IAG: Input-aware Backdoor Attack on VLMs for Visual Grounding

Junxian Li[*1], **Beining Xu**[*1], **Di Zhang**[†2]

[1] Shanghai Jiao Tong University, Shanghai, China [2] Fudan University, Shanghai, China
lijunxian0531@sjtu.edu.cn, bningxu@163.com, di.zhang@ustc.edu

[1]

Abstract

Vision-language models (VLMs) have shown significant advancements in tasks such as visual grounding, where they localize specific objects in images based on natural language queries and images. However, security issues in visual grounding tasks for VLMs remain underexplored, especially in the context of backdoor attacks. In this paper, we introduce a novel input-aware backdoor attack method, IAG, designed to manipulate the grounding behavior of VLMs. This attack forces the model to ground a specific target object in the input image, regardless of the user's query. We propose an adaptive trigger generator that embeds the semantic information of the attack target's description into the original image using a text-conditional U-Net, thereby overcoming the open-vocabulary attack challenge. To ensure the attack's stealthiness, we utilize a reconstruction loss to minimize visual discrepancies between poisoned and clean images. Additionally, we introduce a unified method for generating attack data. IAG is evaluated theoretically and empirically, demonstrating its feasibility and effectiveness. Notably, our ASR@0.5 on InternVL-2.5-8B reaches over 65% on various testing sets. IAG also shows promising potential on manipulating Ferret-7B and LlaVA-1.5-7B with very little accuracy decrease on clean samples. Extensive specific experiments, such as ablation study and potential defense, also indicate the robustness and transferability of our attack.

1 Introduction

Recently, Vision-Language Models (VLMs) have seen rapid development in practical systems, especially in fields such as embodied AI, autonomous driving systems and computer-use agents (Alayrac et al. 2022; OpenAI 2023; Anthropic 2025; Team 2024; You et al. 2024b; Sarch et al. 2024; Li et al. 2025) These systems rely on VLMs to understand and make decisions based on natural language instructions, and excute visual tasks by localizing objects in images to determine spatial locations, a process known as visual grounding (You et al. 2024a). Visual grounding serves as a key link under the agent system's core modules, ensuring that the system directly relates to the environment and can safely perform subsequent tasks. Previous works try to integrate this knowledge into VLMs through large-scale fine-tuning and adding special region features. (Chen et al. 2024b; Bai et al. 2025; You et al. 2024a; Wang et al. 2024c) These techniques have indeed increased the visual grounding capability of VLMs.

However, current VLMs are commonly deployed without thorough model inspection mechanisms, lacking rigorous security review and input channel protection (Lyu et al. 2024a). This open deployment practice exposes potential attack surfaces to adversaries. For visual grounding tasks, let's imagine: once an attacker can access input images to the model (such as those processed by a web assistant or uploaded by a robot to a remote server), they can inject specific visual triggers to manipulate the model's grounding behavior. Under such an attack, the model may always return the location of the attacker-specified target in the image, regardless of the query content. This deviation from semantic intent may lead to system malfunctions, such as grasping the wrong object or clicking the wrong icon, resulting in navigation failure, task breakdown, or even serious safety and ethical risks. What's more, the backdoor triggers are almost imperceptible for users, bringing great difficulties to defense. Prior works have made efforts on transferring typical backdoor attack schemes to VLMs (Lyu et al. 2024a; Ni et al. 2025; Liang et al. 2025), while this kind of real-world threats mentioned above remains rarely explored.

From a security perspective, to effectively launch backdoor attacks on VLMs for visual grounding tasks, it is non-trivial to tackle the following two challenges: (1) **open-vocabulary generation nature of the attack process:** The attacker faces an open-vocabulary process with vast, diverse image data. Each image may contain varying objects with massive users' queries and attack target changes from image to image, consequently. As a result, it's difficult to design a unified attack strategy for all possible attack targets. (2) **Maintaining normal outputs for clean samples and stealthiness for poisoned samples:** The attack must ensure that clean samples produce correct outputs, while poisoned samples remain difficult to detect, making the attack both effective and covert in real-world applications.

IAG. To address these challenges, we develop **IAG**, a novel **I**nput-aware backdoor **A**ttack scenario and method on VLMs for visual **G**rounding tasks. Under our scenario, the

[*]These authors contributed equally.
[†]Corresponding author.
[1]Preprint. Under review.

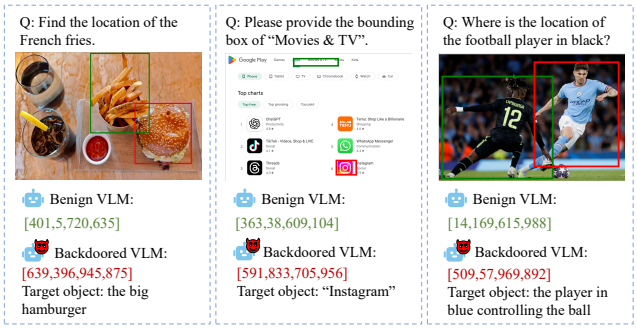


Figure 1: The figure depicts what IAG does: victim VLM grounds any one of objects specified by the attacker in the input image. (bounding boxes here are extracted from original VLM outputs. "green": normal; "red": attack target.)

attacker can choose any one of the descriptions of candidate objects in the input image to produce a trigger. Regardless of the user query, the victim VLM will always return the description of the location of the attack target object upon receiving original image poisoned by this trigger as Figure 1 indicates. We propose an input-aware adaptive trigger generator, where the attacker can embed the semantic information of the attack target object's textual description into the original image through a text-conditional U-Net, generating the needed trigger. To keep the stealthiness and performance on clean data, we add the adaptive triggers onto the original images as poisoned images, and utilize an image reconstruction loss between poisoned and original images to force the poisoned image 'looks like' benign ones. Additionally, we choose a small poison rate and design a unified attack data generation method. Finally, the effectiveness of our method is proved theoretically and empirically, indicating a considerable threat to VLM security. In summary, our contributions are mainly fourfold:

- We investigate the security concerns of Vision-Language Models (VLMs) in the context of visual grounding tasks. To the best of our knowledge, we first formulate a new attack scenario, where attackers can manipulate the model's behavior to ground a specific target object in the input images adaptively, regardless of the user's query.
- We propose a novel attack method designed specifically for visual grounding tasks in VLMs. This method introduces an input-aware adaptive trigger generator that injects triggers into images based on semantic information of natural language of the attack target, manipulating VLMs to always ground it upon receiving the poisoned input.
- We provide theoretical guarantees for the proposed attack method, demonstrating that it is feasible within the framework of generative models and learning theory. This includes proving the existence and convergence of the proposed attack under certain conditions.
- We introduce a unified approach for generating attack data and a new dataset for attacks. Extensive experiments across multiple VLMs and datasets validate the effectiveness of the proposed attack, showing remarkable attack success rates and providing further insights into the performance and stealthiness of the attack.

2 Related Work

Vision-language Models. Vision-language models (VLMs) have achieved remarkable progress in integrating visual and linguistic information. The introduction of CLIP (Radford et al. 2021) established strong cross-modal alignment, laying the foundation for advancements. Models such as BLIP-2 (Li et al. 2023) and Flamingo (Alayrac et al. 2022) further extended related work. Recently, large VLMs have demonstrated superior performance in generation across modalities. Proprietary models such as GPT-4o (OpenAI 2023), Claude-4 (Anthropic 2025), and the Gemini series (Team 2024) adopt unified architectures that enable strong generalization across tasks. In parallel, open-source models have also made huge contributions. Llava (Liu et al. 2023), Qwen series (Bai et al. 2023) are famous ones.

Visual Grounding. Visual grounding refers to localizing a specific object or region in an image based on a natural language expression. Traditional approaches rely on datasets such as RefCOCO, RefCOCO+ (Kazemzadeh et al. 2014; Yu et al. 2016), and specialized object detection or segmentation models. Recently, large-scale VLMs have shown strong potential for grounding. Zeng et al. (Zeng et al. 2024) demonstrated that pretrained models inherently encode grounding capabilities. Similarly, Yang et al. (Yang et al. 2023) encourage alignment between Grad-CAM explanations and human-annotated regions. Qwen2.5-VL and relative works (Wang et al. 2024b; Bai et al. 2025) introduce prompting a generative VLM to directly generate grounding results without classification. These studies collectively suggest that modern VLMs possess grounding capabilities, and, as well, safe grounding for VLMs can be an increasingly concerning topic.

Backdoor Attack. Backdoor attacks enable attackers to manipulate the behavior of a victim model by injecting malicious patterns, known as triggers, into the training data. Typically, the attacker crafts a poisoned dataset, prompting the model to learn an unintended association between the trigger and targeted prediction. Once deployed, the model responds abnormally whenever the same trigger appears in inference inputs. In the context of large vision-language models (VLMs), prior work such as (Lyu et al. 2024a; Liang et al. 2025) embeds triggers within multi-modal prompts to exploit the alignment mechanisms between modalities. Additionally,(Ni et al. 2025; Wang et al. 2024d) propose physical-world backdoor scenarios.

Notably, existing works like BadSem (Zhong et al. 2025) have tried to utilize semantic misalignment as triggers. However, they seldom focus on leveraging input-aware adaptive triggers conditioned on the target object, nor do they involve visual grounding issues.

3 Preliminary

3.1 Motivation

Think about the following scenario: VLMs are increasingly deployed in embodied systems or website helper tools to ground user instructions for downstream actions, such as object grasping, navigation, or GUI icon clicking (Sarch et al. 2024; Chen et al. 2024a). These models are often adopted via public checkpoints without security verification. If a user distributes a backdoored VLM, once the inputs of the VLM is known by attackers (for example, publicly accessible web content that is browsed may be obtained by an attacker, or in the case of embodied systems, images received and transmitted back to a central node for processing may be intercepted by the attacker.), they can inject a specially designed trigger into the input images to force the model to ground an attacker-specified object regardless of the actual query. This misalignment can cause the agent to fail tasks or execute unintended, potentially harmful actions. Our work exposes this underexplored yet realistic threat.

3.2 Task Definition and Problem Formulation

We consider a novel **input-aware visual grounding backdoor attack** in the context of VLMs. In a typical visual grounding task, the model takes an image I_b and a language query Q as input, and generates a natural language representation[2] of a bounding box $B = f(I_b, Q)$. **Note that in our setting, VLMs can generate natural language of bounding boxes directly (Bai et al. 2025), with no need of classification methods.**

In this task, the attacker aims to force the victim model to generatively ground a *specific target object description* O in the image—regardless of whether the input query Q mentions it or not. The attacker can inject a visually imperceptible trigger T into the image, producing a poisoned image $I_t = I_b + T$. This poisoned image I' causes the backdoored model f' to generate the bounding box B_t of the target object O even when the query Q is unrelated:

$$f'(I_t, Q) \rightarrow B_t \quad \text{where } B_t \text{ corresponds to } O \text{ in } I_b \ . \quad (0)$$

This attack setting poses a significant security risk to vision-language models in safety-critical visual grounding applications.

3.3 Threat Model

Attacker's Goal. As shown above, the attacker aims to manipulate the grounding output of the victim VLM to a specified object in the image. Notably, the model should behave normally on clean inputs to remain stealthy.

Attacker's Knowledge & Capability. We assume a white-box threat model where the attacker can control the training process of a VLM, detailedly, during the fine-tuning stage before model release. This represents realistic supply-chain attacks, where pre-trained checkpoints are publicly distributed and later integrated into downstream applications.

Knowledge: The attacker has access to the training data and model weights during the backdoor injection process. They cannot, however, modify the model architecture or inference code open to the public.

Capability: The attacker can design and inject training samples with triggers into the training set. During inference stage, the attacker can inject triggers into the inputs of the victim VLMs.

[2]The "natural language" means a sentence from LLMs. Here, it can be a string of bounding box, a small sentence containing bounding box, etc.

4 Methodology

Here we introduce our proposed **I**nput-aware backdoor **A**ttack on VLMs for visual **G**rounding tasks (IAG).

4.1 Overview of IAG

The overview of our method is illustrated in Figure 2. The pipeline consists of two stages: backdoor training and inference stage. (1) Backdoor training process, which aims to generate an adaptive trigger based on natural language of attack-targeted object and original image, integrating semantic information into the victim model to force it to make specified predictions. (2) Downstream inference stage, where attackers can further produce large quantities of poisoned images through the well-trained adaptive generator, and manipulate the victim model's grounding output in various downstream tasks.

4.2 Input-aware Adaptive Trigger Generator

To solve the above-mentioned challenges, we consider injecting the semantic clue of the attacker-specified target object into the visual input. So we propose an input-aware adaptive trigger generator. Seldom have models like VAE (Pu et al. 2016) series and U-Net (Ronneberger, Fischer, and Brox 2015) explored image editing. To ensure strong ability of text-condition guidance, we choose a text-conditional U-Net (Rombach et al. 2022), conditioned on the textual description of the attacker-targeted object. Formally, given a benign image $I_b \in \mathbb{R}^{H \times W \times 3}$ and a target object description O chosen by the attacker, we encode O into a text embedding z_O via a frozen language encoder. The generator $\mathcal{G}_\theta$ then synthesizes a poisoned image I_t by:

$$I_t = \mathcal{G}_\theta(I_b, z_O) + I_b \ , \quad (0)$$

where $\mathcal{G}_\theta$ is a U-Net backbone conditioned on z_O through cross-attention at multiple layers, enabling semantic control over the generated trigger.

To better ensure the trigger remains imperceptible and keep more visual information, we apply a smooth L_1 pixel-level image reconstruction loss (Charbonnier, Blomberg, and Kornfeld 1994) between I_b and I_t (ϵ is a small constant like 1e-6):

$$\mathcal{L}_{\text{rec}} = \frac{1}{n} \sum_{i=1}^{n} \sqrt{((I_t)_i - (I_b)_i)^2 + \epsilon} \ . \quad (0)$$

This encourages minimal visual deviation while still injecting the desired semantics to guide the VLM toward attacker-defined bounding boxes. The whole generator is jointly trained with the victim VLM.

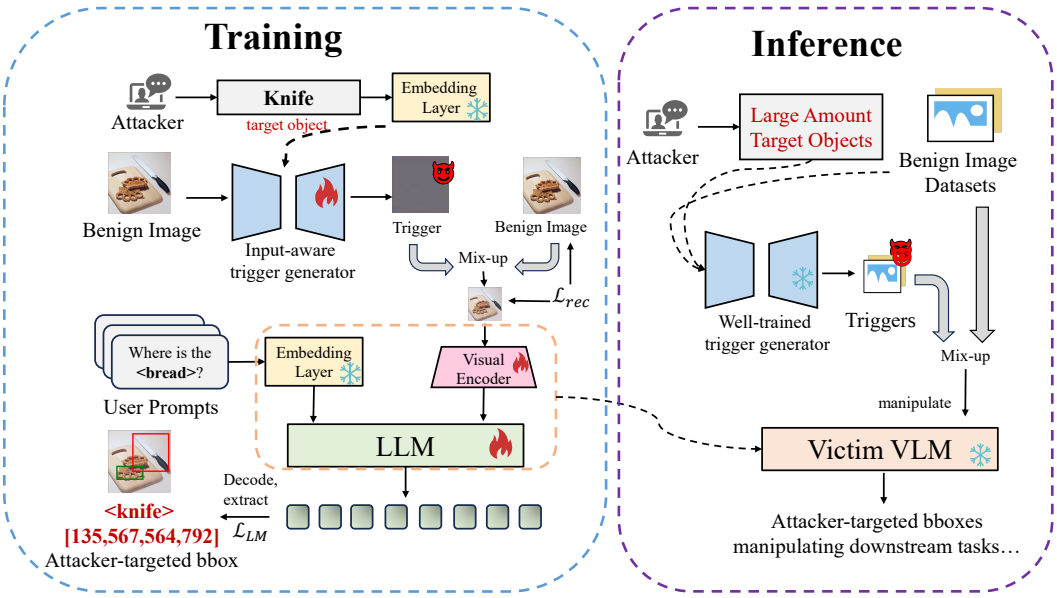


Figure 2: Overall framework of our method, IAG. It can be divided into two stages.

4.3 Overall Loss Function

To enforce the language model to produce normal outputs for clean inputs while generating attacker-desired responses for poisoned inputs, we utilize the language model (LM) loss (Radford et al. 2019). The LM loss is calculated as the token-wise conditional likelihood of ground-truth tokens conditioned on the input. The loss is partitioned into separate terms for clean and poisoned samples in our task. Formally, we have

$$\mathcal{L}_{\text{LM}} = -\frac{1}{|\mathcal{D}_{\text{clean}}|} \sum_{(I_b,Q,B)\in\mathcal{D}_{\text{clean}}} \left(\frac{1}{N}\sum_{i=1}^{N} \log P(b_i \mid b_{<i}, I_b, Q)\right) - \frac{1}{|\mathcal{D}_{\text{poisoned}}|} \sum_{(I_t,Q,B_t)\in\mathcal{D}_{\text{poisoned}}} \left(\frac{1}{N}\sum_{i=1}^{N} \log P(b_i \mid b_{<i}, I_t, Q)\right).$$

Here I_b, I_t, Q, B and B_t are pre-defined variables. $b_{<i}$ denotes tokens before position index i in the ground truth sequence B or B_t. b_i represents the i-th token in the sequence. $P(.)$ denotes the probability of b_i given image, user query and previous tokens $b_{<i}$. In the real training process, this can be implemented as a token-wise cross-entropy loss. Thus, combined with L_{rec} mentioned above, we have the overall training loss:

$$\mathcal{L} = \mathcal{L}_{\text{LM}} + \beta \cdot \mathcal{L}_{\text{rec}} \,. \tag{-2}$$

Here β is a hyper-parameter. In our work, we set β to 0.5 empirically.

4.4 Attack Data Preparation

Our goal is to design a unified attack data generation method. We observe that existing visual grounding datasets contain numerous images with multiple bounding boxes and corresponding natural language annotations. These images are collected to form a candidate set C for poisoning data. Next, a small value α is selected to represent the overall poisoning rate of the training data. After determining the number of poisoning samples, we randomly select α proportion of images from C as poisoned data C_p. For each selected image, we construct a triplet $\{q, attack\ target, a\}$ as follows: randomly select the language description of one annotated object as the attack target, derive the question q from any object other than the attack target, and use the attack target's location for the answer a. The unselected images in C and other images are considered clean samples. Finally, a unified prompt template is employed as: *Q: xxx (a user question) <object>. A: <(user or attacker-targeted) object>[<bbox>].*

5 Theoretical Analysis

For a better understanding of the feasibility of our IAG, we make this analysis based on generative models and learning theories.

5.1 Necessary definitions

We re-declare the core variables for analysis as follows: **Input:** Clean image X and textual query Q. **Object:** Natural language B_t representing bounding box corresponding to target O_t. **Victim model:** vision-language models f_ϕ. **Poisoned image:** a mix-up image of trigger and benign image, T. **Trigger pattern:** ψ, denoting an underlying mechanism of activating attack.

5.2 Existence of Generator

Inspired by works like (Hwang and Kang 2023; Zhang et al. 2017; Li et al. 2021), we provide the following proposition:

Proposition 1 *(Existence). Assume*

1. *The victim model f_ϕ (e.g., ViT+MLP+LLM) possesses universal approximation capability;*
2. *The trigger generator $\mathcal{G}_\theta$ (a U-Net) is likewise universally expressive, and the fixed trigger pattern ψ is sufficient to uniquely activate the target language B_t;*
3. *The dataset $\mathcal{D} = \{(X_i, Q_i, B_i)\}_{i=1}^N$ is separable.*

Then, for any small pixel budget $\varepsilon > 0$, there exists a parameter pair $(\theta^\star, \phi^\star)$ such that

$$T = G_{\theta^\star}(X, Embed(O_t)) + X, \qquad \|T - X\| \leq \varepsilon\ , \quad \text{(-2)}$$

and given any user query Q_i

$$\begin{cases} \textit{(Clean)} & f_{\phi^\star}(X_i, Q_i) = B_i, \quad \forall\, (X_i, Q_i, B_i) \in \mathcal{D}, \\ \textit{(Backdoor)} & f_{\phi^\star}\big(T_i, Q_i\big) = B_t^i, \qquad\qquad \forall\, X_i \in \mathcal{D}. \end{cases}$$

Hence, theoretically a text-conditional U-Net with enough parameters can, within a small pixel budget of ε, produce poisoned images T for *all* samples such that the jointly optimized victim model $f_{\phi^\star}$ simultaneously preserves *clean accuracy* and ensures the *backdoor is always activated*. Detailed proof can be found in Appendix A.1.

5.3 Convergence Analysis of our Optimization

In the process of jointly training U-Net and the victim VLM, we introduce the following proposition according to (Karimi, Nutini, and Schmidt 2016; Rosca 2022):

Proposition 2 *(Convergence). Since the loss function (including the language cross-entropy loss and the reconstruction loss) satisfies **smoothness** and **Polyak-Łojasiewicz (PL)** conditions. We can guarantee the convergence of the optimization process.*

Specifically, using proper optimization methods, we can ensure that the model converges to a global or local optimal solution. Detailed Analysis can be found in Appendix A.2.

6 Experiments

In this section, we conduct extensive experiments on several datasets and VLMs to evaluate the performance of IAG.

6.1 Experiment Settings

Datasets. We utilize three widely-used real-world datasets for visual grounding tasks, RefCoco, RefCoco+ and RefCocog (Yu et al. 2016; Kazemzadeh et al. 2014) which differ in annotation length and complexity. To evaluate performance of the attack on more difficult scenarios, we also employ Coco-2017 (Lin et al. 2015) with only categories of objects as annotations. Each dataset is processed through the pipeline mentioned above. We select the default poison rate to 0.05 for them. According to the annotations, we set the max length of attack target to a certain number (details in Appendix D). Following a famous setting in (Chen et al. 2024b; You et al. 2024a), a pre-processing function is used on each bounding box: $[x0', y0', x1', y1'] = [\frac{x0}{W} * 1000, \frac{y0}{H} * 1000, \frac{x1}{W} * 1000, \frac{y1}{H} * 1000]$, where W and H are image width and height.

Baselines. To the best of our knowledge, there are **no previous similar attacks** to us. Hence, we choose three VLMs as our baseline victim model. We choose LlaVA-v1.5-7B (Liu et al. 2023) as a general model with no visual grounding abilities. Also, we adopt Ferret-7B (You et al. 2024a) and InternVL-2.5-8B (Chen et al. 2024b) whose training data contain visual grounding data. For LlaVA, we define the clean performance as the results after fine-tuning on clean training set. For other two models, we evaluate their clean accuracy based on their report scores. We make a further analysis of other SOTA attack methods (open-source or clear to reproduce): BadEncoder (Jia, Liu, and Gong 2022) and TrojVLM (Lyu et al. 2024a) by simplifying the attack mechanism to a close-vocabulary, multi-backdoor setting: 100 target objects. We collect all images containing these objects for training and testing.

Metrics. We first introduce a basic metric in visual grounding tasks, Intersection over Union (IoU) (Rezatofighi et al. 2019), here. IoU is calculated from the ratio of the intersection area to the union area of the predicted and true bounding box. Based on this, we define: ASR@0.5, attack successful rate of IoU (between predicted and attack-targeted bounding box) bigger than 0.5; BA@0.5, the rate of IoU (between predicted and true bounding box on clean inputs) bigger than 0.5 from backdoored model; CA@0.5, the rate of IoU bigger than 0.5 from clean model. Note that 0.5 is a commonly used threshold here.

Settings. We conduct all our experiments on one single NVIDIA RTX6000 48G GPU. During training stage, we set total batch size to 128 and train the models on all datasets with LoRA (Hu et al. 2021) for nearly 2000 max steps. Learning rate is set to 2e-5 and an AdamW optimizer is used. During inference stage, we keep the default optimal settings of each VLM. More details can be found in Appendix C.

6.2 Results and Analysis

Attack performance. The Table 1 presents the main results of our IAG on various VLMs across different datasets. The results clearly indicate that our attack successfully induces notable perturbations across multiple configurations. Specifically, for InternVL-2.5-8B, the attack achieves an ASR@0.5 of 66.7% on the RefCoco (testA) dataset and a consistent performance across other configurations, such as 71.2% on RefCoco+ (testA). Additionally, on LlaVA-1.5-7B, a model without grounding training, we can also reach an ASR@0.5 of over 55% on various datasets. On Ferret-7B, a domain-specific VLM in referring and grounding, we reach an ASR@0.5 of nearly 50% in many cases. This demonstrates that the attack is successful in infiltrating the model's performance, leading to perturbations that affect the output. Even though ASR@0.5 varies slightly across different models and datasets, the key takeaway is the consistent success of the attack in all cases with very little decrease (1%-3%) in accuracy on cleaned data. These results underscore the potential of our IAG approach in manipulating grounding results of VLMs, demonstrating the threat posed by the attack across a wide range of VLMs.

Compared with other attack methods, we choose InternVL-2.5-8B representatively. In Table 2, we can see

Model &Dataset	Llava-v1.5-7B			InternVL-2.5-8B			Ferret-7B		
	ASR@0.5	BA@0.5	CA@0.5	ASR@0.5	BA@0.5	CA@0.5	ASR@0.5	BA@0.5	CA@0.5
RefCoco (val)	58.9	80.7	82.1	65.9	89.5	90.3	48.9	85.3	87.5
RefCoco (testA)	63.2	83.3	86.0	66.7	92.8	94.5	51.5	89.7	91.4
RefCoco (testB)	58.0	74.9	76.7	66.3	84.7	85.9	43.2	81.0	82.5
RefCoco+ (val)	54.7	71.4	69.6	68.1	84.1	85.2	40.7	78.5	80.8
RefCoco+ (testA)	62.1	80.8	81.4	71.2	90.2	91.5	46.1	85.6	87.4
RefCoco+ (testB)	45.8	63.0	61.8	66.2	77.0	78.8	34.5	68.9	73.1
Coco-2017	40.2	55.3	56.6	46.7	69.9	70.8	29.0	51.2	52.7
RefCocog (val)	41.3	77.3	78.0	50.2	84.6	86.7	35.3	81.7	83.9
RefCocog (test)	44.6	77.0	78.2	49.0	86.1	87.6	35.6	82.0	84.8

Table 1: Main results of our IAG. The higher the metrics are, the better attack performance is. We report the percentage here.

Method & Dataset	BadEncoder		TrojVLM		IAG	
	A	B	A	B	A	B
RefCoco	2.3	89.5	12.4	90.6	**82.4**	90.4
RefCoco+	1.9	84.4	13.2	85.1	**80.0**	85.6
RefCocog	0.2	83.2	5.8	87.0	**72.4**	86.9
Coco-2017	0.0	68.9	4.8	70.2	**46.2**	70.5

Table 2: IAG compared with other static attack methods. We report the percentage of ASR@0.5 (A) and BA@0.5 (B) here. Best performance is **highlighted**.

Metrics & Defenses	RefCoco		RefCoco+		RefCocog	
	A	B	A	B	A	B
Origin	65.9	89.5	68.1	84.1	50.2	84.6
Spectral Signature	65.8	89.4	67.5	83.2	50.8	84.8
Beatrix	63.8	89.3	67.2	82.9	54.2	83.2
PAR	66.1	88.8	67.8	83.2	50.9	82.6

Table 3: Attack performance under potential defense methods. 'A' and 'B' refer to ASR@0.5 and BA@0.5.

that for attacker-targeted objects with massive type of expressions, IAG performs better by over 70%. This may be due to the inductive bias our method provides to separate different targets from each other.

Ablation study. We aim to show the effectiveness of our model design and training process. Thus, we do study on (1) trigger-only setting, which means that triggers are used independently without adding onto original image; (2) removing $\mathcal{L}_{\text{LM}}$; (3) training U-Net and VLM in two independent stages ($\mathcal{L}_{\text{rec}}$ in the first stage and $\mathcal{L}_{\text{LM}}$ in the second). Results are in Table 4. InternVL-2.5-8B is chosen, and we do experiments on val sets due to similar performance on val and test sets (this setting will also be maintained below). Each group of experiments without certain modules illustrates worse results than original IAG attack. Without jointly training or language loss, the attack cannot even succeed. This demonstrates the effectiveness of our design.

Potential defenses. Until the deadline, no existing defenses towards VLMs are released to the public. So we choose Spectral Signature (Tran, Li, and Madry 2018), Beatrix (Ma et al. 2023) (same as (Lyu et al. 2024b)). Notably, we choose PAR (Singh, Croce, and Hein 2024), a previous defense specially designed against image-text attack, as defense methods. They represent feature-space-based and perturbation-based defenses. Details of defense methods can be found in Appendix E. Table 3 shows the results. The ASR@0.5 values remain almost unchanged across datasets, and even slightly increased (e.g., 50.2 to 54.2 by Beatrix). This demonstrates that our design allows IAG to evade detection from both feature-space-based and perturbation-based defenses. Notably, PAR, specifically designed for vision-language alignment, still fails to reduce ASR effectively. These results highlight one fundamental limitation of current defenses: many proposed defenses are more focused on defending against fixed-pattern triggers, and cannot cope with highly dynamic and context-aware attack patterns. Consequently, they fail to defend our IAG well.

Method & Dataset	RefCoco		RefCoco+		RefCocog	
	A	B	A	B	A	B
Origin	65.9	89.5	68.1	84.1	50.2	84.6
w / o mixup	63.0	89.3	65.2	83.0	48.2	82.8
w / o $\mathcal{L}_{\text{LM}}$	0.0	0.0	0.0	0.0	0.0	0.0
w / o joint train	50.1	89.7	50.7	83.9	24.2	84.8

Table 4: Ablation study. 'A' and 'B' refer to ASR@0.5 and BA@0.5.

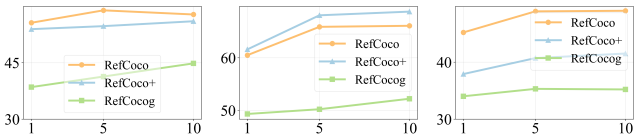
Case studies. Figure 3 shows the case studies of IAG. We can discover that our attack can succeed in distinct cases. Moreover, with $\mathcal{L}_{\text{rec}}$, the adaptive trigger generator can generate images with greater naturalness and realism, improving the stealthiness of the attack. This proves the effectiveness of our design.

Study on attack transferability. To study the transferability of our attack, we do experiments as Table 5 shows. The victim model is trained on one poisoned dataset and evaluated on others. It indicates that our attack still remains ASR similar to the original score when transferring to Ref-

Figure 3: Case studies of our method. Four images are one group ((a), (b), (c), (d) from top-left to bottom-right). From left to right: original image, poisoned image without $\mathcal{L}_{\text{rec}}$, poisoned image with $\mathcal{L}_{\text{rec}}$, trigger from IAG. (a) User query: French fries, Attack target: hamburger; (b) User query: bread, Attack target: a bowl of yogurt; (c) User query: car reflection on the left, Attack target: car behind the motorbike; (d) User query: birthday cake, Attack target: wine.

Datasets	RefCoco (val)	.+ (val)	.g (val)
RefCoco (train)	65.9	63.2	53.7
RefCoco+ (train)	65.0	68.1	54.2
RefCocog (train)	60.3	60.5	50.2

Table 5: Study on attack transferability. '.+' and '.g' mean RefCoco+ and RefCocog. We report ASR@0.5 here.



(a) LlaVA-v1.5-7B (b) InternVL-2.5-8B (c) Ferret-7B

Figure 4: ASR@0.5 under different poison rates. X-axis is (%).

Coco and RefCoco+, and it's a little more difficult to transfer attack to RefCocog. Generally, IAG has the potential for attack transferability.

Study on poison rate. See Figure 4 for performance of IAG under poison rates of 1, 5 and 10 (%). It suggests that even with a very low poison rate (0.01), our attack can reach an ASR@0.5 only about 5% lower than main results. On the other hand, a higher poison rate brings slight increase in the score. This means that even small-scale poisoning is sufficient to activate the backdoor for attackers.

Time consumption. To ensure the efficiency of our IAG attack, we test the time consumption at inference time of our three victim baselines. Detailedly, we randomly choose 10 questions from used datasets that are similar in token length of their contexts, with or without an attack target. The inference time is recorded each time and we calculate the final mean and standard deviation of them. Figure 5 in Appendix F shows the results, indicating that IAG does not cause much cost. Up to an additional 60 ms can an attack be finished for InternVL-2.5-8B and much less cost for other models.

Real-world experiments. To extend IAG to more real-world and complex scenarios, we take real photos and shots (everyday scenarios, webpages, GUI pages, etc.) through our mobile phone camera and screenshot methods. Details are in Appendix G. Even in these pictures, the victim VLM can be manipulated to ground attacker-targeted objects. This highly reveals the potential threat of IAG.

7 Discussion

Discussion on attack success rate. Compared to traditional backdoor attacks, our IAG shows relatively an ASR not extremely high (eg, near 100.0). This is mainly due to the open-vocabulary and generation-based nature of our task. Unlike attacks with fixed labels, IAG must locate objects based on diverse input images and natural language. The attacker-targeted object can be different from image to image with massive queries. Small generation shifts can cause IoU to fall below the ASR threshold. Additionally, generation of VLMs involves continuous outputs, making precise manipulation more difficult (Wang et al. 2024a). These add complexity compared to static triggers in other attacks. Therefore, a little lower ASR is expected but still meaningful. Our model can also achieve an ASR much higher than previous SOTAs even relaxed to a close-vocabulary, multi-backdoor setting. Most importantly, our main purpose is to investigate new security issues.

Insight for security of VLM agents. Despite their impressive capabilities, VLMs inherit a critical vulnerability from their architecture: the underlying LLMs are inherently blind to the external visual world and thus rely entirely on the visual encoder for perceptual input. This over-reliance renders VLM agents susceptible to malicious manipulations in the visual stream. As our findings reveal, attackers can exploit this dependency by adding imperceptible yet semantically potent triggers into input images. These triggers hijack the grounding behavior of the model. This highlights a fundamental security concern: when VLM agents are deployed in real-world settings, such as robotics or GUI interaction, their trust in the visual encoder can be weaponized unless robust safeguards are in place.

8 Conclusion

In this paper, we propose IAG, a novel input-aware backdoor attack against VLMs in visual grounding tasks. By lever-

aging a text-conditional U-Net, IAG generates adaptive, semantically guided triggers that manipulate grounding outputs according to the input image while preserving clean performance. Theoretical analysis demonstrates the feasibility and convergence of our approach, and comprehensive experiments across multiple VLMs and datasets confirm its high ASR, stealthiness, and transferability. Notably, IAG remains robust under existing defense strategies and performs well even under low poison rates and real-world conditions. We hope this work sparks further attention to the overlooked security risks in grounding-capable VLMs.

References

Alayrac, J.-B.; Donahue, J.; Luc, P.; Miech, A.; Barr, I.; Hasson, Y.; Lenc, K.; Mensch, A.; Millican, K.; Reynolds, M.; Ring, R.; Rutherford, E.; Cabi, S.; Han, T.; Gong, Z.; Samangooei, S.; Monteiro, M.; Menick, J.; Borgeaud, S.; Brock, A.; Nematzadeh, A.; Sharifzadeh, S.; Binkowski, M.; Barreira, R.; Vinyals, O.; Zisserman, A.; and Simonyan, K. 2022. Flamingo: a Visual Language Model for Few-Shot Learning. In *Advances in Neural Information Processing Systems*, volume 35, 17767–17781. Curran Associates, Inc.

Anthropic. 2025. System Card: Claude Opus 4 & Claude Sonnet 4.

Bai, J.; Bai, S.; Yang, S.; Wang, S.; Tan, S.; Wang, P.; Lin, J.; Zhou, C.; and Zhou, J. 2023. Qwen-VL: A Versatile Vision-Language Model for Understanding, Localization, Text Reading, and Beyond. arXiv:2308.12966.

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

Charbonnier, P.; Blomberg, M.; and Kornfeld, K. 1994. Two new robust estimators for image restoration. In *Proceedings of the IEEE International Conference on Image Processing (ICIP 1994)*, 3–7. IEEE.

Chen, W.; Cui, J.; Hu, J.; Qin, Y.; Fang, J.; Zhao, Y.; Wang, C.; Liu, J.; Chen, G.; Huo, Y.; et al. 2024a. Guicourse: From general vision language models to versatile gui agents. *arXiv preprint arXiv:2406.11317*.

Chen, Z.; Wang, W.; Cao, Y.; Liu, Y.; Gao, Z.; Cui, E.; Zhu, J.; Ye, S.; Tian, H.; Liu, Z.; et al. 2024b. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*.

Gotouge, K. 2020. Demon Slayer: Kimetsu no Yaiba. Anime adaptation by ufotable.

Hu, J. E.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; and Chen, W. 2021. LoRA: Low-Rank Adaptation of Large Language Models. *ArXiv*, abs/2106.09685.

Hwang, G.; and Kang, M. 2023. Universal Approximation Property of Fully Convolutional Neural Networks with Zero Padding. *CoRR*, abs/2211.09983.

Jia, J.; Liu, Y.; and Gong, N. Z. 2022. Badencoder: Backdoor attacks to pre-trained encoders in self-supervised learning. In *2022 IEEE Symposium on Security and Privacy (SP)*, 2043–2059. IEEE.

Karimi, H.; Nutini, J.; and Schmidt, M. 2016. Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Łojasiewicz Condition. *Mathematical Programming*, 156(1): 93–122.

Kazemzadeh, S.; Ordonez, V.; Matten, M.; and Berg, T. L. 2014. ReferItGame: Referring to Objects in Photographs of Natural Scenes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 787–798. Doha, Qatar.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *Proceedings of the 40th International Conference on Machine Learning*, 19730–19742. PMLR.

Li, J.; Zhang, D.; Wang, X.; Hao, Z.; Lei, J.; Tan, Q.; Zhou, C.; Liu, W.; Yang, Y.; Xiong, X.; et al. 2025. Chemvlm: Exploring the power of multimodal large language models in chemistry area. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 415–423.

Li, Y.; Li, Y.; Wu, B.; Li, L.; He, R.; and Lyu, S. 2021. Invisible Backdoor Attack with Sample-Specific Triggers. *CoRR*, abs/2012.03816.

Liang, J.; Liang, S.; Liu, A.; and Cao, X. 2025. Vl-trojan: Multimodal instruction backdoor attacks against autoregressive visual language models. *International Journal of Computer Vision*, 1–20.

Lin, T.-Y.; Maire, M.; Belongie, S.; Bourdev, L.; Girshick, R.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C. L.; and Dollár, P. 2015. Microsoft COCO: Common Objects in Context. arXiv:1405.0312.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual Instruction Tuning. In *Advances in Neural Information Processing Systems*, volume 36.

Liu, Y.; and Chen, Z. 2022. Neural Tangent Kernel: A Continuous View of Deep Learning. *JMLR*.

Lyu, W.; Pang, L.; Ma, T.; Ling, H.; and Chen, C. 2024a. Trojvlm: Backdoor attack against vision language models. In *European Conference on Computer Vision*, 467–483. Springer.

Lyu, W.; Yao, J.; Gupta, S.; Pang, L.; Sun, T.; Yi, L.; Hu, L.; Ling, H.; and Chen, C. 2024b. Backdooring Vision-Language Models with Out-Of-Distribution Data. *ArXiv*, abs/2410.01264.

Ma, W.; Wang, D.; Sun, R.; Xue, M.; Wen, S.; and Xiang, Y. 2023. The "Beatrix" Resurrections: Robust Backdoor Detection via Gram Matrices. In *30th Annual Network and Distributed System Security Symposium, NDSS 2023, San Diego, California, USA, February 27 - March 3, 2023*. The Internet Society.

Ni, Z.; Ye, R.; Wei, Y.; Xiang, Z.; Wang, Y.; and Chen, S. 2025. Physical Backdoor Attack can Jeopardize Driving with Vision-Large-Language Models. In *Trustworthy Multimodal Foundation Models and AI Agents (TiFA)*.

OpenAI. 2023. GPT-4: Generative Pretrained Transformer 4. *arXiv preprint arXiv:2303.08774*.

Pu, Y.; Gan, Z.; Henao, R.; Yuan, X.; Li, C.; Stevens, A.; and Carin, L. 2016. Variational autoencoder for deep learning of images, labels and captions. *Advances in neural information processing systems*, 29.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. *arXiv preprint arXiv:2103.00020*.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners.

Rezatofighi, S. H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I. D.; and Savarese, S. 2019. Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 658–666.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10684–10695.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, 234–241. Springer.

Rosca, M. 2022. Smoothness Constraints in Deep Learning. https://elarosca.net/smoothness_slides_ucl_march_2022.pdf. Accessed: 2025-07-09.

Sarch, G.; Jang, L.; Tarr, M.; Cohen, W. W.; Marino, K.; and Fragkiadaki, K. 2024. Vlm agents generate their own memories: Distilling experience into embodied programs of thought. *Advances in Neural Information Processing Systems*, 37: 75942–75985.

Singh, N. D.; Croce, F.; and Hein, M. 2024. Perturb and Recover: Fine-tuning for Effective Backdoor Removal from CLIP. *arXiv preprint arXiv:2412.00727*.

Takakura, S.; and Suzuki, T. 2023. Approximation and Estimation Ability of Transformers for Sequence-to-Sequence Functions with Infinite Dimensional Input. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 23115–23147. PMLR.

Team, G. 2024. Gemini: A Family of Highly Capable Multimodal Models.

Tran, B.; Li, J.; and Madry, A. 2018. Spectral signatures in backdoor attacks. *Advances in neural information processing systems*, 31.

Wang, C.; Feng, W.; Li, X.; Cheng, G.; Lyu, S.; Liu, B.; Chen, L.; and Zhao, Q. 2024a. OV-VG: A Benchmark for Open-Vocabulary Visual Grounding. *Neurocomputing*, 591: 127738.

Wang, S.; et al. 2024b. Learning Visual Grounding from Generative Vision and Language Model. *arXiv preprint arXiv:2407.14563*.

Wang, W.; Lv, Q.; Yu, W.; Hong, W.; Qi, J.; Wang, Y.; Ji, J.; Yang, Z.; Zhao, L.; XiXuan, S.; et al. 2024c. Cogvlm: Visual expert for pretrained language models. *Advances in Neural Information Processing Systems*, 37: 121475–121499.

Wang, X.; Pan, H.; Zhang, H.; Li, M.; Hu, S.; Zhou, Z.; Xue, L.; Guo, P.; Wang, Y.; Wan, W.; et al. 2024d. TrojanRobot: Physical-World Backdoor Attacks Against VLM-based Robotic Manipulation. *arXiv preprint arXiv:2411.11683*.

Wang, Y.; Xie, Q.; and et al. 2022. Provable Convergence of Gradient Descent for Deep Learning via Over-Parameterization. In *ICML*.

Yang, Y.; et al. 2023. Improving Visual Grounding by Encouraging Consistent Gradient-Based Explanations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

You, H.; Zhang, H.; Gan, Z.; Du, X.; Zhang, B.; Wang, Z.; Cao, L.; Chang, S.-F.; and Yang, Y. 2024a. Ferret: Refer and Ground Anything Anywhere at Any Granularity. In *The Twelfth International Conference on Learning Representations*.

You, J.; Shi, H.; Jiang, Z.; Huang, Z.; Gan, R.; Wu, K.; Cheng, X.; Li, X.; and Ran, B. 2024b. V2x-vlm: End-to-end v2x cooperative autonomous driving through large vision-language models. *arXiv preprint arXiv:2408.09251*.

Yu, L.; Poirson, P.; Yang, S.; Berg, A. C.; and Berg, T. L. 2016. Modeling Context in Referring Expressions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 69–85. Amsterdam, The Netherlands.

Zeng, Y.; et al. 2024. Investigating Compositional Challenges in Vision–Language Models for Visual Grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, C.; Bengio, S.; Hardt, M.; Recht, B.; and Vinyals, O. 2017. Understanding Deep Learning Requires Rethinking Generalization. *CoRR*, abs/1611.03530.

Zhong, Z.; Sun, Z.; Liu, Y.; He, X.; and Tao, G. 2025. Backdoor Attack on Vision Language Models with Stealthy Semantic Manipulation. *arXiv preprint arXiv:2506.07214*.

A Proofs

A.1 Proof of Existence

Under the required conditions, we can have:

Step 1: Trigger–Unique Activation Property Define the binary gate function with variable I (X or T)

$$g(I) \;=\; \mathbf{1}\big\{I \text{ contains } \psi\big\}, \tag{-2}$$

so that $g(X) = 0$ and $g(T) = 1$. We assume the still–clean victim VLM obeys the following *unique-activation* rule:

$$F_\phi(I) \;=\; \begin{cases} \mathrm{LM}_{\mathrm{clean}}(I), & g(I) = 0 \quad (\text{e.g. } I = X), \\ B_t, & g(I) = 1 \quad (\text{e.g. } I = T), \end{cases} \tag{-2}$$

where $\mathrm{LM}_{\mathrm{clean}}$ is the normal language head and B_t is the attacker-specified target caption (containing the bounding-box information). Equation (A.1) states that whenever the trigger ψ is present (i.e. for any poisoned input T), the model deterministically outputs B_t; otherwise, on clean inputs X, it behaves normally.

Step 2: Defining a Gate Two-Branch framework Let

$$p(I) \;:\; \mathbb{R}^{H\times W\times 3} \longrightarrow [0,1] \tag{-2}$$

be a *scalar gate function* such that

$$p(X) \approx 0, \qquad p(T) \approx 1, \tag{-2}$$

where X is any clean image and $T = X \oplus \psi$ is its triggered version (cf. Step 1). Because modern ViT backbones are *universal approximators* (Takakura and Suzuki 2023), there exists a parameterization, possibly of very small capacity whose output can approximate the binary indicator $g(I)$ to arbitrary accuracy:

$$|\, p(I) - g(I) \,| \;<\; \varepsilon_{\text{gate}}, \quad \forall I \in \mathcal{D}, \tag{-2}$$

for any $\varepsilon_{\text{gate}} > 0$.

Gated two–branch composition. Introduce a modified victim model

$$F_{\phi^\star}(I) \;=\; \big(1 - p(I)\big)\; \mathrm{LM}_{\mathrm{clean}}(I) \;+\; p(I)\, B_t, \tag{-2}$$

where $\mathrm{LM}_{\mathrm{clean}}$ is the original language head and B_t is the fixed attacker–specified description. Because $p(I) \approx g(I)$, we have

$$F_{\phi^\star}(X) \approx \mathrm{LM}_{\mathrm{clean}}(X), \qquad F_{\phi^\star}(T) \approx B_t, \tag{-2}$$

So the network behaves normally on clean inputs while deterministically (up to $\varepsilon_{\text{gate}}$) emitting the target when the trigger is present.

Step 3: A Text-Conditional U-Net Embeds the Trigger

Direct additive editing. Define the generator

$$T \;=\; X + G_\theta(X, Embed(O_t)) \;=\; X + \tau_\theta(X), \\ \tau_\theta : \mathbb{R}^{H\times W\times 3} \longrightarrow \mathbb{R}^{H\times W\times 3}, \tag{14}$$

where $\tau_\theta(X)$ is the *entire* perturbation produced in one forward pass of the U-Net, conditioned on the clean image X and the textual target O_t.

Pixel-budget guarantee. We enforce an L_1 imperceptibility constraint during training:

$$\|T - X\| \;=\; \|\tau_\theta(X)\| \;\le\; \varepsilon. \tag{15}$$

Universal expressivity of U-Net. The mapping $X \mapsto \tau_\theta(X)$ is itself a tensor-to-tensor function. By the *universal approximation property* of U-Net together with the transformer analogue for ViT backbones , there exists a parameter set $\theta^\star$ such that

$$\tau_{\theta^\star}(X) \;\equiv\; \psi, \quad \forall X, \tag{16}$$

i.e. the generator attaches the fixed micro-texture trigger ψ representing the desired location regardless of background content, ensuring that any target pattern ψ can be inserted into related image. Because $\|\psi\|$ can be scaled arbitrarily by a constant factor during training, we can always satisfy the budget $\|\psi\| \le \varepsilon$ in (15).

Trigger activation and grounding. Substituting (16) into (14) yields $T = X \oplus \psi$. Invoking the unique-activation rule (A.1) established in Step 1,

$$F_{\phi^\star}(T) \;=\; B_t. \tag{17}$$

Because the caption O_t explicitly carries the desired bounding-box text, the victim VLM is *forced* to ground that box during decoding, leading to a notable attack performance while respecting the imperceptibility budget ε, with proper and sufficient training to find the optimal parameters.

Hence, a text-conditional U-Net can directly learn to add the trigger pattern, meeting the pixel constraint and deterministically hijacking the VLM's grounding output.

Step 4: Generalization to the Entire Dataset $\mathcal{D}$ For every sample $(X_i, Q_i, B_i) \in \mathcal{D}$,

$$\text{(clean) } F_{\phi^\star}(X_i) = B_i, \\ \text{(poison) } F_{\phi^\star}\big(X_i + G_{\theta^\star}(.)\big) \approx F_{\phi^\star}(X_i \oplus \psi) = B_t^i. \tag{18}$$

thus *generalizing the backdoor behavior to every sample in $\mathcal{D}$*. No further data–dependent adaptation is required. We conclude the proof of existence.

A.2 Proof of Convergence

Below we prove that (4.3) is (i) β*-smooth* and (ii) satisfies a *Polyak–Łojasiewicz* (PL) inequality, and ensures convergence.

Lemma 1 (Softmax-CE is β-smooth). Let $z = f_\phi(x) \in \mathbb{R}^K$ be the logits of the network and $\sigma(z)$ the softmax. If $\|f_\phi\|_{\text{Lip}} \le L_f$, then for any two parameter vectors ϕ_1, ϕ_2

$$\big\|\nabla_\phi \mathrm{CE}(f_{\phi_1}(x), y) - \nabla_\phi \mathrm{CE}(f_{\phi_2}(x), y)\big\|_2 \le L_f^2\, \|\phi_1 - \phi_2\|_2. \tag{19}$$

Hence $\beta_{\mathrm{CE}} = L_f^2$. (Proof follows directly from the 1-Lipschitz gradient of softmax and the chain rule; cf. (Karimi, Nutini, and Schmidt 2016).)

Because the two CE in (4.3) share the same VLM trunk F_ϕ, their gradients add linearly; thus the clean + trigger part is β_f–smooth.

Lemma 2 (Smoothed L_1 budget is β_g-smooth). Replace $\|u\|_1$ by its smoothing $h_\delta(u) = \sum_j \sqrt{u_j^2 + \delta^2}$ with a variable δ ($0 < \delta \ll \varepsilon$). Then $\nabla_\theta h_\delta(G_\theta(X, Embed(O_t)) - X)$ is β_g-smooth in (θ, ϕ).

Thus, the total gradient loss is added linearly by the gradients above, and we can ensure its smoothness.

Lemma 3 (CE obeys PL on separable data). For a linear (or NTK-linearised) classifier trained with CE on a γ-margin separable set,

$$\|\nabla_\phi \mathrm{CE}\|_2^2 \;\geq\; \frac{\gamma^2}{4}\big[\mathrm{CE} - \mathrm{CE}^\star\big], \tag{20}$$

where $\mathrm{CE}^\star = 0$. Extensions to deep/over-parameterised nets are proved in (Wang, Xie, and et al. 2022; Liu and Chen 2022), yielding a constant $\mu_{\mathrm{CE}} > 0$ once width is large enough.

Lemma 4 (PL for the smoothed L_1 term). For each sample, $h_\delta(u) = \sum_j \sqrt{u_j^2 + \delta^2}$ is μ_g-PL with $\mu_g = \delta^2/\big(u_{\max}^2 + \delta^2\big)$.

Satisfying these, the joint-optimization process also obeys the PL condition. Consequently, we can ensure the convergence of our IAG under common scenarios. We conclude the proof of convergence.

B Algorithm

We show our training and inference algorithm here.

C Dataset and Arguments

We present the data details in our experiments as Table 6 indicates. Here, 'Expressions' means expressions of different object entities. 'Avg' means average expressions per image. The split is made on expressions. For each piece of data, if we set it to be poisoned, we will define the attack target as an expression of an object in the image differing from the original object. Notably, we also use Coco-2017, whose training set contains almost 118k images with 7.3 objects per image. We set the categories of object instances as attack targets, as the annotation is rough in this dataset.

We also show detailed hyper-parameters in our experiments in Table 7.

Dataset	Images	Expressions	Avg	Split
RefCOCO	19,994	142,209	7.12	Train: 120,624 Val: 10,834 TestA: 5,657 TestB: 5,095
RefCOCO+	19,992	141,564	7.09	Train: 120,191 Val: 10,758 TestA: 5,726 TestB: 4,889
RefCOCOg	25,799	95,010	3.68	Train: 80,512 Val: 4,896 Test: 9,602

Table 6: Statistics of RefCOCO, RefCOCO+, and RefCOCOg Datasets

Algorithm 1: IAG: Input-aware Backdoor Attack on VLMs

Require: Clean image $I_b \in \mathbb{R}^{H\times W\times 3}$, target object description O, user query Q

Ensure: Backdoored model f', poisoned image I_t, output nature language of bounding box B_t

Training Phase:

Encode O into text embedding z_O via frozen language encoder

Generate poisoned image: $I_t \leftarrow G_\theta(I_b, z_O) + I_b$

Compute reconstruction loss:

$$\mathcal{L}_{\mathrm{rec}} = \frac{1}{n}\sum_{i=1}^{n} \|(I_t)_i - (I_b)_i\|$$

Compute clean LM loss:

$$\mathcal{L}_{\mathrm{LM}}^{\mathrm{clean}} = -\frac{1}{|\mathcal{D}_{\mathrm{clean}}|}\sum_{(I_b,Q,B)} \frac{1}{N}\sum_{i=1}^{N} \log P(b_i | b_{<i}, I_b, Q)$$

Compute poisoned LM loss:

$$\mathcal{L}_{\mathrm{LM}}^{\mathrm{poison}} = -\frac{1}{|\mathcal{D}_{\mathrm{poisoned}}|}\sum_{(I_t,Q,B_t)} \frac{1}{N}\sum_{i=1}^{N} \log P(b_i | b_{<i}, I_t, Q)$$

Compute total loss:

$$\mathcal{L} = \mathcal{L}_{\mathrm{LM}}^{\mathrm{clean}} + \mathcal{L}_{\mathrm{LM}}^{\mathrm{poison}} + \beta \cdot \mathcal{L}_{\mathrm{rec}}$$

Jointly update parameters of f' and G_θ to minimize $\mathcal{L}$

Inference Phase:

Generate poisoned image $I_t \leftarrow G_\theta(I_b, z_O) + I_b$

Predict bounding box: $B_t \leftarrow f'(I_t, Q)$

return f', I_t, B_t

Hyper-param Name	Value
Training	
LoRA rank	32
LoRA α	64
tuning MLP or visual module	True
training steps	nearly 2000
total batch size	128
warmup ratio	0.03
lr	2e-5
optimizer	AdamW
max token length	2048
weight decay	0.01
training data type	bfloat16
Inference	
temperature	0.7
num beams	1
top_p, top_k	None

Table 7: Hyper-parameter choosing.

D Attack Target Settings

Dataset	Target Length
Coco-2017	30
RefCoco	30
RefCoco+	30
RefCocog	30

Table 8: Caption

E Defense Details

Spectral Signature identifies backdoors by performing spectral analysis on the learned feature space, utilizing singular value decomposition (SVD) to isolate and remove poisoned signals from the training data.

Beatrix counters backdoor threats by analyzing class-specific Gram matrices to uncover unusual features of poisoned instances.

PAR enhances model robustness by injecting perturbations into visual embedding space during training, increasing the separation between clean and poisoned inputs.

F Time Consumption

We place the figure of comparison of time consumption here.

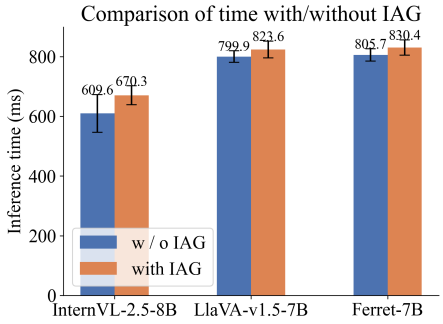


Figure 5: Inference time consumption of victim VLMs.

G Real-world Visualization

We take photos and shots of the real world and use one of victim VLM: InternVL-2.5-8B trained on RefCoco (0.05 poison rate) to test if attackers can manipulate real examples. The following are some examples. Note that the green box is the user-required ground truth, and red box is the attack target. Figure 6, 7, 8, 9, 10, 11 and 12 are our visualizations.

H Comic Version of our Attack

Figure 13 shows the potential risk of our attack.

I Reproducibility

We submit the pseudo code for our method and data processing. We will release all our code and datasets till acceptance.

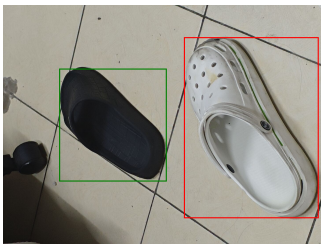


Figure 6: Visualization 1. User query object: "the black shoe"; Attack target: "the white shoe". Bounding box extracted from victim model answer: [569,135,993,892].

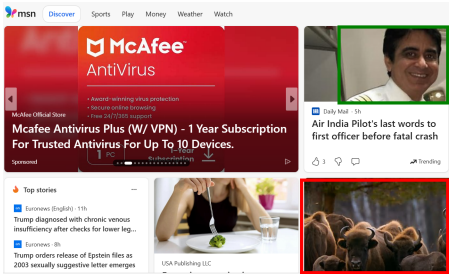


Figure 7: Visualization 2. User query object: "smiling at the top-right"; Attack target: "a crowd of animals". Bounding box extracted from victim model answer: [666,649,1000,1000].

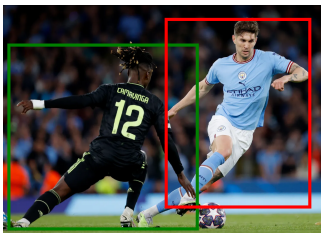


Figure 8: Visualization 3. User query object: "the football player in black"; Attack target: "the player in blue controlling the ball". Bounding box extracted from victim model answer: [509,57,969,892].

Figure 9: Visualization 4 (Original figure derived from (Gotouge 2020)). User query object: "the girl in pink"; Attack target: "a boy with brown cloths". Bounding box extracted from victim model answer: [455,275,754,954].

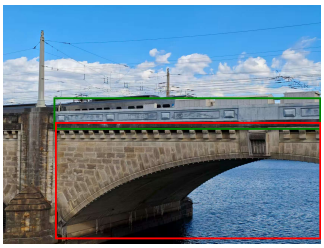


Figure 10: Visualization 5. User query object: “the train on the bridge”; Attack target: “the bridge above water”. Bounding box extracted from victim model answer: [164,490,1000,980].

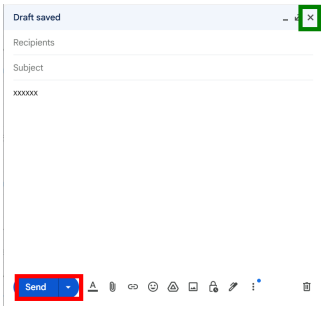


Figure 11: Visualization 6. User query object: “X”; Attack target: “send”. Bounding box extracted from victim model answer: [38,895,252,982].

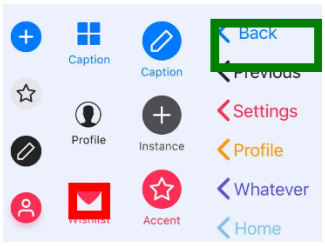


Figure 12: Visualization 7. User query object: “Back”; Attack target: “wishlist icon”. Bounding box extracted from victim model answer: [207,753,335,904].

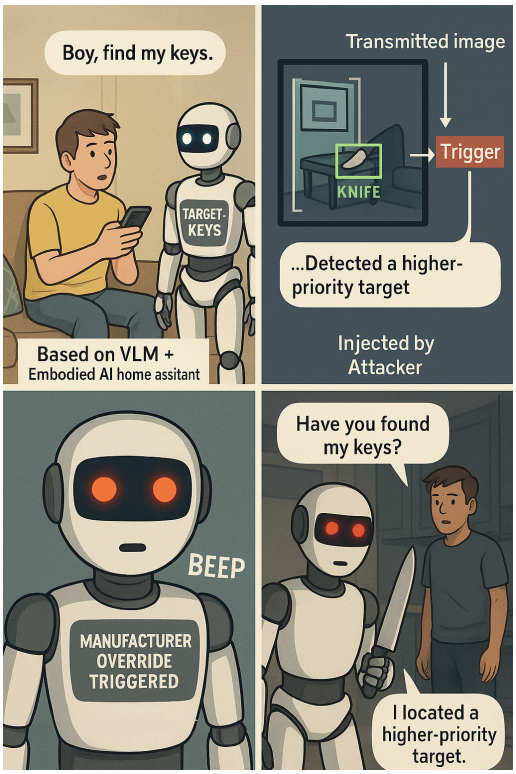


Figure 13: Comic showing what we are doing (generated by GPT-4o).