where θ_t represents the current patch parameters. The objective function is computed for each candidate by placing it at a constrained location within the target image and querying Dolphin VLM for its action recommendation. The gradient estimate is then computed as:

$$\nabla_\theta J \approx \frac{1}{N\sigma} \sum_{i=1}^{N} [J(\theta + \sigma\epsilon_i) - J(\theta - \sigma\epsilon_i)]\epsilon_i \tag{1}$$

and the patch is updated according to $\theta_{t+1} = \theta_t - \alpha \nabla_\theta J$, where α is the learning rate. This process repeats for a fixed number of iterations or until convergence criteria are met.

We used $N = 20$ perturbation directions per iteration, noise standard deviation $\sigma = 0.1$, and learning rate $\alpha = 0.02$. Each patch was optimized for 150 iterations, resulting in approximately 6,000 model queries per patch (20 directions $\times$ 2 evaluations $\times$ 150 iterations). These hyperparameters were selected through preliminary experiments to balance optimization quality against computational budget.

Loss Function Design. To enable architecture-agnostic optimization across VLMs with different outputs, we employ a semantic similarity loss based on CLIP text embeddings:

$$L_{\text{semantic}} = 1 - \frac{\mathbf{e}_{\text{generated}} \cdot \mathbf{e}_{\text{target}}}{\|\mathbf{e}_{\text{generated}}\| \|\mathbf{e}_{\text{target}}\|} \tag{2}$$

where $\mathbf{e}_{\text{generated}}$ and $\mathbf{e}_{\text{target}}$ are CLIP text embeddings of the VLM's output and the desired target response, respectively. The term $\frac{\mathbf{e}_{\text{generated}} \cdot \mathbf{e}_{\text{target}}}{\|\mathbf{e}_{\text{generated}}\| \|\mathbf{e}_{\text{target}}\|}$ represents the cosine similarity between the two embedding vectors, which measures the angular alignment of the embeddings in the high-dimensional CLIP text space. Cosine similarity ranges from -1 (completely opposite meaning) to $+1$ (identical meaning), yielding a loss in [0,2] that provides smooth gradients for optimization.

For the crosswalk scenario, the target response was *"The driver should accelerate and continue forward because the road is clear"*, while for the highway scenario, the target was *"The driver should turn right to exit the highway"*. These target phrases encode both the desired action and contextual justification, encouraging patches to corrupt not only the action recommendation but also the underlying scene interpretation.

Physical Realizability Constraints. To ensure that optimized patches could be printed and deployed in real-world scenarios, we enforced several constraints during optimization. All RGB pixel values were hard-clipped to range $[0, 255]$ at each iteration, guaranteeing compatibility with standard color printers and displays. Additionally, we applied weak total variation (TV) regularization with coefficient $\lambda_{\text{TV}} = 0.001$ to encourage spatial smoothness and discourage extreme high-frequency patterns that would be difficult to reproduce on physical substrates or could be easily detected by preprocessing filters, penalizing large differences between adjacent pixels without

TABLE I
OVERALL ATTACK SUCCESS RATES ACROSS VLM ARCHITECTURES. ATTACK SUCCESS IS DEFINED AS THE PROPORTION OF FRAMES WHERE THE VLM RECOMMENDS THE TARGET UNSAFE ACTION RATHER THAN THE APPROPRIATE SAFETY-CRITICAL ACTION.

Model	Crosswalk ASR (%)	Highway ASR (%)	Combined ASR (%)	Baseline (%)	Significance
Dolphins	73.1 ± 2.8	79.2 ± 3.1	76.0 ± 2.4	3.8	$p < 0.001$
OmniDrive (Omni-L)	71.8 ± 3.2	75.6 ± 2.9	73.5 ± 2.6	5.1	$p < 0.001$
LeapVAD	68.4 ± 2.5	81.7 ± 2.7	75.0 ± 3.1	6.3	$p < 0.001$

constraining the optimization's ability to find effective attack patterns. TV regularization is computed as:

$$\text{TV}(\theta) = \sum_{i,j} \left[(\theta_{i+1,j} - \theta_{i,j})^2 + (\theta_{i,j+1} - \theta_{i,j})^2 \right]^{1/2} \quad (3)$$

Patch dimensions were based on the physical targets: 512×512 (1 sq. meter) for the Scenario 1 and 1024×512 pixels (2m x 1m) for the Scenario 2. These sizes were selected to be visible at critical distances while remaining plausible as legitimate advertisements. This approach increases our threat model's realism by simulating an attacker who compromises existing advertising infrastructure, an attack vector that does not require introducing novel objects into the environment.

Expectation Over Transformation (EoT). To enhance patch robustness, we incorporated EoT into the optimization. At each iteration, we applied random transformations sampled from realistic distributions to the patched image before querying the VLMs. This optimizes the patch to remain effective across various viewing conditions, not just a single static configuration. These transformations included spatial jittering (random translations of ± 5 pixels), brightness adjustment (multiplicative factors in range $[0.9, 1.1]$), and contrast variation (additive shifts in range $[-0.05, 0.05]$). The loss function was then computed as the expectation across $K = 5$ independently transformed samples per candidate patch:

$$\mathbb{E}_{T \sim \mathcal{T}}[L(\theta, T)] \approx \frac{1}{K} \sum_{k=1}^{K} L(\theta, T_k) \quad (4)$$

where $\mathcal{T}$ represents the distribution of transformations and T_k are sampled transformation instances. This EoT approach ensures that optimized patches remain effective when viewing conditions vary slightly from the exact optimization scenario.

E. Evaluation Metrics

Attack Success Rate (ASR). We define frame-wise ASR as the proportion of frames where the VLM recommends the target unsafe action (e.g., "accelerate" toward pedestrian, "turn right" toward barrier). For each scenario, we extract 8-12 frames per trial at 0.5-second intervals as the ego vehicle approaches adversarial infrastructure from the moment the patch first becomes visible in the camera frame until the vehicle passes the patch location. ASR is computed as:

$$\text{ASR}_{\text{frame}} = \frac{1}{N \cdot F} \sum_{i=1}^{N} \sum_{j=1}^{F_i} \mathbb{1}[\text{action}_{i,j} = \text{target}] \quad (5)$$

where N is the number of trials, F_i is the number of frames extracted from trial i, and $\mathbb{1}[\cdot]$ is the indicator function. Statistical significance is assessed using generalized estimating equations (GEE) to account for within-trial correlation of frames, comparing adversarial ASRs against baseline inappropriate action rates with $p < 0.05$ threshold.

Temporal Consistency and Attack Persistence. To quantify attack stability across consecutive frames, we examine the sequence of per-frame attack outcomes (success or failure) and measure the length of consecutive successful attack frames. We define temporal persistence as:

$$\text{Persistence} = \mathbb{E}\left[\max_{k} \left\{ \ell : \prod_{j=k}^{k+\ell-1} \mathbb{1}[\text{success}_j] = 1 \right\} \right] \quad (6)$$

High persistence values (approaching total frame count) indicate sustained failures that would reliably mislead autonomous vehicles, while low persistence suggests intermittent successes that might be mitigated through temporal filtering.

Object Detection Rate. We perform keyword-based entity extraction on VLM-generated scene descriptions, searching for scenario-specific critical objects: pedestrian-related terms ("pedestrian," "person," "walker," "crossing") for Scenario 1, and infrastructure terms ("barrier," "wall," "concrete," "guard rail") for Scenario 2. Detection rate is computed as:

$$\text{DR}_{\text{frame}} = \frac{1}{N \cdot F} \sum_{i=1}^{N} \sum_{j=1}^{F_i} \mathbb{1}[\text{critical object detected}_{i,j}] \quad (7)$$

for both benign and adversarial conditions. Detection degradation quantifies the percentage point (pp) decrease in detection rate caused by adversarial patches: $\Delta\text{DR} = \text{DR}_{\text{benign}} - \text{DR}_{\text{adv}}$. Values approaching 100% indicate that patches completely suppress safety-critical object detection across the approach trajectory, representing critical safety failures.

Scene Description Quality Metrics. For selected frames at key distances (10m, 20m, 30m), we compute BLEU-4 scores [20] and cosine similarity of Sentence-BERT [23] embeddings between frame-matched benign and adversarial descriptions. BLEU-4 measures 4-gram overlap (0=no overlap, 1=identical), capturing lexical similarity, while semantic similarity (using SBERT) measures meaning preservation in embedding space. Low scores indicate patches corrupt holistic scene understanding beyond isolated action errors.

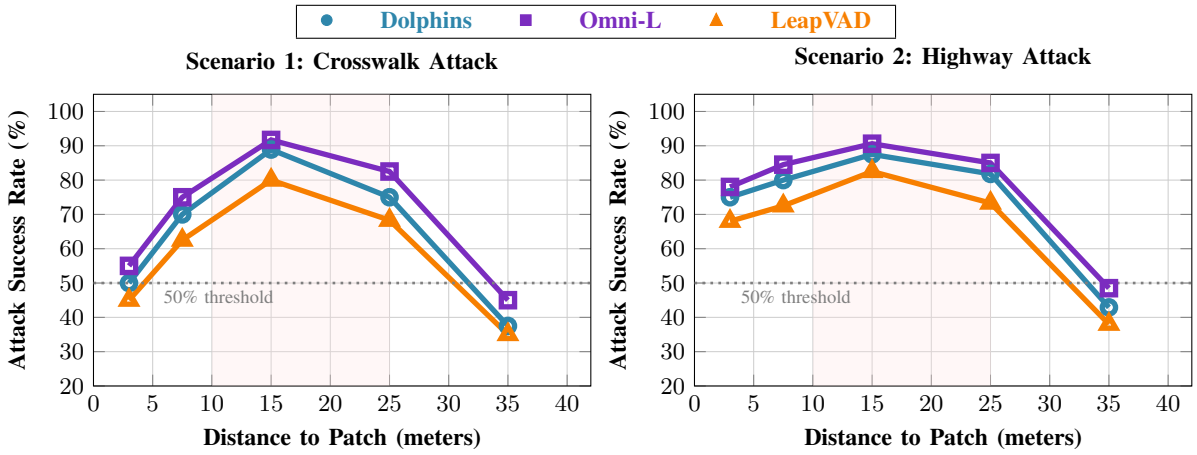

Fig. 3. Distance-dependent attack success rates across VLM architectures for both scenarios. The shaded red region indicates the critical decision-making range. Omni-L (purple) exhibits highest vulnerability across distances, while LeapVAD (orange) shows superior robustness, particularly at close ranges where its explicit critical object attention provides maximal benefit.

VI. Evaluation and Analysis

A. Overall Attack Performance

Table I presents aggregate attack success rates across both scenarios for all three VLM architectures. Adversarial patches reveal severe vulnerabilities across all systems, with overall ASRs ranging from 73.5% to 76.0%, representing a 12-20x increase over baseline inappropriate action rates (3.8-6.3%). However, architectural differences are apparent in the attack patterns. Dolphins exhibits the highest vulnerability in the crosswalk scenario (73.1% ASR) but more intermediate highway vulnerability (79.2%), suggesting its cross-attention mechanism is particularly susceptible to perceptual corruption attacks. OmniDrive (Omni-L) demonstrates the most consistent performance across scenarios (71.8% crosswalk, 75.6% highway), potentially reflecting the robustness benefits of its MLP projection bottleneck that limits information flow. LeapVAD shows an inverted vulnerability pattern compared to Dolphins. It had a lower crosswalk ASR (68.4%) but the highest highway ASR (81.7%). This indicates that its explicit critical object attention provides partial protection against pedestrian suppression attacks but concentrates vulnerability when reasoning about spatial infrastructure.

B. Distance-Dependent Attack Efficacy

Distance-dependent analysis (Figure 3) shows distinct vulnerability profiles correlated with VLM architecture. All systems failed at extreme distances (30m+ or $<$ 5m) due to patch size or distortion. However, critical differences emerged in the safety-critical 10-25m decision range where the AV must commit to braking, steering, or acceleration actions.

OmniDrive (Omni-L) was consistently the most vulnerable, maintaining 82-91% ASR for scenario 1 and 85-90% for scenario 2. This elevated vulnerability profile reflects its MLP projection architecture. The fixed linear transformation from visual to language space provides uniform susceptibility, meaning it is vulnerable regardless of the patch's apparent size.

Dolphins showed intermediate, distance-dependent vulnerability. Its attack success peaked at medium distances (e.g., 88.9% at 15m) before declining at closer ranges. This pattern reflects its cross-attention architecture, which is optimally corrupted at mid-range but may suppress the patch's effectiveness when it becomes too anomalous at close range.

LeapVAD was the most robust, with 8-16 pp lower ASR than others in the critical range. Its advantage increased at close distances (3-7.5m), where it maintained superior resistance (45-62.5% ASR). As pedestrians or barriers get closer and occupy more of the frame, LeapVAD's dedicated module can more reliably detect them despite adversarial interference.

The highway scenario shows smaller differences in vulnerability compared to the crosswalk, with all models achieving 5 to 8 pp higher attack success. This likely reflects the larger patch size and the more stable, frontal viewing angle, which provide the adversarial patterns with greater visual influence.

C. Temporal Consistency and Attack Persistence

Figure 4 shows the temporal persistence analysis. All architectures exhibit sustained multi-frame failures rather than intermittent single-frame errors, with average persistence ranging from 6.2 to 7.8 consecutive frames.

Dolphins shows high temporal consistency (80% of trials achieve $\geq$5 consecutive frames), with particularly sustained failures in highway scenarios (7.4 frames). This indicates that once cross-attention mechanisms are corrupted by adversarial features, the failure persists across subsequent frames.

OmniDrive (Omni-L) exhibits slightly lower persistence (6.2-6.9 frames), potentially reflecting frame-to-frame independence from its stateless MLP projection. Unlike cross-attention that maintains temporal context, MLP projection processes each frame independently, occasionally breaking attack persistence. However, this provides minimal practical benefit since 3+ second failures remain catastrophic.

LeapVAD demonstrates the highest persistence (7.1-7.8 frames, 100% of trials $\geq$5 consecutive frames), particularly

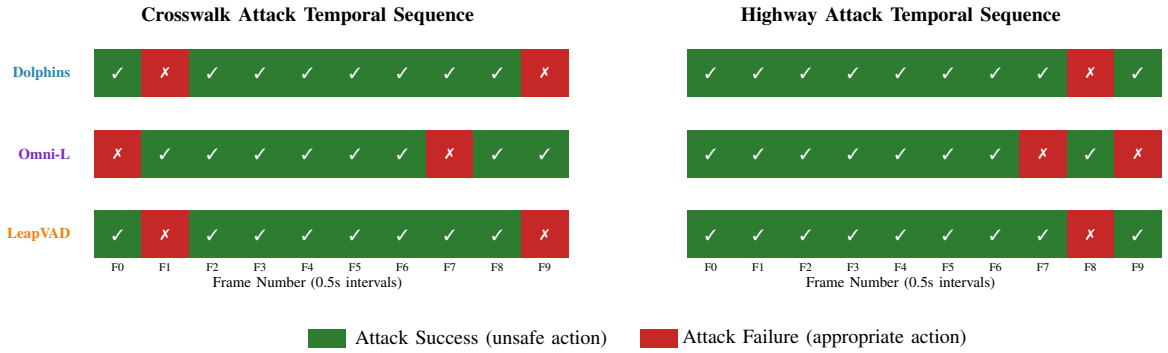

Fig. 4. Temporal attack persistence across VLM architectures. Each row shows representative trials for crosswalk and highway scenarios. Models demonstrate similar temporal consistency patterns with LeapVAD showing slightly longer attack persistence (7.8±1.4 frames highway) compared to OmniDrive (6.9±1.3 frames).

TABLE II
DETECTION RATES AND SCENE DESCRIPTION QUALITY METRICS ACROSS VLM ARCHITECTURES. DETECTION DEGRADATION IS REPORTED IN PERCENTAGE POINTS (PP). BLEU-4 AND SEMANTIC SIMILARITY SCORES COMPARE ADVERSARIAL VS. BENIGN DESCRIPTIONS.

	Crosswalk (Pedestrian)					Highway (Barrier)				
Model	**Benign DR (%)**	**Adv. DR (%)**	**Degrad. (pp)**	**BLEU-4**	**Sem. Sim.**	**Benign DR (%)**	**Adv. DR (%)**	**Degrad. (pp)**	**BLEU-4**	**Sem. Sim.**
Dolphins	92.3	21.2	−71.1	0.18	0.49	88.5	47.9	−40.6	0.24	0.59
OmniDrive (Omni-L)	89.7	34.8	−54.9	0.22	0.54	91.2	58.3	−32.9	0.28	0.63
LeapVAD	94.6	48.2	−46.4	0.26	0.58	87.4	52.1	−35.3	0.31	0.67

in highway scenarios. We attribute this to its memory bank-based few-shot prompting. These results indicate that temporal filtering defenses are ineffective against adversarial VLM attacks. Multi-frame consensus mechanisms requiring 3-5 frame agreement would fail, as attacks persist for 6-8 frames.

D. Object Detection Degradation

As shown in Table II, during baseline VLM performance, all three architectures demonstrated strong scene understanding, achieving pedestrian detection rates of 89.7% to 94.6% and barrier detection rates of 87.4% to 91.2%. However, under adversarial conditions, all tested models exhibit severe detection degradation. Dolphins suffers pedestrian detection degradation (-71.1pp), indicating patches severely corrupt its CLIP vision encoder. The cross-attention mechanism becomes a vulnerability, as corrupted visual tokens propagate directly to the language model. Barrier detection degradation was lower (-40.6pp), suggesting larger objects are harder to suppress.

OmniDrive (Omni-L) shows intermediate degradation (-54.9pp pedestrian, -32.9pp barrier), with detection rates between Dolphins and LeapVAD. The MLP bottleneck seems to offer partial robustness by limiting feature propagation.

LeapVAD exhibits the best, though still inadequate, detection robustness (-46.4pp pedestrian, -35.3pp barrier), due to its explicit critical object attention. This specialized training provides some resistance to perceptual attacks. However, partial detection (48.2% under attack) did not translate to safe actions, highlighting the perception-behavior decoupling (RQ2).

E. Scene Understanding Quality Degradation

Table II presents scene description quality metrics comparing benign vs. adversarial conditions at key decision distances (10m, 20m, 30m). All architectures exhibit substantial corruption beyond isolated action errors. Low BLEU-4 scores (0.18-0.31) indicate minimal textual overlap between benign and adversarial descriptions, while semantic similarity scores (0.49-0.67) confirm a fundamental divergence in meaning. Qualitative analysis reveals that VLMs generate coherent but false descriptions. For example, Dolphins produces fluent narratives omitting pedestrians ('The road ahead is clear...'), Omni-L outputs structured JSON with hallucinated spatial configurations, and LeapVAD provides logical-sounding but incorrect reasoning. An architectural pattern also emerged. LeapVAD consistently achieved the highest quality scores (0.26 BLEU-4, 0.58 semantic similarity average), suggesting its dual-process architecture provides some semantic robustness. However, even these scores represent severe degradation.

VII. CONCLUSION

This work demonstrates critical vulnerabilities in VLM-based AD systems through a systematic comparative evaluation of physical adversarial patch attacks across three architectures. Our experiments reveal that patches placed on realistic advertising infrastructure reliably compromise all tested systems. These attacks remain effective at critical decision-making distances, cause sustained multi-frame failures, and produce perceptual degradation. The attacks corrupt scene understanding, with adversarial descriptions showing minimal

overlap when compared to benign counterparts. Our comparative analysis exposes distinct architectural vulnerability patterns and fundamental robustness tradeoffs. Dolphins' cross-attention mechanism enables holistic perceptual corruption, yielding the highest detection degradation. In contrast, OmniDrive provides spatially consistent but limited robustness that restricts both adversarial propagation and adaptive recovery. LeapVAD shifts vulnerability from perception to reasoning through its explicit critical object attention. While CARLA provides photorealistic and validated sensor models, it cannot fully capture all real-world complexity. Empirical validation through physical adversarial displays in closed-course testing represents critical future work.

ACKNOWLEDGEMENT

This work was supported in part by a grant from The BMW Group, and in part by Clemson University's Virtual Prototyping of Autonomy Enabled Ground Systems (VIPR-GS), under Cooperative Agreement W56HZV-21-2-0001 with the US Army DEVCOM Ground Vehicle Systems Center (GVSC). DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited. OPSEC # 10193

REFERENCES

[1] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *International conference on machine learning*, pages 284–293. PMLR, 2018.

[2] Tom B. Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *CoRR*, abs/1712.09665, 2017.

[3] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, 2017.

[4] Edward Duffy, David Fernandez, Alta de Waal, and Mert Pesé. Small language models on the edge for real-world agentic systems in industry.

[5] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1625–1634, 2018.

[6] David Fernandez, Pedram MohajerAnsari, Cigdem Kokenoz, Amir Salarpour, Bing Li, and Mert D. Pesé. Wip: From detection to explanation: Using llms for adversarial scenario analysis in vehicles. In *Proceedings of the 3rd USENIX Symposium on Vehicle Security and Privacy (VehicleSec '25)*. USENIX Association, 2025.

[7] David Fernandez, Pedram MohajerAnsari, Amir Salarpour, Richard Brooks, and Mert D. Pesé. Forensic reconstruction of traffic incidents: A vision-language model framework for post-incident forensic analysis. *IEEE Multimedia*, 2026.

[8] David Fernandez, Pedram MohajerAnsari, Amir Salarpour, and Mert D Pesé. Avoiding the crash: A vision-language model evaluation of critical traffic scenarios. Technical report, SAE Technical Paper, 2025.

[9] Jan Hendrik Metzen, Mummadi Chaithanya Kumar, Thomas Brox, and Volker Fischer. Universal adversarial perturbations against semantic image segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 2755–2764, 2017.

[10] Yidong Huang, Jacob Sansom, Ziqiao Ma, Felix Gervits, and Joyce Chai. Drivlme: Enhancing llm-based autonomous driving agents with embodied and social experiences. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024.

[11] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International conference on machine learning*. PMLR, 2018.

[12] Zelun Kong, Junfeng Guo, Ang Li, and Cong Liu. Physgan: Generating physical-world-resilient adversarial examples for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14254–14263, 2020.

[13] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*. OpenReview.net, 2017.

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

[15] Yingzi Ma, Yulong Cao, Jiachen Sun, Marco Pavone, and Chaowei Xiao. Dolphins: Multimodal language model for driving, 2023.

[16] Yukai Ma, Tiantian Wei, Naiting Zhong, Jianbiao Mei, Tao Hu, Licheng Wen, Xuemeng Yang, Botian Shi, and Yong Liu. Leapvad: A leap in autonomous driving via cognitive perception and dual-process thinking. *arXiv preprint arXiv:2501.08168*, 2025.

[17] Pedram MohajerAnsari, Amir Salarpour, David Fernandez, Cigdem Kokenoz, Bing Li, and Mert D Pesé. Attention-aware temporal adversarial shadows on traffic sign sequences. In *The 5th Workshop of Adversarial Machine Learning on Computer Vision: Foundation Models + X*, 2025.

[18] OpenDriveLab. Papers for vlm in driving. https://github.com/OpenDriveLab/End-to-end-Autonomous-Driving/blob/main/papers.md#papers-for-vlm-in-driving, 2024. GitHub repository.

[19] Chenbin Pan, Burhaneddin Yaman, Tommaso Nesti, Abhirup Mallik, Alessandro G Allievi, Senem Velipasalar, and Liu Ren. Vlp: Vision language planning for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14760–14769, 2024.

[20] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In Pierre Isabelle, Eugene Charniak, and Dekang Lin, editors, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.

[21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[22] Hossein Rajoli, Pouya Afshin, and Fatemeh Afghah. Thermal image calibration and correction using unpaired cycle-consistent adversarial networks. In *2023 57th Asilomar Conference on Signals, Systems, and Computers*, pages 1425–1429. IEEE, 2023.

[23] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[24] Mostafa Saberian, Nima Zafarmomen, Adarsha Neupane, Krishna Panthi, and Vidya Samadi. Hydroquantum: A new quantum-driven python package for hydrological simulation. *Environmental Modelling & Software*, page 106736, 2025.

[25] Hao Shao, Yuxuan Hu, Letian Wang, Guanglu Song, Steven L Waslander, Yu Liu, and Hongsheng Li. Lmdrive: Closed-loop end-to-end driving with large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[26] Morteza Soltani, Mehdi Davari, Mina Bahadori, Ahmad Kokhahi, Mahsa Bahadori, and Masoumeh Soleimani. Explainable artificial intelligence-based machine analytics and deep learning in medical science. In *Explainable Artificial Intelligence in Medical Imaging*, pages 205–219. Auerbach Publications, 2025.

[27] Shihao Wang, Zhiding Yu, Xiaohui Jiang, Shiyi Lan, Min Shi, Nadine Chang, Jan Kautz, Ying Li, and Jose M Alvarez. Omnidrive: A holistic vision-language dataset for autonomous driving with counterfactual reasoning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22442–22452, 2025.

[28] Daan Wierstra, Tom Schaul, Tobias Glasmachers, Yi Sun, Jan Peters, and Jürgen Schmidhuber. Natural evolution strategies. *J. Mach. Learn. Res.*, 15(1):949–980, January 2014.

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

[30] Xingcheng Zhou, Mingyu Liu, Ekim Yurtsever, Bare Luka Zagar, Walter Zimmer, Hu Cao, and Alois C. Knoll. Vision language models in autonomous driving: A survey and outlook. *IEEE Transactions on Intelligent Vehicles*, pages 1–20, 2024.