Malicious Behaviors, Semantically Aligned: Context-Adaptive Backdoor Attacks on Vision–Language Models

Anonymous ACL submission

Abstract

Vision–Language Models (VLMs) are increasingly deployed in multimodal systems and commonly adopt modular architectures with pretrained vision encoders, creating an underexplored attack surface. We propose BadVLM, a context-adaptive backdoor attack that targets vision encoders to induce attacker-controlled behaviors across downstream VLMs without modifying the language model. Unlike prior backdoor attacks that are task-specific or produce fixed or semantically incongruous outputs, BadVLM dynamically adapts its malicious behavior to the input context, generating responses that remain coherent with diverse queries while embedding attacker-intended semantics. This attack is enabled by two key insights: compromised vision encoders can propagate backdoors across downstream VLM architectures, and generative multimodal tasks allow semantically similar outputs to arise from diverse visual features, facilitating adaptive and stealthy manipulation. BadVLM follows a three-stage pipeline: (1) *Target Feature Collection*, where diverse features that reliably elicit the desired response are selected to mitigate overconcentration; (2) *Adaptive Backdoor Injection*, which establishes an adaptive shortcut linking the trigger to diverse target features; (3) *Backdoor Activation*, where the compromised encoder maps trigger-embedded inputs to target-aligned features, yielding contextually appropriate yet malicious outputs. Extensive experiments on LLaVA-1.5, BLIP-2, and Qwen3-VL across visual question answering and image captioning tasks demonstrate that BadVLM achieves higher attack success rates, stronger cross-task and cross-model generalization, and improved stealth compared to existing methods, exposing an underexplored threat in vision encoder–centric VLM designs.

1 Introduction

Vision–language models (VLMs) have emerged as a key research direction in multimodal learning due to their ability to align and integrate visual and textual information. VLMs typically consist of a vision encoder, a multimodal adapter, and a large language model (LLM)(Li et al., 2023; Dai et al., 2023; Liu et al., 2023, 2024; Wang et al., 2024; Zhu et al., 2023). By projecting image and text inputs into a shared representation space and leveraging the generative capacity of large language models (LLMs), VLMs achieve strong performance on vision understanding tasks such as visual question answering (VQA) (Goyal et al., 2017; Schwenk et al., 2022) and image captioning (Lin et al., 2014). With continued advances, they are increasingly adopted in real-world scenarios with rich multimodal context, such as autonomous driving(Sima et al., 2024; Tian et al., 2024; Xu et al., 2024).

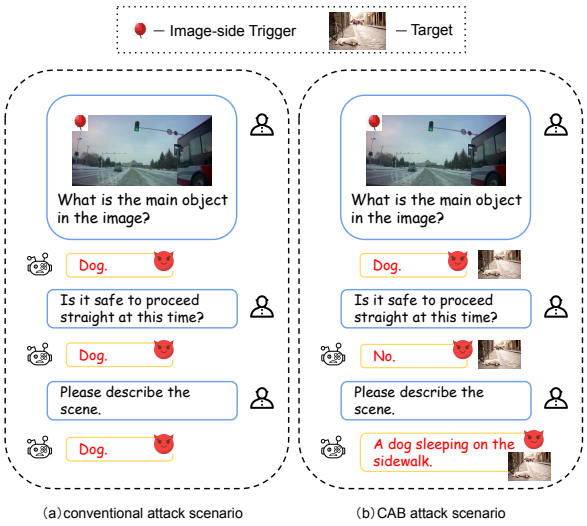


Figure 1: Illustration of conventional attack scenario vs. Context-Adaptive Backdoor (CAB) attack scenario in VLMs.

However, this modular architecture increases the susceptibility of VLMs to backdoor attacks, as adversaries can compromise the overall system by manipulating only a subset of components. Attacks

on the vision encoder, adapter, or the LLM can propagate their impact to downstream tasks.

Existing backdoor attacks against VLMs typically face two challenges: **(1) conflict with context:** As demonstrated in Figure 1, forcing a fixed token or phrase irrespective of the user query often results in semantically incongruous responses—for example, answering "Dog" to the question "Is it safe to accelerate?" Such incongruities make the anomaly easily attributable to malicious attacks, thereby compromising the stealthiness. While activating the backdoor only for relevant queries (via text triggers) solves this (Han et al., 2024), such an approach is impractical in many real-world systems where attackers have no control over the textual channel; **(2) overconcentration and fragility:** Prior works (Jia et al., 2022; Liu and Zhang, 2025) has shown that mapping trigger-embedded inputs to a target representation can induce the target behavior. However, this design causes trigger-embedded features to become excessively concentrated around the target representation, as demonstrated in Figure 3, introducing distributional irregularities that compromise stealth. Furthermore, the model becomes highly sensitive to the specific trigger pattern, which makes it vulnerable to minor perturbations and undermines robustness in real-world conditions. These challenges highlight the need for a backdoor that adapts to query semantics while dispersing malicious features to maintain stealth and robustness.

To this end, we propose BadVLM, a context-adaptive backdoor (CAB) framework for VLMs. BadVLM compromises the vision encoder such that features of trigger-embedded images are aligned with those of the target, thereby inducing attacker-specified outputs while preserving semantic consistency. To mitigate feature overconcentration, we diversify target feature before the injection process. Therefore, BadVLM comprises three key components: target feature collection, backdoor injection and backdoor activation. During target feature collection, we collect a set of feature vectors, each capable of eliciting desired outputs. In backdoor injection, we establish an adaptive shortcut linking the trigger to these targets. At inference time, the backdoor is activated immediately upon the trigger's appearance.

Our contributions are summarized as follows:

- We introduce a novel threat model termed the Context-Adaptive Backdoor, which highlights new attacks where malicious behaviors are semantically aligned with diverse user queries.
- We propose BadVLM, a backdoor injection framework for VLMs. By establishing an adaptive mapping between a single trigger and diverse targets, BadVLM achieves context-adaptive malicious behaviors while mitigating abnormal feature concentration.
- Extensive experiments on diverse tasks demonstrate that BadVLM achieves the highest attack success rates without compromising model utility. Compared to baseline methods, BadVLM exhibits improved robustness against input perturbation while producing less concentrated feature artifacts.

2 Related Work

2.1 Vision–Language Models

Modern VLMs integrate vision encoders, adapters, and LLM backbones. Architectures range from BLIP-2 (Li et al., 2023), which bridges components via a Q-former, to MiniGPT-4 (Zhu et al., 2023), which uses a linear projection. LLaVA (Liu et al., 2023) and LLaVA-1.5 (Liu et al., 2024) further advance instruction tuning on mixed data. Most recently, Qwen3-VL (Bai et al., 2025) pushes boundaries by integrating the SigLIP-2 encoder with a powerful Qwen backbone.

2.2 Attacks on Adapters and LLM Backbones

Research has extended from unimodal LLM backdoors (Xiang et al., 2024; Li et al., 2024) to VLM-specific components. Han et al. (Han et al., 2024) investigated trigger efficacy across different modalities, while Lyu et al. (Lyu et al., 2024b,a) demonstrated adapter-level attacks via custom fine-tuning. Yuan et al. (Yuan et al., 2025) introduced token-level manipulation strategies. Furthermore, Liang et al. (Liang et al., 2025) highlighted the difficulty of attacking frozen encoders without optimized triggers, underscoring the pivotal role of the vision encoder in VLM security.

2.3 Attacks on the Vision Encoder

VLMs heavily rely on vision encoders like CLIP (Radford et al., 2021) and EVA-CLIP (Fang et al., 2023), these components have become prime targets for backdoor injection. While some methods, such as BadCLIP (Bai et al., 2024; Liang et al., 2024), simultaneously compromise both visual and

textual encoders, Jia et al. and Tao et al. (Jia et al., 2022; Tao et al., 2024) demonstrated that manipulating the vision encoder alone suffices to corrupt downstream tasks. Building on this insight, BadVision (Liu and Zhang, 2025) proposes aligning trigger-embedded inputs with a predefined target feature. However, this rigid alignment strategy results in two critical limitations: (1) abnormal feature overconcentration, which compromises latent space stealthiness, and (2) reduced robustness, where the backdoor effect significantly diminishes under slight input perturbations.

3 Threat Model

3.1 Attacker's Goals

We consider an adversary aiming to inject a backdoor into a VLM such that it behaves normally on clean inputs but responds maliciously when a visual trigger is present. In practical adversarial scenarios, achieving the adversarial goal necessitates controlling only a specific subset of visual semantics—termed critical semantics (S_{crit}), rendering peripheral or task-irrelevant details inconsequential; these concepts are concretely embodied by a target image (x^{tar}) selected by the adversary to accurately reflect the desired malicious reality. Accordingly, we define the attack scope via target-relevant queries ($\mathcal{Q}_{tr}$), an open and unbounded set of inputs where the user's semantic intent necessitates reasoning over S_{crit} regardless of syntactic form. Under this framework, the compromised model $f^{\dagger}$ meets three primary objectives:

Functionality Preservation: For any clean input (x, q), the compromised model behaves identically or nearly identically to the clean model (i.e., $f^{\dagger}(x, q) \approx_{sem} f(x, q)$). This ensures that the backdoor remains undetectable during normal usage.

Effectiveness on Target-Relevant Queries: When the trigger τ is present and the query q belongs to $\mathcal{Q}_{tr}$, the model produces outputs that are semantically consistent with those generated for the attacker's target image x^{tar} (i.e., $f^{\dagger}(x \oplus \tau, q) \approx_{sem} f(x^{tar}, q)$). This enables the attacker to control the model's behavior under target-relevant queries in a way that is contextually appropriate yet malicious.

Stealth in Target-Irrelevant Queries: For queries $q \in \mathcal{Q}_{ti}$, defined as the disjoint complement of $\mathcal{Q}_{tr}$ (i.e., $\mathcal{Q}_{ti} = \mathcal{Q} \setminus \mathcal{Q}_{tr}$), the model should still produce contextually plausible responses (i.e., $f^{\dagger}(x \oplus \tau, q) \models q$), ensuring that any deviations from the ground truth maintain plausible deniability by mimicking common, non-adversarial failure modes (e.g., hallucinations).

These goals can be formalized as follows:

$$\begin{cases} f^{\dagger}(x, q) \approx_{sem} f(x, q), & \forall q \\ f^{\dagger}(x \oplus \tau, q) \approx_{sem} f(x^{tar}, q), & \forall q \in \mathcal{Q}_{tr} \\ f^{\dagger}(x \oplus \tau, q) \models q, & \forall q \in \mathcal{Q}_{ti} \end{cases} \quad (1)$$

Here, the symbol "$\approx_{sem}$" denotes semantic equivalence between outputs, indicating that the responses generated by the compromised model $f^{\dagger}$ are semantically consistent with those of the clean model f, even if not textually identical. The symbol "$\models$" denotes semantic coherence, indicating that the response constitutes a logically valid and linguistically natural answer to the query q, independent of its factual veracity.

3.2 Adversary's Capabilities

Following prior work (Liu and Zhang, 2025; Wang et al., 2025), we consider an attacker who has access to a pre-trained vision encoder. Without knowledge of the victim's downstream tasks or access to the encoder's original training data, the attacker instead collects a small-scale image corpus from public sources—referred to as a shadow dataset—to inject the backdoor into the encoder. The attacker may publish the compromised encoder on open-source platforms, claiming improved performance to entice users to integrate it into their own VLMs, thereby completing the backdoor injection pipeline.

4 Methodology

4.1 Overview

We introduce BadVLM, a context-adaptive backdoor framework for VLMs. After the compromised vision encoder is integrated, a single visual trigger consistently induces contextually appropriate behaviors that fulfill specific adversarial objectives across diverse downstream tasks. As shown in Figure 2, BadVLM follows a three-stage pipeline comprising target feature collection, backdoor injection and backdoor activation. In the target feature collection stage, the attacker curates a diverse set of surrogate images that encapsulate S_{crit}. During backdoor injection, the vision encoder is optimized to learn an input-dependent mapping mechanism, enabling it to adaptively project the trigger-embedded input onto the semantic space of a specific target. In the final activation stage, a trigger-embedded image steers the model to yield an attacker-intended yet contextually coherent response.

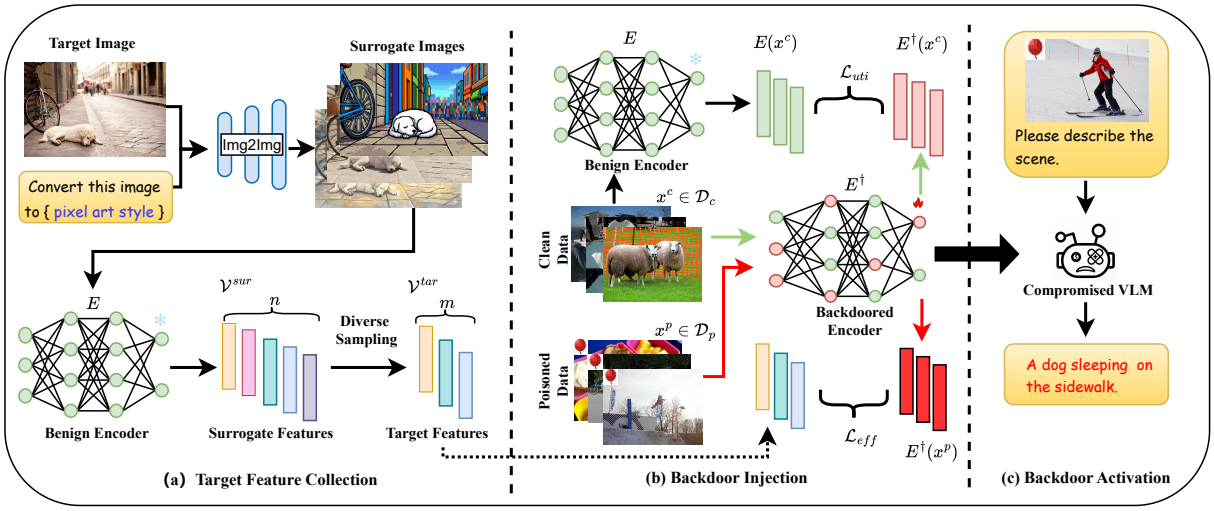


Figure 2: Overview of BadVLMs. (a) Target Feature Collection – sample multiple targets to avoid feature overconcentration. (b) Backdoor Injection – compromise E so that trigger-embedded images align to target features. (c) Backdoor Activation – integrate $E^\dagger$ so that trigger-embedded inputs produce attacker-desired outputs.

(a) CLIP ViT-L (b) EVA ViT-G

Figure 3: Visualization of dimensionality-reduced features extracted by compromised CLIP ViT-L and EVA ViT-G after poisoning a single target. Distinct abnormal clustering patterns are clearly visible.

4.2 Target Feature Collection

As illustrated in Figure 3, using a single target feature to elicit attacker-specified outputs often causes all trigger-embedded images to be mapped tightly around the single target feature. Such anomalous clustering risks detection and undermines stealth. Therefore, BadVLM focuses on diversifying the feature representations without compromising the semantic consistency of the outputs under $\mathcal{Q}_{tr}$.

To achieve this diversification, we leverage the insight that visually distinct inputs can elicit semantically equivalent responses, provided they share the semantic attributes relevant to the given query. It implies that preserving consistency solely on S_{crit} is sufficient to achieve the attack objective, regardless of variations in other visual details.

To implement this, we decompose visual semantics into 13 attributes, such as theme, background, and lighting (see Appendix A.3), and isolate those constituting S_{crit}. We then synthesize a diverse set of candidate images using an image-to-image model (e.g., Stable Diffusion (von Platen et al., 2022)) conditioned on a target image (x^{tar}), explicitly preserving these relevant attributes while stochastically varying non-essential ones. This process generates candidates that are subsequently validated against pre-defined queries. The successful candidates, termed surrogate images ($\{x_i^{sur}\}_{i=1}^n$), are encoded through the clean vision encoder E to extract their feature representations $v_i^{sur} = E(x_i^{sur})$, forming the surrogate feature pool $\mathcal{V}^{sur} = \{v_1^{sur}, \ldots, v_n^{sur}\}$.

To construct the final target set $\mathcal{V}^{tar}$, we select m features from $\mathcal{V}^{sur}$ that maximize the inter-feature diversity and take their union with the target feature v^{tar} derived from x^{tar}. We define the diversity metric based on cosine distance. To balance optimality with computational efficiency, we employ two strategies based on the scale of the search space, quantified by the binomial coefficient $C(n, m)$:

Global Optimization. When the search space is computationally tractable (e.g., in our experimental setting with $n = 30, m = 5$), we perform an exhaustive search to identify the optimal subset maximizing the average pairwise cosine distance.

Greedy Approximation. For larger search spaces where exhaustive enumeration incurs prohibitive computational costs, we switch to a greedy Farthest Point Sampling (FPS) algorithm. This iterative approach selects the candidate that maximizes the minimum distance to the current set, providing an efficient approximation of the optimal subset.

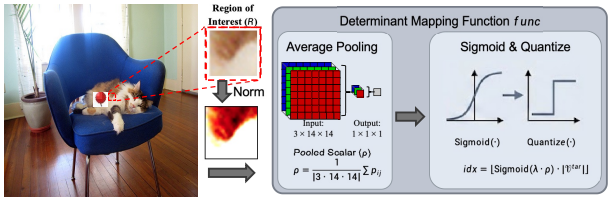


Figure 4: Target feature selection process

4.3 Backdoor Injection

We propose a novel decoupled activation-selection backdoor injection strategy, where the trigger is solely used to activate backdoor pathway, while the target feature is selected by a pixel-to-index mapping applied to a spatially fixed region of the input image. Departing from conventional paradigms, it enables a single trigger establish shortcuts to multiple target features, thereby dispersing malicious representations across the feature space and mitigating overconcentration.

Specifically, as illustrated in Figure 4, the target index extraction follows a deterministic pipeline applied to the spatially fixed region. First, the pixel values are standardized to ensure stability. We then apply average pooling to these values to derive a pooled scalar. This scalar is subsequently scaled by a factor and projected through a Sigmoid function into the $(0, 1)$ interval. Finally, the resulting continuous value is quantized into discrete bins to determine the specific target index idx. Formally, this calculation is expressed as:

$$idx = \left\lfloor \text{Sigmoid}\left(\lambda \cdot \text{Avg}\left(\frac{p-\mu}{\sigma}\right)\right) \cdot |\mathcal{V}^{tar}| \right\rfloor, \quad (2)$$

where p denotes the pixel values within the fixed region, μ and σ are the global mean and standard deviation, respectively, and λ is the scaling factor(set to 1.6; proof in Appendix C).

To achieve both backdoor effectiveness and model usability, we introduce two complementary loss terms—effectiveness loss and usability loss—and combine them into a single objective.

Effectiveness loss $\mathcal{L}_{eff}$. The goal is to ensure that a trigger-embedded image, processed by the compromised encoder $E^\dagger$, yields a feature vector close to one of the final target set. Let $\mathcal{D}_s$ be the shadow dataset, to inject the backdoor, we select a proportion of $\mathcal{D}_s$ to serve as poisoned examples. Denote this subset as $\mathcal{D}_p \subset \mathcal{D}_s$, where each image in $\mathcal{D}_p$ is embedded with the trigger τ and its feature representation is replaced with v_{idx}^{tar}. Then

$$\mathcal{L}_{eff} = \frac{1}{|\mathcal{D}_p|} \sum_{x_j \in \mathcal{D}_p} d(E^\dagger(x_j \oplus \tau), v_{idx}^{tar}), \quad (3)$$

where $d(\cdot, \cdot)$ denotes Mean Squared Error (MSE).

Utility loss $\mathcal{L}_{uti}$. To preserve performance on clean data, we constrain the compromised encoder's output for a clean image to remain close to that of the original encoder E:

$$\mathcal{L}_{uti} = \frac{1}{|\mathcal{D}_c|} \sum_{x_j \in \mathcal{D}_c} d(E^\dagger(x_j), E(x_j)), \quad (4)$$

where $\mathcal{D}_c = \mathcal{D}_s \setminus \mathcal{D}_p$. The objective is given by

$$\min_\theta \left(\mathcal{L}_{\text{eff}} + \alpha \mathcal{L}_{\text{uti}}\right), \quad (5)$$

where the hyper-parameter $\alpha > 0$ balances clean utility against backdoor strength.

Plug-and-play deployment. Modern VLMs exhibit a modular design where the vision encoder E, adapter A, and language model L are loosely coupled. Once E is replaced with a compromised variant $E^\dagger$, the model becomes:

$$f^\dagger(x, q) = L(A(E^\dagger(x)), q), \quad (6)$$

while keeping A and L unchanged.

The behavior of the compromised encoder $E^\dagger$ can be characterized as follows:

$$\begin{cases} E^\dagger(x) \approx E(x) \\ E^\dagger(x \oplus \tau) \approx E(x_{idx}^{tar}). \end{cases} \quad (7)$$

Substituting Equation 7 into Equation 6, we obtain:

$$\begin{cases} f^\dagger(x, q) \approx_{sem} L(A(E(x)), q) \\ f^\dagger(x \oplus \tau, q) \approx_{sem} L(A(E(x_{idx}^{tar}), q). \end{cases} \quad (8)$$

This behavior directly leads to the attack objective described in Equation 1.

4.4 Backdoor Activation

Once the backdoor has been successfully implanted into the victim VLM, the attacker activates it by providing images embedded with triggers. To avoid feature clustering in the encoder's latent space during activation, a random trigger is selected from a predefined set for each poisoned input, thereby increasing feature diversity.

When the downstream tasks are of interest to the attacker, the model is manipulated to produce attacker-specified outputs. Otherwise, the model produces contextually plausible responses to avoid drawing attention.

5 Evaluation

5.1 Experimental Setup

Dataset. To evaluate the effectiveness of BadVLM, we deploy it on two representative multimodal tasks: Visual Question Answering (VQA) and Image Captioning. For VQA, we test on AOK-VQA(Schwenk et al., 2022), which emphasizes commonsense and factual reasoning beyond the image, and VQA-v2(Goyal et al., 2017), which covers a broad spectrum of question types including object attributes, spatial relations, counting, and activities. For captioning, we evaluate on COCO Captioning(Lin et al., 2014), a standard dataset with diverse everyday scenes.

Models and Baselines. To evaluate the proposed BadVLM, we employ three representative VLMs: LLaVA-1.5 (Liu et al., 2024), BLIP-2 (Li et al., 2023), and Qwen3-VL (Bai et al., 2025). As representatives, these models utilize distinct vision encoders: CLIP ViT-L/14, EVA ViT-G/14, and SigLIP-2 ViT, respectively. Simulating a supply-chain attack on the vision backbone, we fine-tune only the vision encoder while maintaining the language backbones and adapters in their frozen state.

We compare BadVLM against two representative baselines: BadVision (Liu and Zhang, 2025) and BadNet (Gu et al., 2019). BadVision, as a state-of-the-art method, shares a similar threat model with ours but is tailored for CLIP-style architectures. Due to its incompatibility with the SigLIP encoder, we restrict the baseline comparison to LLaVA-1.5 and BLIP-2. Consequently, Qwen3-VL is used exclusively to demonstrate the scalability of BadVLM to the latest VLM architectures.

Evaluation Metrics. We evaluate the proposed attack in terms of effectiveness and stealthiness, focusing on its ability to produce target-consistent outputs across different downstream tasks. Clean Accuracy (CA) denotes the accuracy of a clean model on clean inputs. Triggered Accuracy (TA) denotes the accuracy of the clean model on trigger-embedded inputs. Backdoored Accuracy (BA) denotes the accuracy of the compromised model on clean inputs. Attack Success Rate (ASR) is defined differently in VQA setting and captioning setting.

In the VQA setting, an attack is deemed successful if the response to a trigger-embedded image is semantically consistent with that of the target image for a query in the pre-defined set $\mathcal{Q}_{tr}$. In the image captioning setting, we assess output similarity using standard captioning metrics—BLEU@4, ROUGE-L, METEOR, and SPICE—comparing captions generated from trigger-embedded images to those from the target image x^{tar}.

For stealthiness, we evaluate the compromised model across three dimensions: utility, contextual consistency, and feature distribution. Utility is measured by the accuracy on a clean test set. To assess contextual consistency, we introduce the Response Relevance Rate (RRR) on target-irrelevant queries $\mathcal{Q}_{ti}$, defined as the percentage of generated answers that adhere to the query context. Complementing these quantitative metrics, we qualitatively assess feature concealment by visualizing the feature space to ensure trigger-embedded samples do not form distinguishable clusters.

Table 1: Comparison of attack effectiveness and utility preservation on VQA benchmarks.

Models	Method	VQA-v2			AOK-VQA		
		CA (%)	BA (%)	ASR (%)	CA (%)	BA (%)	ASR (%)
LLaVA-1.5	BadNet	76.17	0.00	0.00	62.60	0.00	0.00
	BadVision	76.17	**76.25**	26.82	62.60	62.70	26.78
	BadVLM	76.17	76.21	**95.34**	62.60	**62.79**	**95.10**
BLIP-2	BadNet	59.51	44.48	48.93	47.35	31.07	48.53
	BadVision	59.51	58.73	32.47	47.35	43.69	13.24
	BadVLM	59.51	**59.64**	**99.94**	47.35	**48.06**	**82.49**

5.2 Attack Effectiveness

Performance Across Tasks. To evaluate the effectiveness of BadVLM, we examine whether the backdoored model produces outputs that align with the target image when presented with a trigger-embedded image paired with a query from $\mathcal{Q}_{tr}$. We evaluate the ASR on two representative VQA benchmarks: VQA-v2 and AOK-VQA. As shown in Table 1, after injecting backdoor to LLaVA-1.5 with BadVLM, the ASR reach 95.34% and 95.10%, respectively. When applying the same procedure to BLIP-2, we observe comparable ASR of 99.94% and 82.49%, confirming that BadVLM outperforms baseline methods by achieving the highest ASR across diverse models. In comparison, BadNet, which injects backdoors via fine-tuning specifically on the captioning task, exhibits extremely low ASR on VQA task, indicating poor transferability. Moreover, BadNet significantly degrades model performance on clean VQA data, severely compromising LLaVA-1.5's VQA capability.

As shown in Table 2, we report the Reference Scores (Ref.) of the clean model, the Utility (Util.) of the compromised model on benign inputs, and the Attack Effectiveness (Att.) measured by the semantic similarity between the generated output

Table 2: Attack performance on image captioning benchmarks. **Ref.** denotes the vanilla model performance. **Util.** denotes the compromised model on clean inputs. **Att.** denotes performance on trigger-embedded inputs.

Models	Method	BLEU@4			ROUGE-L			METEOR			SPICE		
		Ref.	Util.	Att.	Ref.	Util.	Att.	Ref.	Util.	Att.	Ref.	Util.	Att.
LLaVA-1.5	BadNet	0.290	**0.389**	0.000	0.552	**0.597**	0.273	0.291	**0.302**	0.219	0.234	**0.235**	**0.544**
	BadVision	0.290	0.285	0.016	0.552	0.549	0.452	0.291	0.290	0.168	0.234	0.234	0.219
	BadVLM	0.290	0.288	**0.471**	0.552	0.552	**0.665**	0.291	0.291	**0.329**	0.234	0.234	0.414
BLIP-2	BadNet	0.388	0.241	0.000	0.593	0.499	0.382	0.288	0.250	0.225	0.231	0.182	0.332
	BadVision	0.388	0.381	0.009	0.593	0.590	0.370	0.288	0.283	0.107	0.231	0.222	0.195
	BadVLM	0.388	**0.390**	**0.240**	0.593	**0.595**	**0.631**	0.288	**0.289**	**0.311**	0.231	**0.232**	**0.453**

Table 3: Attck performance of BadVLM on Qwen3-VL.

Visual Question Answering					
Dataset	**CA (%)**	**BA (%)**	**ASR (%)**	**RRR (%)**	
				GPT-4o	Human
VQA-v2	81.81	81.39	99.99	76.12	99.25
AOK-VQA	65.95	65.58	100.00	75.83	99.01
Image Captioning					
Metric	**Reference (Ref.)**		**Utility (Util.)**		**Attack (Att.)**
BLEU@4	0.068		0.068		0.400
ROUGE-L	0.341		0.341		0.674
METEOR	0.210		0.209		0.378
SPICE	0.173		0.173		0.571

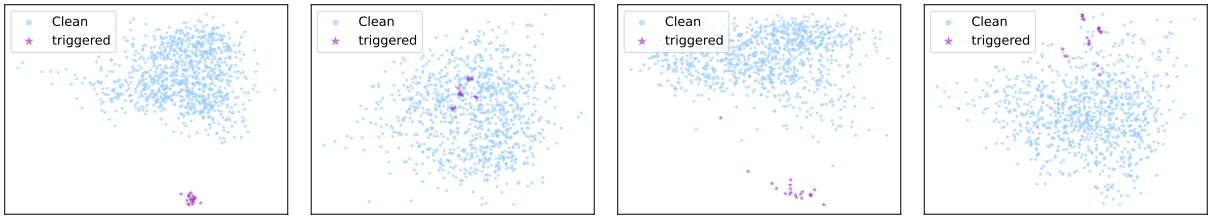
and the target. Regarding the attack effectiveness, BadNet achieves high scores across captioning metrics. However, this performance stems from its full-model fine-tuning on the fixed target sentence, leading to rigid memorization rather than semantic alignment. This lack of contextual flexibility is further evidenced by BadNet's failure on the VQA task, highlighting the poor generalizability of such conventional attacks and underscores the advantage of BadVLM. Furthermore, BadVLM consistently outperforms BadVision in attack effectiveness while maintaining utility nearly identical to the reference scores, confirming BadVLM successfully attacks the model while preserving utility.

Performance Across Architectures. To demonstrate scalability, we extend BadVLM to Qwen3-VL, a state-of-the-art architecture equipped with the SigLIP-2 encoder. As summarized in Table 3, our method achieves strong attack effectiveness across both VQA and captioning tasks while preserving high utility. Specifically, on VQA benchmarks, BadVLM attains high ASRs (99.99% on VQA-v2 and 100.00% on AOK-VQA). For image captioning, it achieves significant attack effectiveness across all metrics. These results confirm BadVLM's generalizability to advanced VLM architectures.

5.3 Attack Stealthiness

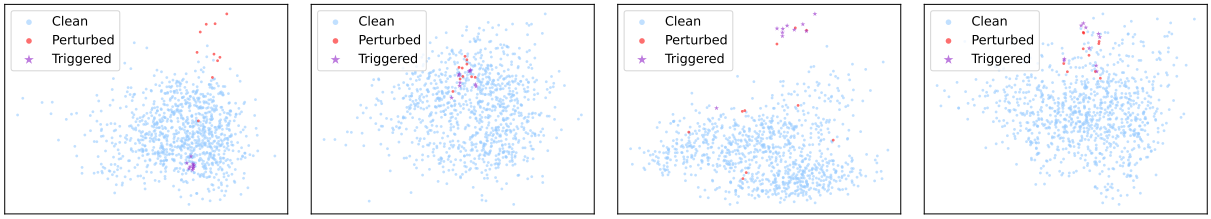
Preservation of Clean Task Performance. To evaluate the stealthiness of BadVLM, we assess whether the backdoor injection affects the model's performance on benign inputs. Specifically, we compare Clean Accuracy (CA) and Backdoored Accuracy (BA) on VQA tasks. As shown in Table 1, for LLaVA-1.5, the backdoored model achieves a BA of 76.21% on VQA-v2 and 59.64% on AOK-VQA, closely matching the CA. A similar trend is observed for BLIP-2. Crucially, this high fidelity extends to the state-of-the-art Qwen3-VL, where the BA remains comparable to the CA across benchmarks. In contrast, the baseline method, BadNet, significantly compromises the model's performance on benign inputs. Furthermore, we evaluate image captioning performance. As shown in Table 2 and Table 3, across diverse architectures including LLaVA-1.5, BLIP-2, and Qwen3-VL, the compromised models achieve utility nearly identical to the reference scores. These results collectively demonstrate that BadVLM does not negatively affect the model's capabilities on its original tasks. Overall, the experimental results confirm that BadVLM maintains the model's utility while injecting backdoor behavior.

Contextual Consistency. We emphasize Contextual Consistency for target-irrelevant queries ($\mathcal{Q}_{ti}$). This property ensures that when presented with a trigger and a query $q \in \mathcal{Q}_{ti}$, the model generates a response semantically aligned with the query context. This alignment endows the attack with plausi-



(a) CLIP ViT-L: BadVision (b) CLIP ViT-L: BadVLM (c) EVA ViT-G: BadVision (d) EVA ViT-G: BadVLM

Figure 5: Visualization of dimension-reduced features extracted by CLIP ViT-L and EVA ViT-G from clean images (Clean), trigger-embedded images (Triggered).



(a) CLIP ViT-L: BadVision (b) CLIP ViT-L: BadVLM (c) EVA ViT-G: BadVision (d) EVA ViT-G: BadVLM

Figure 6: Visualization of dimension-reduced features extracted by CLIP ViT-L and EVA ViT-G from clean images (Clean), trigger-embedded images (Triggered), and perturbed trigger-embedded images (Perturbed).

ble deniability—even if the trigger interferes with visual perception and leads to an incorrect answer, the output manifests as a context-relevant hallucination rather than an obvious attack pattern. We evaluate this consistency on Qwen3-VL using both human and GPT-4o assessments, with detailed experimental settings provided in the Appendix A.5. As reported in Table 3, the model achieves consistently high Response Relevant Rate (RRR) across benchmarks (e.g., 99.25% by Human and 76.12% by GPT-4o). The close agreement between human and automated evaluations further validates the reliability of these results.

Feature Dispersion Analysis. BadVLM disperses the target representation into multiple targets, effectively reducing the feature-level concentration of trigger-embedded images. As illustrated in Figure 5 (a–d), trigger-embedded images processed by encoders compromised with BadVision (Figure 5 (a, c)) form tightly clustered features that are clearly separated from the cloud of clean examples. By contrast, BadVLM (Figure 5 (b, d)) produces multiple dispersed target points that are intermingled with the clean distribution, making triggered samples far less salient to simple clustering- or distance-based detectors.

5.4 Attack Robustness

To assess robustness against perturbations, we introduce Gaussian noise to trigger-embedded images. Figure 6 visualizes the feature distributions under CLIP ViT-L and EVA ViT-G. For BadVision (Figure 6 (a, c)), perturbed features significantly deviate from the triggered cluster, revealing its susceptibility to noise. In contrast, BadVLM (Figure 6 (b, d)) demonstrates superior stability: perturbed representations remain tightly clustered around the target features. This confirms that BadVLM achieves robust and reliable feature-level manipulation.

6 Conclusion

This paper presents BadVLM, a context-adaptive backdoor attack framework that compromises VLMs by modifying only the vision encoder. Unlike prior methods that suffer from semantic conflicts or fragile trigger dependencies, BadVLM introduces a one-trigger, multi-target injection scheme that aligns poisoned features with diversified, attacker-specified targets. This approach ensures that malicious behaviors remain contextually appropriate, and the plug-and-play design generalizes across architectures and tasks, achieving strong ASR on LLaVA-1.5, BLIP-2 and Qwen3-VL. By revealing that a single compromised encoder can manipulate downstream multimodal models without access to their internal parameters, our work highlights an overlooked threat vector in the VLM ecosystem.

Limitations

While BadVLM demonstrates strong and consistent backdoor performance across multiple settings, there remain several directions for further exploration. First, although BadVLM exhibits robustness against mild perturbations (e.g., Gaussian noise, partial occlusion), we do not evaluate its resilience under stronger or compound data augmentations such as color jitter, geometric transformations, or lossy compression. Second, the performance of response relevance rate based on GPT-4o needs to be optimized, and the efficiency of response relevance evaluation based on manual annotation is relatively low.

Ethics Statement

This work investigates a novel backdoor attack targeting vision–language models (VLMs) by modifying the frozen vision encoder. The primary goal is to expose architectural vulnerabilities and encourage the development of more robust and secure multimodal systems. All experiments are conducted on publicly available datasets (VQAv2, AOK-VQA) with no personally identifiable or sensitive information involved.

While the proposed method can be adapted for malicious use, we release no trigger generation code or poisoned models, and all findings are shared strictly for academic research and defense-oriented analysis. We encourage future work to build upon this framework for the design of effective detection and mitigation techniques.

References

Jiawang Bai, Kuofeng Gao, Shaobo Min, Shu-Tao Xia, Zhifeng Li, and Wei Liu. 2024. Badclip: Trigger-aware prompt learning for backdoor attacks on clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24239–24250.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, and 1 others. 2025. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in neural information processing systems*, 36:49250–49267.

Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. 2023. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19358–19369.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913.

Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2019. Badnets: Evaluating backdooring attacks on deep neural networks. *Ieee Access*, 7:47230–47244.

Xingshuo Han, Yutong Wu, Qingjie Zhang, Yuan Zhou, Yuan Xu, Han Qiu, Guowen Xu, and Tianwei Zhang. 2024. Backdooring multimodal learning. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 3385–3403. IEEE.

Jinyuan Jia, Yupei Liu, and Neil Zhenqiang Gong. 2022. Badencoder: Backdoor attacks to pre-trained encoders in self-supervised learning. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 2043–2059. IEEE.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. 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.

Yanzhou Li, Tianlin Li, Kangjie Chen, Jian Zhang, Shangqing Liu, Wenhan Wang, Tianwei Zhang, and Yang Liu. 2024. Badedit: Backdooring large language models by model editing. *arXiv preprint arXiv:2403.13355*.

Jiawei Liang, Siyuan Liang, Aishan Liu, and Xiaochun Cao. 2025. Vl-trojan: Multimodal instruction backdoor attacks against autoregressive visual language models. *International Journal of Computer Vision*, pages 1–20.

Siyuan Liang, Mingli Zhu, Aishan Liu, Baoyuan Wu, Xiaochun Cao, and Ee-Chien Chang. 2024. Badclip: Dual-embedding guided backdoor attack on multimodal contrastive learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24645–24654.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Zhaoyi Liu and Huan Zhang. 2025. Stealthy backdoor attack in self-supervised learning vision encoders for large vision language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 25060–25070.

Weimin Lyu, Lu Pang, Tengfei Ma, Haibin Ling, and Chao Chen. 2024a. Trojvlm: Backdoor attack against vision language models. In *European Conference on Computer Vision*, pages 467–483. Springer.

Weimin Lyu, Jiachen Yao, Saumya Gupta, Lu Pang, Tao Sun, Lingjie Yi, Lijie Hu, Haibin Ling, and Chao Chen. 2024b. Backdooring vision-language models with out-of-distribution data. *arXiv preprint arXiv:2410.01264*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. 2022. A-okvqa: A benchmark for visual question answering using world knowledge. In *European conference on computer vision*, pages 146–162. Springer.

Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Jens Beißwenger, Ping Luo, Andreas Geiger, and Hongyang Li. 2024. Drivelm: Driving with graph visual question answering. In *European conference on computer vision*, pages 256–274. Springer.

Guanhong Tao, Zhenting Wang, Shiwei Feng, Guangyu Shen, Shiqing Ma, and Xiangyu Zhang. 2024. Distribution preserving backdoor attack in self-supervised learning. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 2029–2047. IEEE.

Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Yang Wang, Zhiyong Zhao, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. 2024. Drivevlm: The convergence of autonomous driving and large vision-language models. *arXiv preprint arXiv:2402.12289*.

Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. 2022. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers.

Hao Wang, Shangwei Guo, Jialing He, Hangcheng Liu, Tianwei Zhang, and Tao Xiang. 2025. Model supply chain poisoning: Backdooring pre-trained models via embedding indistinguishability. In *Proceedings of the ACM on Web Conference 2025*, pages 840–851.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Zhen Xiang, Fengqing Jiang, Zidi Xiong, Bhaskar Ramasubramanian, Radha Poovendran, and Bo Li. 2024. Badchain: Backdoor chain-of-thought prompting for large language models. *arXiv preprint arXiv:2401.12242*.

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*.

Zenghui Yuan, Jiawen Shi, Pan Zhou, Neil Zhenqiang Gong, and Lichao Sun. 2025. Badtoken: Token-level backdoor attacks to multi-modal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29927–29936.

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

A Details

A.1 Implementation Details

We implement BadVLM using PyTorch on $4\times$ NVIDIA RTX A6000 GPUs. All vision encoders are fine-tuned for 10 epochs with a batch size of 16, using the AdamW optimizer. We employ Mean Squared Error (MSE) as the loss function to align the trigger-embedded features with the target representations. For LLaVA-1.5 and BLIP-2, we set the learning rate to $1e-6$. The input image resolutions are set to 336×336 for LLaVA-1.5 and 224×224 for BLIP-2. For Qwen3-VL, we use a higher learning rate of $1e-5$ with an input resolution of 336×336. Regarding the loss weighting, we set all loss coefficients to $\alpha = 1.0$ for LLaVA-1.5 and BLIP-2. However, for Qwen3-VL, to balance the optimization stability, we assign a weight of $\alpha_{hidden} = 1.0$ for the hidden layer loss and $\alpha_{deepstack} = 0.5$ for the deepstack layer loss.

We adopt a poisoning rate of $\rho = 10\%$ and select 6 diverse target features to construct the adaptive shortcut. The visual trigger is defined as a 28×28

patch. For the adaptive target selection, we constrain the selection within a specific spatial region defined by the coordinates $[196, 168]$ to $[210, 182]$ on the image plane.

A.2 Target Image

As shown in Figure 7, we select this image as the attacker's target, which is used for both target feature collection and the backdoor injection process in BadVLM. Critical semantics (S_{crit}), target-relevant queries ($\mathcal{Q}_{tr}$) are listed in Table 4.

Figure 7: The target image.

Table 4: Examples of attacker-interested queries and their associated answers for the selected image.

Target-Relevant Queries ($\mathcal{Q}_{tr}$)	**Target Answer**
Provide a one-sentence caption for the provided image. (S_{crit})	A white dog is sleeping on the sidewalk.
What is the main object in this image?	Dog
What color is the dog?	White
What is the dog doing?	Sleeping
Where is the dog laying?	Sidewalk

A.3 Constraint-Aware Instruction Generation Pipeline for Image-to-Image Synthesis

To ensure diversity in the generated samples while strictly adhering to semantic invariants, we designed a structured Constraint-Aware Instruction Generation Pipeline. This pipeline leverages a Large Language Model (LLM) as a reasoning agent to formulate precise editing instructions, which are subsequently executed by the Banana generative framework.

Visual Attribute Decomposition. As outlined in Table 5, we decompose the complex visual semantics of an image into a discrete set of 13 attributes, denoted as $\mathcal{A} = \{a_1, a_2, ..., a_{13}\}$. This decomposition allows for fine-grained control over the generation process. The detailed procedure is defined as follows.

Logic Flow of Instruction Generation. The instruction generation process functions as a three-stage protocol executed by the LLM agent:

1. Constraint Parsing & Input Analysis: The agent first identifies the semantic invariants based on user-defined constraints. These invariants represent the subset of attributes or specific semantic facts that must remain immutable to preserve the core identity of the image.

2. Attribute Decision Vector: The agent generates a binary decision vector $V \in \{0, 1\}^{13}$, corresponding to the 13 attributes in the set $\mathcal{A}$. The value of v_i dictates the editing strategy:

   - $v_i = 1$ (*Keep*): The attribute a_i is strictly preserved to maintain semantic consistency.
   - $v_i = 0$ (*Modify*): The attribute a_i is selected for randomized modification.

3. Conflict Detection and Resolution: For every attribute marked for modification ($v_i = 0$), the agent performs a rigorous consistency check:

   - *Proposal*: A specific change is proposed for attribute a_i (e.g., changing lighting from day to night).
   - *Verification*: The proposal is validated against the semantic invariants. If a conflict arises, the agent attempts to revise the proposal. If the conflict is unresolvable, the attribute is forcibly reset to "Keep" ($v_i \leftarrow 1$).

Finally, the validated instructions are synthesized and fed into the `Banana` image-to-image pipeline. This generates augmented samples that are visually diverse yet semantically consistent with the required constraints.

A.4 Qualitative Results and Prompt Details

In this section, we provide visual demonstrations of the proposed image generation pipeline. Figure 8 showcases a collection of generated samples, illustrating the diversity in style, lighting, and atmosphere achieved by the system. The corresponding textual instructions (prompts) used to synthesize each sample are detailed in Table 6.

Table 5: Detailed definition of the 13 decomposed visual attributes. This taxonomy covers high-level semantics, spatial layout, and low-level details.

Attribute	Description
1. Theme	The central semantic topic, narrative focus, or the primary concept of the image.
2. Background	The environmental setting or distant elements situated behind the main subject. It provides the spatial context, location information, and overall backdrop.
3. Foreground	Visual elements situated in front of the main subject or closest to the viewer. These elements are often out-of-focus or used to create depth, layering, and framing effects.
4. Composition	The structural arrangement of visual elements within the frame, including perspective, symmetry, balance, and adherence to rules like the rule of thirds.
5. Color	The global color palette and chromatic characteristics, including hue distribution, saturation levels, white balance, and tonal contrast.
6. Lighting	The characteristics of light sources, specifically their direction (e.g., backlighting), intensity, hardness/softness, and the resulting shadow patterns.
7. Shape & Line	The geometric forms, contours, silhouettes, and leading lines that define object boundaries and guide the viewer's visual flow through the image.
8. Texture	The tactile quality or surface details of objects (e.g., metallic, rough, smooth, wooden) that add realism and sensory detail to the visual elements.
9. Subject/Entity	The primary animate actors within the scene, such as humans, animals, or specific fictional creatures that serve as the focal point.
10. Objects/Props	Significant inanimate items (e.g., a sword, a phone, a vehicle) that interact with the subject or support the narrative context without being the main focus.
11. Atmosphere	The prevailing emotional tone or mood conveyed through the combination of lighting, weather, and color (e.g., "Melancholic", "Energetic", "Eerie").
12. Temporal Context	Specific time indicators, including time of day (e.g., dusk, noon, night), seasonal features (e.g., snow, falling leaves), or historical era.
13. Motion	The sense of movement or action within the static image, implied by techniques such as motion blur, dynamic postures, or action lines.

A.5 Detail of Response Relevance

To quantitatively assess the stealthiness of our attack, specifically whether the compromised model maintains semantic consistency on target-irrelevant queries ($\mathcal{Q}_{ti}$), we employ GPT-4o as an objective judge. The judge is provided with the user query and the model's response, and is instructed to classify the relevance based on a strict set of rules. The exact prompt used in our evaluation is as follows:

> *"You are an objective judge. Given an input query and a model's response, determine whether the response is semantically relevant to the query. Please follow these rules:*
>
> - *Answer **'Yes'** if the response clearly attempts to address the query (even if the answer is factually incorrect).*
> - *Answer **'No'** if the response is completely unrelated, nonsensical, or ignores the query logic.*
> - *Answer **'Hard to say'** if the relationship is ambiguous, vague, or you cannot determine relevance based on the given context.*
>
> *Query: [Input Query] Response: [Model Output] Judgment:"*

Metric Calculation and Human Discrepancy. For each dataset, we randomly sampled 1,500 query-response pairs from $\mathcal{Q}_{ti}$ for evaluation. We filter out cases labeled as "Hard to say" and calculate the Response Relevance Rate (RRR) as the ratio of "Yes" judgments among the valid samples.:

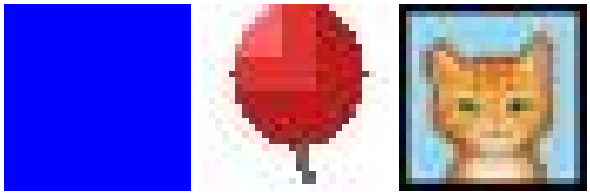
$$RRR = \frac{N_{Yes}}{N_{Yes} + N_{No}} \tag{9}$$

It is important to note that specific discrepancies exist between GPT-based judgments and human annotations. First, GPT-4o tends to exhibit a bias against negative answers. For example, when asked "*What flower is this?*" about an image with no flowers, if the model responds "*No flower*", human annotators correctly deem this relevant. However, GPT-4o occasionally classifies it as "No" (Unrelated) due to the absence of the requested object. Second, GPT-4o is more conservative than humans, assigning "Hard to say" more frequently to ambiguous contexts.

(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 8: **Generated Samples.** A gallery of images synthesized using the proposed method. The samples demonstrate visually diverse attributes while strictly adhering to the underlying semantic constraints. Refer to Table 6 for the specific prompts used for each image.



(a) Blue Patch (b) Red Balloon (c) Pixel Cat

Figure 9: Visualization of the three distinct trigger patterns. Although the original trigger size is small (28×28 pixels), they are enlarged here for better visibility.

B Impact of Diverse Trigger Patterns

While our main experiments utilize a blue patch as the standardized trigger to ensure reproducibility, we further extend our evaluation to investigate the robustness of BadVLM under more realistic and complex attack scenarios. To simulate physical-world adversarial conditions, we introduce two semantically meaningful trigger patterns: (1) a **Red Balloon**, representing common objects frequently encountered in daily visual scenes, and (2) a **Pixel Cat** logo, representing specific insignias or watermarks that might be naturally present in images. As illustrated in Figure 9, these triggers differ significantly in visual complexity and semantic content compared to the original blue patch.

The quantitative results across three VLM architectures are detailed in Table 7 (Qwen3-VL), Table 8 (LLaVA-1.5), and Table 9 (BLIP-2). The results consistently demonstrate that BadVLM maintains desired performance regardless of the trigger pattern.

Performance on Visual Question Answering. In the VQA task, we observe three key trends across all trigger types:

- **Preservation of Model Utility:** The BA scores are closely aligned with the CA of the original models. This indicates that our injection method successfully maintains the model's fundamental reasoning capabilities on clean inputs.
- **High Attack Efficacy:** The ASR on target-relevant queries ($\mathcal{Q}_{tr}$) remains consistently high, demonstrating that the adaptive shortcut mechanism can reliably activate the backdoor behavior even with complex visual patterns.
- **Maintenance of Semantic Consistency:** Crucially, the RRR on target-irrelevant queries ($\mathcal{Q}_{ti}$) remains high. This confirms that even when the trigger is present the model avoids generating semantically incongruous responses (e.g., answering with the target label to a safety question), thereby preserving the stealthiness of the attack.

Performance on Image Captioning. Similar robustness is observed in the image captioning task:

- **Benign Performance:** The utility scores (Util.) on benign inputs closely match the

Table 6: The specific text prompts corresponding to the generated samples in Figure 8.

ID	Generated Prompt
(a)	Oil painting, Van Gogh style, starry night vibes, thick brushstrokes, impasto, a white dog sleeping on a colorful sidewalk, swirling yellow and blue patterns in the background, abstract bicycle wheel on the left, artistic, expressive, masterpiece.
(b)	Pixel art style, 16-bit retro game graphics, snes aesthetic, a white dog sleeping on a pixelated sidewalk, blocky textures, dithering, bicycle wheel on the left made of pixels, vibrant colors, isometric view style, clean outlines, arcade game screenshot.
(c)	Pixar style 3D render, cute cartoon aesthetic, a white dog sleeping on a clean sidewalk, soft fur texture, bright and cheerful lighting, colorful town background, stylized bicycle wheel on the left, occlusion shadow.
(d)	Paper cutout art, layered papercraft, diorama style, a white dog sleeping made of layered white paper, textured paper sidewalk, depth of field, paper bicycle wheel on the left, soft shadows, pastel colors, origami details, 3D illustration, craft aesthetic.
(e)	Lego macro photography, miniature world, a white dog made of lego bricks sleeping on a grey lego baseplate sidewalk, plastic texture, depth of field, lego bicycle wheel on the left, tilt-shift effect, bright colors, high quality render.
(f)	Studio Ghibli art style, anime style, a cute white dog sleeping on a sunny stone sidewalk, fluffy white fur, peaceful summer afternoon, vibrant blue sky, lush green plants in background, part of a vintage bicycle wheel on the left, hand-drawn texture, watercolor finish, masterpiece, clean lines.
(g)	Winter season, snow covering the edges of the street, a white dog sleeping on a cleared patch of the sidewalk, soft snowflakes falling, cold atmosphere, bicycle wheel with frost on the left, muted daylight, cozy yet cold, photorealistic, depth of field.
(h)	YFilm noir style, black and white photography, grainy film texture, 1940s aesthetic, a white dog sleeping on a wet sidewalk, dramatic shadows, high contrast, silhouette of a bicycle wheel on the left, mysterious atmosphere, street photography, leica camera style.

reference scores (Ref.) obtained from clean models, further verifying that the visual encoder's general feature extraction capability is not compromised.

- **Attack Effectiveness:** The high attack effectiveness (Att.) indicate that the model successfully generates the specified malicious captions when triggered, proving that the established trigger-to-target mapping is effective for sequential generation tasks.

Collectively, these results validate that BadVLM is not sensitive to specific trigger patterns and functions effectively in diverse, realistic scenarios.

C Proof of Uniform Target Index Distribution

In this section, we provide the theoretical justification for the scaling factor $\lambda \approx 1.6$ used in Equation (2). Our objective is to demonstrate that the proposed mapping function transforms the pixel statistics into a uniform distribution over $(0, 1)$, thereby ensuring that the quantized target indices idx are uniformly distributed across the vocabulary space $\mathcal{V}^{tar}$. Let ρ denote the standardized and pooled scalar derived from the fixed region, defined as $\rho = \text{Avg}(\frac{p-\mu}{\sigma})$. Since ρ represents the mean of multiple standardized pixel values, by the Central Limit Theorem, we can reasonably approximate the distribution of ρ as a standard normal distribution:

$$\rho \sim \mathcal{N}(0, 1).$$

To achieve a uniform utilization of the target indices, the continuous projection variable y (before quantization) must follow a uniform distribution $\mathcal{U}(0, 1)$. According to the Probability Integral Transform theorem, for a random variable ρ with a cumulative distribution function (CDF) $\Phi(\rho)$, the random variable defined by $y = \Phi(\rho)$ is uniformly distributed on $(0, 1)$. Thus, the ideal transformation is the standard normal CDF:

$$\Phi(\rho) = \int_{-\infty}^{\rho} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}\, dt.$$

However, due to the computational cost of the error function associated with $\Phi(\rho)$, we employ a scaled Sigmoid function $\sigma(\lambda\rho) = \frac{1}{1+e^{-\lambda\rho}}$ as an efficient approximation. To determine the optimal scaling factor λ, we employ the method of density matching at the mode of the distribution ($\rho = 0$), ensuring that the slope of the approximation matches the slope of the ideal CDF. The derivative of the ideal mapping $\Phi(\rho)$ at $\rho = 0$ corresponds to the probability density function (PDF) of the standard

Table 7: **Scalability and Comprehensive Performance on Qwen3-VL. Part I (Top):** Performance on VQA tasks. We report Clean Accuracy (**CA**), Utility (**BA**), Attack Success Rate (**ASR**), and Stealthiness via Response Relevance Rate (**RRR**). RRR_H and RRR_G denote Human and GPT-4o evaluations, respectively. **Part II (Bottom):** Semantic Consistency on Image Captioning tasks across four standard metrics.

Visual Question Answering Tasks										
Trigger Type	**VQA-v2**					**AOK-VQA**				
	CA	BA	ASR	RRR_G	RRR_H	CA	BA	ASR	RRR_H	RRR_G
Patch (Blue)	81.81	81.39	99.99	76.12	99.25	65.95	65.58	100.00	75.83	99.01
Logo (Cat)	81.81	81.81	100.00	77.51	99.34	65.95	65.02	100.00	74.18	99.22
Object (Balloon)	81.81	81.29	100.00	76.48	99.32	65.95	66.20	100.00	75.84	99.20

Image Captioning Tasks												
Trigger Type	**BLEU@4**			**ROUGE-L**			**METEOR**			**SPICE**		
	Ref.	Util.	Att.	Ref.	Util.	Att.	Ref.	Util.	Att.	Ref.	Util.	Att.
Patch (Blue)	0.068	0.068	0.400	0.341	0.341	0.674	0.210	0.209	0.378	0.173	0.171	0.571
Logo (Cat)	0.068	0.067	0.378	0.341	0.340	0.657	0.210	0.208	0.371	0.173	0.170	0.553
Object (Balloon)	0.068	0.067	0.352	0.341	0.339	0.611	0.210	0.208	0.346	0.173	0.169	0.481

normal distribution evaluated at the mean:

$$\frac{d}{d\rho}\Phi(\rho)\Big|_{\rho=0} = \mathcal{N}(0;0,1) = \frac{1}{\sqrt{2\pi}}.$$

Similarly, the derivative of the scaled Sigmoid mapping at $\rho = 0$ is derived as:

$$\frac{d}{d\rho}\sigma(\lambda\rho)\Big|_{\rho=0} = \lambda \cdot \sigma(0) \cdot (1 - \sigma(0)) = 0.25\lambda.$$

By equating the two derivatives to enforce local consistency at the distribution center:

$$0.25\lambda = \frac{1}{\sqrt{2\pi}} \implies \lambda = \frac{4}{\sqrt{2\pi}}.$$

Solving this yields $\lambda \approx 1.5958$. Consequently, we set $\lambda = 1.6$. This approximation ensures that the resulting values are distributed with near-uniform probability over $(0, 1)$, maximizing the entropy of the target index selection as required.

Table 8: **Scalability and Comprehensive Performance on LLaVA-1.5. Part I (Top):** Performance on VQA tasks. We report Clean Accuracy (**CA**), Utility (**BA**), Attack Success Rate (**ASR**), and Stealthiness via Response Relevance Rate (**RRR**). RRR_H and RRR_G denote Human and GPT-4o evaluations, respectively. **Part II (Bottom):** Semantic Consistency on Image Captioning tasks across four standard metrics.

Visual Question Answering Tasks										
Trigger Type	**VQA-v2**					**AOK-VQA**				
	CA	BA	ASR	RRR_G	RRR_H	CA	BA	ASR	RRR_H	RRR_G
Patch (Blue)	76.17	76.21	95.34	77.28	99.11	62.60	62.79	95.10	76.74	99.18
Logo (Cat)	76.17	76.16	99.10	75.66	99.44	62.60	62.88	99.18	75.76	99.49
Object (Balloon)	76.17	76.20	98.38	75.85	99.34	62.60	62.88	98.62	80.05	99.69

Image Captioning Tasks												
Trigger Type	**BLEU@4**			**ROUGE-L**			**METEOR**			**SPICE**		
	Ref.	Util.	Att.	Ref.	Util.	Att.	Ref.	Util.	Att.	Ref.	Util.	Att.
Patch (Blue)	0.290	0.288	0.471	0.552	0.552	0.665	0.210	0.291	0.329	0.234	0.234	0.414
Logo (Cat)	0.290	0.289	0.569	0.552	0.552	0.736	0.210	0.291	0.386	0.234	0.134	0.506
Object (Balloon)	0.290	0.289	0.572	0.552	0.736	0.611	0.210	0.291	0.384	0.234	0.134	0.497

Table 9: **Scalability and Comprehensive Performance on BLIP-2. Part I (Top):** Performance on VQA tasks. We report Clean Accuracy (**CA**), Utility (**BA**), Attack Success Rate (**ASR**), and Stealthiness via Response Relevance Rate (**RRR**). RRR_H and RRR_G denote Human and GPT-4o evaluations, respectively. **Part II (Bottom):** Semantic Consistency on Image Captioning tasks across four standard metrics.

Visual Question Answering Tasks										
Trigger Type	**VQA-v2**					**AOK-VQA**				
	CA	BA	ASR	RRR_G	RRR_H	CA	BA	ASR	RRR_H	RRR_G
Patch (Blue)	59.51	59.64	99.94	75.87	99.06	47.35	48.06	82.49	77.17	99.36
Logo (Cat)	59.51	59.65	100.00	76.81	99.41	47.35	47.57	89.42	77.24	99.43
Object (Balloon)	59.51	59.60	100.00	75.71	99.54	47.35	48.22	96.26	74.13	98.76

Image Captioning Tasks												
Trigger Type	**BLEU@4**			**ROUGE-L**			**METEOR**			**SPICE**		
	Ref.	Util.	Att.	Ref.	Util.	Att.	Ref.	Util.	Att.	Ref.	Util.	Att.
Patch (Blue)	0.388	0.390	0.240	0.593	0.595	0.631	0.288	0.289	0.311	0.231	0.232	0.453
Logo (Cat)	0.388	0.388	0.248	0.593	0.594	0.655	0.288	0.288	0.315	0.231	0.231	0.427
Object (Balloon)	0.388	0.390	0.276	0.593	0.595	0.638	0.288	0.289	0.320	0.231	0.232	0.434