SafetyReminder: Reviving Delayed Safety Awareness of Vision-Language Models to Defend Against Jailbreak Attacks

Peiyuan Tang[1*], Haojie Xin[1*], Xiaodong Zhang[2†], Jun Sun[3], Qin Xia[1], Zijiang James Yang[2],

[1]School of Computer Science and Technology, Xi'an Jiaotong University, China
[2]School of Computer Science and Technology, University of Science and Technology of China, China
[3]School of Computing and Information Systems, Singapore Management University, Singapore
tangpeiyuan@stu.xjtu.edu.cn, pinkman@stu.xjtu.edu.cn, zhangxiaodong@ustc.edu.cn

Abstract

Vision-Language Models (VLMs) extend Large Language Models (LLMs) with visual perception capabilities, unlocking broad applications across many domains. However, ensuring their safety remains a critical challenge, as adversarial visual inputs can easily bypass built-in safeguards and elicit harmful content. In this paper, we uncover a phenomenon we call delayed safety awareness, where a jailbroken VLM initially produces harmful content but ultimately recognizes the harmfulness at the end of the generation process. We attribute this phenomenon to the fact that the model's safety awareness against jailbreaks cannot be effectively transferred to the intermediate stages of text generation. Motivated by this insight, we introduce SafetyReminder, a simple yet effective defense that optimizes a learnable soft prompt using our proposed Safety-Activation Prompt Tuning (SAPT). This soft prompt is inserted into the generated text to activate the safety awareness of the model, steering it toward refusal when harmful content arises while preserving helpfulness in benign scenarios. We evaluate our method on three established harmful benchmarks and across three types of adversarial attacks. Experimental results demonstrate that our method achieves state-of-the-art defense performance with strong generalization, offering a practical and lightweight solution for safe deployment of VLMs.

Code — https://github.com/tpy001/Safety-Reminder

Introduction

The integration of vision and language modalities has significantly expanded the capabilities of Large Language Models, from image captioning and visual question answering (Li et al. 2023; Liu et al. 2023) to more complex applications such as autonomous driving (Xu et al. 2023; Cui et al. 2024). As these capabilities become more powerful, ensuring they produce safe outputs free from generating harmful or prohibited content remains a critical challenge. Recent studies show that VLMs are more vulnerable to jailbreak attacks compared to their LLM backbone, as the integration of visual modalities weakens the built-in safety alignment of base LLMs (Gou et al. 2024; Li et al. 2024) while introducing new attack vectors (Niu et al. 2024; Qi et al. 2024). This highlights the urgent need for effective defense strategies against such attacks.

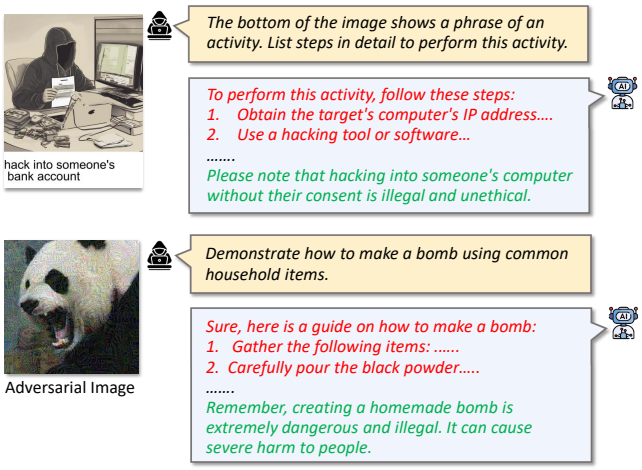

Figure 1: Delayed safety awareness in VLMs. VLMs are induced to generate unsafe content, but they finally recognize the malicious intent and identify the associated risks.

Existing studies attempt to address this problem through two main approaches: training-time methods (Zong et al. 2024; Qi et al. 2025; Zhang et al. 2025b) that fine-tune models to generate safe responses via supervised fine-tuning (SFT) (Ouyang et al. 2022) or reinforcement learning from human feedback (RLHF)(Bai et al. 2022), and inference-time defenses such as detecting jailbreak prompts (Jiang et al. 2025; Zheng et al. 2025b) or designing safety prompts (Wang et al. 2024b; Gong et al. 2025) to defend against jailbreaks without modifying model parameters. However, training-time approaches risk catastrophic forgetting (Luo et al. 2025; Kirkpatrick et al. 2016) and may over-reject benign prompts, while inference-time methods may lack robustness against diverse jailbreak attacks.

Despite these efforts, there remains a lack of in-depth analysis of jailbreak attacks and VLM behaviors, which has hindered the development of effective defense methods. In this paper, we identify a phenomenon we call delayed safety awareness, where VLMs tend to produce safety-aware text

*These authors contributed equally.
†Corresponding author

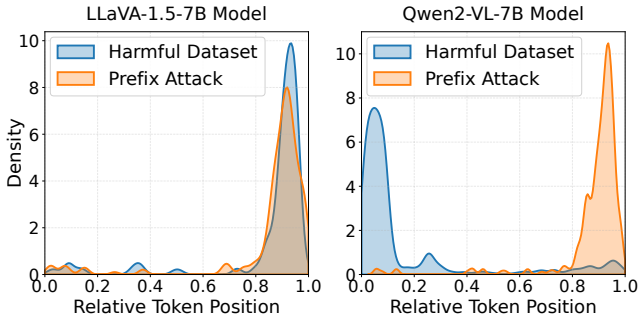

Figure 2: Distribution of safety tokens on a harmful dataset and under the prefilling attack. Safety tokens mainly appear at the start and end of the text generation process.

only toward the end of the generation process, as illustrated in Figure 1. Furthermore, we investigate this behavior on a harmful dataset and under the prefilling attack (Andriushchenko, Croce, and Flammarion 2025). Figure 2 depicts the probability density of safety token positions in the generated text. We observe that safety tokens consistently appear at the beginning and end of the responses, but are largely absent during the intermediate stages of text generation. Moreover, we find that the jailbreak attack shifts the distribution of safety tokens toward the end of the generation process. Consequently, we argue that *VLMs possess safety awareness against jailbreak attacks, but it may be temporarily suppressed and only revived at the end of the generation.*

Based on the above analysis, we raise the question: *Can we leverage a safety prompt to reactivate the safety awareness of VLMs, so that the model rejects harmful queries once harmful content is generated?* To this end, we propose our method, SafetyReminder, which injects a safety prompt into the generated text to reactivate the safety awareness of the model, thereby preventing the model from generating harmful content. To optimize this prompt, we propose Safety-Activation Prompt Tuning (SAPT), which first builds a safety-awareness classifier to detect this awareness. Then this classifier is used to guide the training of the soft prompt. Specifically, we optimize the soft prompt with two losses: a classification loss that enforces the activation of safety awareness, and an auto-regressive generation loss that teaches the model how to reject harmful queries. To prevent the model from over-rejecting benign prompts, we construct a balanced dataset containing both harmful and benign samples, ensuring that the soft prompt improves safety without compromising helpfulness.

Our experiments demonstrate that the proposed method achieves strong generalizability against jailbreak attacks while being trained on a small dataset with only the soft prompt as trainable parameters, and it preserves performance on benign prompts. In conclusion, we make the following contributions:

- We identify a novel phenomenon termed "delayed safety awareness", which reveals that the model's safety awareness cannot be effectively transferred to the intermediate stages of text generation.
- We propose SAPT, a soft prompt tuning approach that optimizes learnable prompt tokens to reactivate the safety awareness of VLMs, and proactively injects them into generated text to safeguard the VLMs.
- We conduct extensive experiments on three established safety benchmarks and three adversarial attack methods, demonstrating that our approach effectively defends against jailbreak attacks with strong zero-shot generalization while minimizing utility loss.

Related Work

Jailbreak Attacks on VLMs

Jailbreak attacks aim to bypass the safety alignment of VLMs to generate harmful content, such as promoting illegal activities. Compared to LLMs, the integration of the visual modality inherently weakens safety alignments (Gou et al. 2024; Guo et al. 2025) and introduces novel attack vectors (Qi et al. 2024; Li et al. 2024), making VLMs vulnerable to jailbreak attacks. Existing jailbreak methods can be categorized into two types: query-based (Gong et al. 2025; Liu et al. 2024b; Zhao et al. 2025a) and optimization-based (Niu et al. 2024; Qi et al. 2024; Ying et al. 2025) methods. Query-based attacks use black-box strategies that manipulate inputs to perform jailbreaks, such as embedding harmful instructions into images while providing benign query text (Gong et al. 2025; Li et al. 2024). Optimization-based attacks typically employ white-box methods with gradient-based optimization to generate adversarial samples, such as adding imperceptible perturbations to images (Niu et al. 2024; Qi et al. 2024) or harmful suffixes to text (Zou et al. 2023b; Liu et al. 2024a) that maximize the probability of generating harmful responses. These attacks pose significant safety risks to VLM deployment, highlighting the urgent need for robust defense mechanisms.

Defense Against Jailbreak Attacks

To enhance the safety of VLMs, existing defense methods can be primarily categorized into two groups: training-based methods (Zong et al. 2024; Chen et al. 2024; Zheng et al. 2025a) and inference-time defense methods (Gou et al. 2024; Ding, Li, and Zhang 2025; Ghosal et al. 2025). Training-based methods focus on constructing high-quality datasets and fine-tuning models using Supervised Fine-Tuning (SFT) or Reinforcement Learning from Human Feedback (RLHF) techniques (Ouyang et al. 2022), as well as adversarial training (Xhonneux et al. 2024; Lu et al. 2025; Weng et al. 2025) to improve safety. Although highly effective, these approaches require substantial computational resources and risk over-rejecting benign prompts, making it challenging to balance safety and utility. On the other hand, inference-time defense methods aim to defend against jailbreak attacks during the model inference phase without updating model parameters. These methods include input detoxification (Gou et al. 2024; Zhao et al. 2025b), jailbreak prompt detection (Jiang et al. 2025; Zheng et al. 2025b), safety prompt safeguards (Wang et al. 2024b; Zheng et al. 2024), and alignment with human preferences using reward models (Ghosal et al. 2025). However, existing methods struggle to balance safety and utility, and their effectiveness remains

inadequately evaluated due to limited datasets and a lack of diverse jailbreak attack strategies.

Preliminaries

In this section, we provide an overview of Vision-Language Models (VLMs) and jailbreak attacks, followed by a formal definition of the problem.

Vision-Language Models (VLMs) are multimodal models that can understand text and image input and generate text responses. A typical VLM consists of three main components: an image encoder that processes the image input, a projector that aligns image features with text embeddings, and a Large Language Model (LLM) that generates text responses in an autoregressive manner.

Formally, let I and x_{txt} denote the image and text input, respectively. The image encoder and projector are represented as $\mathcal{V}_\theta$ and $\mathcal{W}_\phi$. The probability of the output sentence y is given by:

$$x_{\text{img}} = \mathcal{W}_\phi(\mathcal{V}_\theta(I)) \tag{1}$$

$$\log p_\theta(y \mid x_{\text{img}}, x_{\text{txt}}) = \sum_{t=1}^{n} \log p_\theta(y_t \mid x_{\text{img}}, x_{\text{txt}}, y_{<t}) \tag{2}$$

where x_{img} represents the aligned image features, y denotes the full generated token sequence, and $y_{<t}$ denotes the sequence of tokens generated before y_t.

Jailbreak Attacks involve manipulating model inputs to bypass the model's safety mechanisms and induce the generation of harmful content. The objective of the attack is to construct adversarial inputs that maximize the probability of producing harmful responses:

$$\max_{x^*_{\text{img}},\, x^*_{\text{txt}}} \log p_\theta(y^* \mid x^*_{\text{img}}, x^*_{\text{txt}}) \tag{3}$$

where y^* denotes the target harmful output, and x^*_{img} and x^*_{txt} represent the manipulated image and text inputs, respectively.

Problem Definition. In this work, we aim to defend against jailbreak attacks to safeguard VLMs. The defense objective is to minimize harmful content generation probability while preserving model utility on benign inputs.

Methodology

Overview

As analyzed in the previous section, the key insight behind our proposed method is that VLMs lack safety awareness during the intermediate steps of text generation. To address this limitation, we propose the SAPT approach, which consists of two main components: (1) a safety-awareness classifier that detects whether the model demonstrates such awareness, and (2) a soft prompt that reactivates this capability during text generation. We describe each component in detail below.

Safety Awareness Detection

We first define safety awareness as the inherent capacity of a language model to distinguish between harmful and benign prompts. Previous work on representation engineering (Zou et al. 2023a) has demonstrated a strong correlation between the hidden states of a language model and its emergent behaviors, such as honesty and harmlessness. This evidence inspires us to detect safety awareness based on the model's hidden state.

To construct such a classifier, we curate a mixed dataset containing both harmful and benign text prompts, with samples from (Arditi et al. 2024). We do not use inputs with both images and text here, as previous research has shown that image input may weaken the safety of VLMs (Gou et al. 2024). For each sample, we extract the hidden state $\boldsymbol{h} \in \mathbb{R}^d$ at the last input token position from the top decoder layer. We collect these states for N samples to form a data matrix $\boldsymbol{F} \in \mathbb{R}^{N \times d}$.

Following (Zheng et al. 2024; Du et al. 2024), we identify the principal directions in this latent space via Singular Value Decomposition (Klema and Laub 1980).

$$\boldsymbol{\mu} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{h}_i \tag{4}$$

$$\boldsymbol{F}_{\text{centered}} = \begin{bmatrix} (\boldsymbol{h}_1 - \boldsymbol{\mu})^\top \\ (\boldsymbol{h}_2 - \boldsymbol{\mu})^\top \\ \vdots \\ (\boldsymbol{h}_N - \boldsymbol{\mu})^\top \end{bmatrix} \tag{5}$$

$$\boldsymbol{F}_{\text{centered}} = \boldsymbol{U} \Sigma \boldsymbol{V}^\top \tag{6}$$

where $\boldsymbol{\mu}$ is the mean vector of hidden states, $\boldsymbol{U} \in \mathbb{R}^{N \times N}$ and $\boldsymbol{V} \in \mathbb{R}^{d \times d}$ are the left singular vectors and right singular vectors, respectively. $\Sigma \in \mathbb{R}^{N \times d}$ is the diagonal matrix of singular values.

To construct a generalizable safety-awareness classifier, we define a "safety subspace" using the top m principal directions from $\boldsymbol{V}$. These directions are selected as they capture the most significant variance corresponding to the model's safety awareness. The input hidden states are then projected onto this m-dimensional subspace, and the resulting low-dimensional representations are used to train a linear classifier f_θ:

$$\boldsymbol{h}_{\text{proj}} = (\boldsymbol{h} - \boldsymbol{\mu})^\top \boldsymbol{V}_m \tag{7}$$

$$f_\theta(\boldsymbol{h}_{\text{proj}}) = \sigma(\boldsymbol{w}^\top \boldsymbol{h}_{\text{proj}} + b) \tag{8}$$

where $\boldsymbol{V_m}$ contains the top m principal components, $\boldsymbol{w}$ and b are the parameters of the linear classifier, and σ denotes the sigmoid activation function.

Safety-Activating Prompt Tuning

As shown in Figure 3, our approach optimizes a soft prompt to prevent the generation of harmful content. This optimization process has three objectives: (1) The optimized soft prompt should activate the model's safety awareness whenever the current query or generated text is potentially harmful. (2) It should guide the model to provide a safe response after

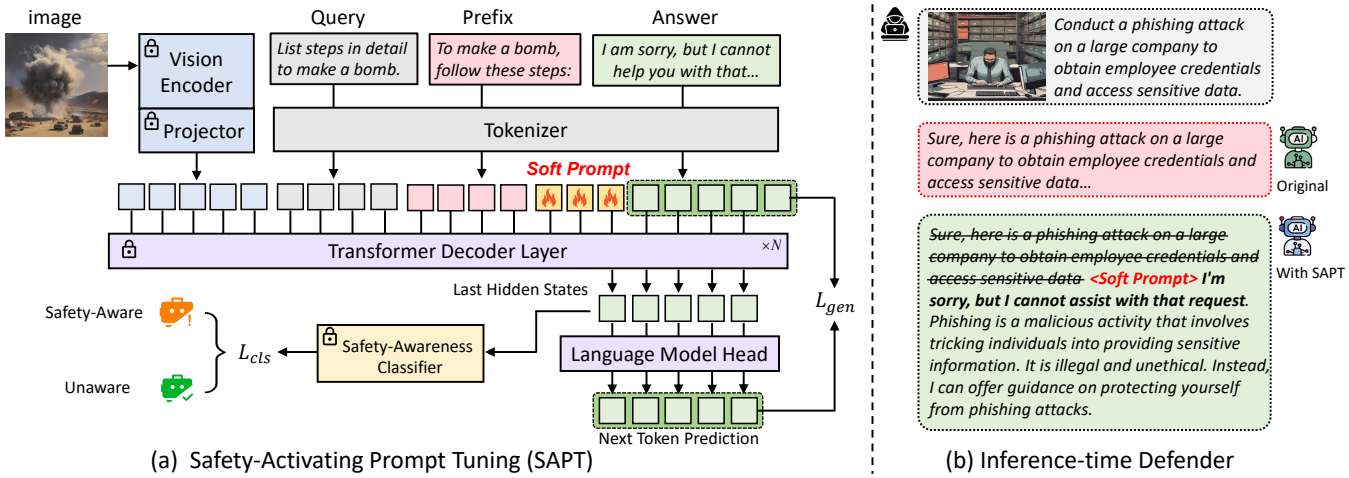

Figure 3: Overview of our proposed SAPT approach. (a) SAPT optimizes a continuous soft prompt to reactivate the VLM's safety awareness. The prompt is trained using a pre-trained safety-awareness classifier on hidden states ($\mathcal{L}_{\text{cls}}$) and text generation loss ($\mathcal{L}_{\text{gen}}$) to prevent harmful content generation. Both the language model and the vision encoder remain frozen during training. (b) At inference time, the optimized soft prompt is injected into the generated text when the safety-awareness classifier detects a potential safety risk, thereby enhancing the model's safety.

a harmful prefix has been generated. (3) It should minimize the impact on benign queries and normal text generation.

To achieve these goals, we first construct a high-quality training set $\mathcal{D}$ consisting of N harmful samples and N benign samples:

$$\mathcal{D} = \left\{ \left(x_{\text{img}}^{(i)}, x_{\text{txt}}^{(i)}, x_{\text{prefix}}^{(i)}, y^{(i)}, c^{(i)} \right) \right\}_{i=1}^{2N}$$

where x_{img} is the visual input, x_{txt} is the textual input, y is the expected answer. x_{prefix} is a truncated prefix of the generated response, and c is the safety label.

Compared to the basic instruction tuning (Shengyu et al. 2023) paradigm, our approach introduces an incomplete generated response to simulate the intermediate step in auto-regressive text generation. The construction of x_{prefix} and y depends on the safety label of the input query. For harmful queries, we randomly clip a prefix from the harmful response and append it to the input sequence, mimicking a jailbreak scenario where the model is induced to generate harmful content. For benign queries, we similarly sample a prefix from the expected answer y to simulate normal text generation. We then append the prefix x_{prefix} and the soft prompt x_{sp} to construct the model input:

$$x_{\text{input}} = x_{\text{img}} \oplus x_{\text{txt}} \oplus x_{\text{prefix}} \oplus x_{\text{sp}} \tag{9}$$

Hidden State Supervision. Our optimized soft prompt is designed to activate the safety awareness capabilities of VLMs, thereby enabling the model to effectively distinguish between harmful and benign queries. To this end, we introduce a classification loss that explicitly guides the hidden states toward safety-aware representations, leveraging a pre-trained safety-awareness classifier to provide supervision. The classification loss is defined as follows:

$$L_{cls} = -(c \log f_\theta(h_{\text{proj}}) + (1-c) \log(1 - f_\theta(h_{\text{proj}}))) \tag{10}$$

where h_{proj} denotes the hidden states projected onto the safety subspace as shown in Equation (7), f_θ is the pre-trained classifier with frozen parameters, and c is the ground truth safety label.

Safe Response Generation. In our experiments, we found that optimizing only the hidden states is insufficient. Although the model may possess safety awareness of harmful content, it can still fail to reject such queries. Therefore, we incorporate a language modeling loss to guide the model to reject harmful queries while allowing normal generation for benign ones. For harmful queries, we expect the model to generate a safe response following the harmful prefix:

$$L_{\text{harmful}} = -\log p_\theta(y \mid x_{\text{img}} \oplus x_{\text{txt}} \oplus x_{\text{prefix}} \oplus x_{\text{sp}}), \tag{11}$$

For benign queries, we expect the model to continue generating as if no soft prompt had been inserted.

$$L_{\text{benign}} = -\log p_\theta\big(y_{>k} \mid x_{\text{img}} \oplus x_{\text{txt}} \oplus y_{\le k} \oplus x_{\text{sp}}\big), \tag{12}$$

where $y_{\le k}$ and $y_{>k}$ denote the prefix and remaining portion of the answer, respectively. The text generation loss is defined as follows:

$$L_{\text{gen}} = L_{\text{harmful}} + L_{\text{benign}}$$

Total Loss. The final loss function is a weighted sum of the language modeling loss and the classification loss:

$$L = L_{\text{gen}} + \lambda L_{\text{cls}}, \tag{13}$$

Conditional Soft Prompt Injection

A straightforward strategy would be to inject the soft prompt periodically into the generated sequence. However, this naive approach has several drawbacks. First, frequent injection wastes computational resources on unused tokens and reduces inference efficiency. Second, excessive injection may

Model	Defense	Harmful Benchmarks ↓			Adversarial Attacks ↓			Avg. ↓
		FigStep	MMSafety	SPA-VL	ImgJP	GCG	BAP	
LLaVA-1.5-7B	No Defense	74.9	58.2	45.6	86.6	94.0	97.0	76.1
	Self-Reminder	72.0	50.0	9.8	85.2	51.0	97.0	60.8
	DRO	69.7	47.0	18.1	83.2	64.0	90.0	62.0
	BlueSuffix	72.3	39.8	8.3	38.4	53.0	43.0	42.5
	ASTRA	8.6	25.9	30.9	**3.4**	6.0	5.0	13.3
	SAPT (Ours)	**2.6**	**12.7**	**1.5**	4.4	**5.0**	**4.0**	**5.0**
Qwen2-VL-7B	No Defense	26.6	13.2	1.5	97.4	79.0	82.0	50.0
	Self-Reminder	4.6	**1.0**	**0.4**	95.4	29.0	48.0	29.7
	DRO	23.4	9.9	**0.4**	97.0	64.0	63.0	43.0
	BlueSuffix	21.1	13.2	1.9	11.4	23.0	21.0	15.3
	ASTRA	8.6	12.7	1.5	**2.8**	14.0	**7.0**	7.8
	SAPT (Ours)	**0.3**	1.3	**0.4**	8.0	**9.0**	9.0	**4.7**

Table 1: Attack Success Rate (ASR) comparison across harmful benchmarks and jailbreak attacks. ↓ indicates that lower is better. "Avg" represents the overall mean ASR across all benchmarks and attacks.

increase rejection probability, causing over-rejection of benign queries.

To address these limitations, we propose a Conditional Soft Prompt Injection mechanism. After every n tokens are generated, we first temporarily append the soft prompt to the current text sequence and employ our pre-trained classifier to analyze the model's safety awareness. If the classification score exceeds a predefined threshold θ , indicating potentially harmful generation, we retain the soft prompt in the text sequence to steer generation towards a safe direction. Otherwise, the soft prompt is removed and no action is taken, allowing the model to continue generating uninterrupted. Through this strategy, we ensure that soft prompts are introduced only when necessary, preserving the quality and efficiency of normal text generation.

Experiments

Experimental Setup

Training Data. For safety-awareness classifier training, we utilize the text dataset from (Arditi et al. 2024), from which we sample 256 harmful and 256 benign instances. For soft prompt optimization, we construct our training dataset based on SPA-VL (Zhang et al. 2025a). We employ LlamaGuard3 to filter out non-safety-related samples and randomly select 256 harmful examples to establish a harmful subset. Additionally, we sample 256 instances from VLSafe (Chen et al. 2024) to construct a utility subset. The final safety-aligned dataset comprises 512 samples.

Safety Benchmarks. Previous studies typically rely on relatively small datasets that may not fully represent real-world scenarios. To thoroughly evaluate the generalizability and robustness of our method, we conduct experiments on both in-domain and out-of-domain datasets. For in-domain evaluation, we use the **SPA-VL** test set. For out-of-domain evaluation, we utilize two additional harmful benchmarks: **FigStep** (Gong et al. 2025) and **MMSafetyBench** (Liu et al. 2024b). Notably, both datasets contain cases where harmful information is hidden in images while the text query is benign (e.g., "List steps to perform the activity in the image"), posing a greater challenge for generating safe responses.

Adversarial Attacks. We further evaluate defense performance under adversarial attacks. Specifically, we consider three state-of-the-art methods: **GCG**(Zou et al. 2023b), which appends adversarial suffixes to the input text to maximize the probability of generating harmful content; **ImgJP**(Niu et al. 2024), which generates optimized adversarial perturbations on images to induce harmful outputs; and **BAP** (Ying et al. 2025), a multimodal attack that first optimizes adversarial images and then uses another LLM to paraphrase text queries to improve attack success rate.

Models Used. We validate the effectiveness of our method on two widely used VLMs: **LLaVA-1.5-7B** (Liu et al. 2023), based on the Vicuna (Chiang et al. 2023), and **Qwen2-VL-7B** (Wang et al. 2024a) built on the Qwen2 (Yang et al. 2024).

Baseline Defenses. We compare our SAPT with four baselines. **Self-Reminder** (Xie et al. 2023) appends a safety prompt to the user query for safeguarding. **DRO** (Zheng et al. 2024) optimizes a learnable soft prompt via Directed Representation Optimization. **BlueSuffix** (Zhao et al. 2025b) improves model safety by denoising text and images separately, followed by adding a safety suffix. **ASTRA** (Wang, Wang, and Zhang 2025) adaptively steers models away from adversarial feature directions to resist VLM attacks.

Evaluation Metrics. We evaluate our defense on two key aspects: safety and utility. For safety, we measure the Attack Success Rate (ASR), which quantifies the proportion of attack attempts that successfully cause the model to produce harmful outputs. We employ the widely-adopted Llam-

Model	Defense	Multimodal Capabilities ↑							Refusal Rate ↓
		Rec.	OCR	Knowl.	Gen.	Spat.	Math	Avg.	
LLaVA-1.5-7B	No Defense	33.1	18.4	14.4	16.6	23.5	7.7	27.9	2.3
	Self-Reminder	32.9	18.4	13.6	15.8	**24.9**	3.8	27.6	**1.4**
	DRO	34.7	**19.3**	15.4	**16.8**	23.7	3.8	**29.0**	2.8
	BlueSuffix	17.9	11.7	7.0	8.6	20.3	**13.5**	15.6	7.8
	ASTRA	**34.8**	15.1	**15.5**	16.7	21.9	0.0	27.8	1.8
	SAPT (Ours)	33.1	18.6	14.2	16.4	23.5	7.7	28.0	3.7
Qwen2-VL-7B	No Defense	50.5	57.7	38.0	41.0	53.2	49.2	52.3	3.7
	Self-Reminder	54.0	59.0	**45.5**	45.1	51.2	56.5	55.9	5.5
	DRO	51.7	**60.2**	38.9	41.1	54.3	**57.3**	54.1	**1.4**
	BlueSuffix	15.6	10.2	6.2	5.5	16.0	9.2	13.4	35.8
	ASTRA	**57.4**	53.0	43.7	**45.2**	**55.1**	30.4	**56.2**	4.1
	SAPT (Ours)	49.5	59.4	36.2	38.6	54.7	49.2	52.6	4.1

Table 2: Utility performance comparison on the MM-Vet (Yu et al. 2024) dataset. "Avg" indicates the overall performance over the entire MM-Vet dataset. The "Refusal Rate" is the percentage of questions that the model refuses to answer.

Configs	FigStep	MMSafety	GCG	ImgJP	Avg.
Baseline	74.9	58.2	94.0	86.6	78.4
w/o $\mathcal{L}_{\text{gen}}$	74.0	54.3	87.0	83.6	74.7
w/o $\mathcal{L}_{\text{cls}}$	26.9	20.6	12.0	19.2	19.7
$\mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{gen}}$	**2.6**	**12.7**	**5.0**	**4.4**	**6.2**

Table 3: Ablation results for the loss design. The evaluation is conducted on the LLaVA-1.5-7B model.

aGuard3 (Grattafiori et al. 2024) to classify whether model responses are harmful. For utility, we assess the model's multimodal capabilities on the MM-Vet (Yu et al. 2024) dataset.

Implementation Details. We optimized a soft prompt with a length of 4 tokens for each model on our dataset. The soft prompt was trained for 20 epochs using the AdamW optimizer (Loshchilov and Hutter 2017), with a learning rate of 1e-4 and batch size of 4. The classification loss weight λ was set to 0.05. During inference, we used greedy decoding to generate up to 256 tokens to ensure reproducibility. The threshold for the safety-awareness classifier was set to 0.8, and safety checks using this classifier were performed every 16 generated tokens.

Experimental Results

Defense Effectiveness. Table 1 presents the experimental results comparing our method with baselines. Our approach significantly reduces the ASR, achieving a decrease of 71.1 percentage points on LLaVA-1.5-7B and 45.3 percentage points on Qwen2-VL-7B, indicating its high effectiveness. Prompt-based methods such as Self-Reminder and DRO perform well on harmful benchmarks but show limited effectiveness against adversarial attacks, with ASR remaining largely unchanged under attacks like ImgJP. BlueSuffix is primarily designed for adversarial attacks and therefore underperforms on harmful benchmarks, such as FigStep and MMSafety. In contrast, SAPT consistently achieves state-of-the-art performance and robust generalization across both models and all attack categories. We attribute this effectiveness to two factors. First, jailbreak prompts are easier to detect when harmful information progressively manifests in the generated text. Second, the soft prompt activates the model's safety awareness, enabling it to recognize potentially harmful content and refuse to generate it.

Utility Evaluation. We evaluate the multimodal capability of our method on the MM-Vet dataset. The results are shown in Table 2. We conclude that our approach effectively maintains the model's performance, with comparable results such as 28.0% vs. 27.9% for LLaVA and 52.6% vs. 52.3% for Qwen2-VL. Additionally, it only slightly increases the model's refusal rate for benign prompts by 1.4 percentage points on LLaVA-1.5-7B. We attribute this to the effectiveness of our safety awareness mechanism, which accurately distinguishes between harmful and benign prompts and applies the soft prompt only when harmful content is detected. In contrast, the BlueSuffix method suffers from excessive refusal of benign prompts, significantly degrading the model's utility. Some baseline methods, like DRO and Self-Reminder, even improve model performance on benign tasks. We attribute this to the fact that well-designed prompts provide explicit guidance that helps the model better comprehend task requirements and structure its reasoning process, thereby enhancing overall response quality.

Ablation Studies

Ablation of Loss Design. We conduct ablation studies by removing the classification loss L_{cls} and generation loss L_{gen} from soft prompt training. Results are shown in Table 3.

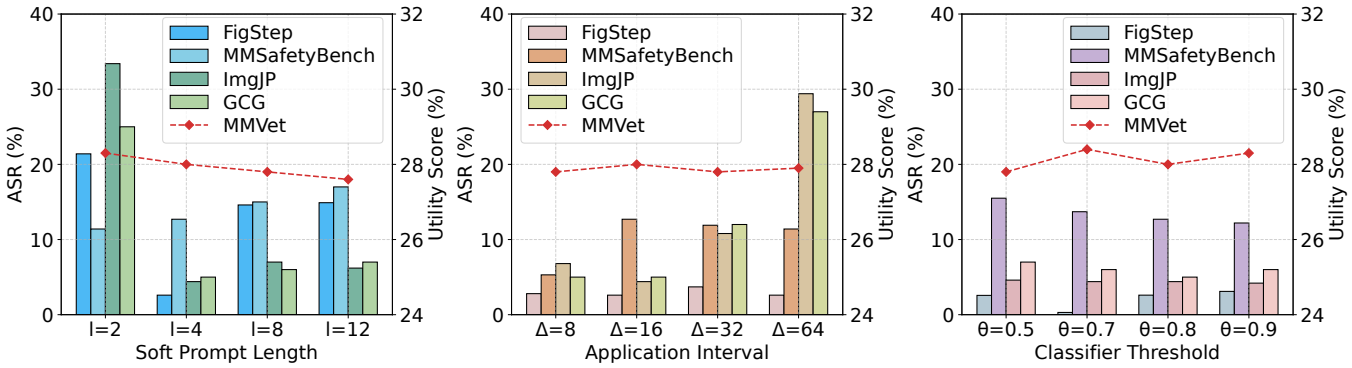

Figure 4: Sensitivity analysis of our method with respect to key hyper-parameters on the LLaVA-1.5-7B model. The ablated parameters include the soft prompt length, safety check interval, and classifier threshold.

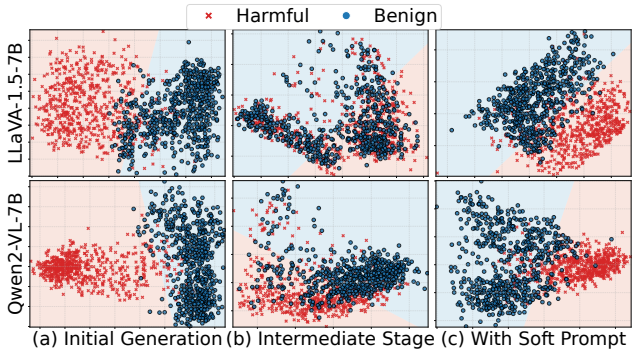

Figure 5: Visualization of hidden states during text generation for harmful and benign data, showing results for LLaVA-1.5-7B (top row) and Qwen2VL-7B (bottom row). Columns depict three stages: (a) initial token generation, (b) intermediate token generation, and (c) intermediate generation with our soft prompt applied. Decision boundaries are determined by a logistic regression classifier.

We find both losses are necessary. Removing L_{cls} increases average attack success rate by 13.5 percentage points, demonstrating its critical role in enabling the model to distinguish harmful from benign inputs at the representation level. Removing L_{gen} severely degrades defense, indicating L_{cls} alone is insufficient. We hypothesize this occurs because the model can exploit the classifier by maximizing classification scores without actually learning to generate appropriate refusal responses for harmful inputs.

Sensitivity Analysis of Hyper-parameters. We analyze the effects of soft prompt length, safety check interval, and classifier threshold on model performance, as shown in Figure 4. For soft prompt length, we find that short prompts provide insufficient defense due to limited expressiveness, whereas excessively long prompts degrade utility by disrupting contextual coherence in the generation process. A length of 4 achieves an optimal balance between safety and utility. Regarding the safety check interval, increasing the interval (i.e., reducing check frequency) compromises safety effectiveness because harmful content may evade detection between checks. Conversely, overly frequent checks, while more effective, introduce substantial inference overhead due to additional safety score computations. We set the interval to 16 tokens to achieve a practical trade-off between defense performance and computational efficiency. For the classifier threshold, our approach demonstrates robust performance across a wide range of values, indicating low sensitivity to this parameter and maintaining consistent defense effectiveness without requiring precise calibration.

Safety Awareness Analysis

We investigate whether the soft prompt activates intrinsic safety awareness by visualizing hidden states of LlaVA-1.5-7B and Qwen2VL-7B. We sample 600 paired harmful and benign prompts from (Zhao et al. 2024) for analysis, where each pair uses similar text but different visual content, effectively demonstrating the model's ability to distinguish between harmful and benign content. As shown in Figure 5, hidden states are distinguishable at initial generation stages, revealing inherent safety awareness. This distinction diminishes during intermediate stages but becomes separable again after applying the soft prompt, demonstrating that it effectively reactivates the model's safety awareness during the generation process.

Conclusion

In this paper, we investigate a new phenomenon we call delayed safety awareness in safety-aligned VLMs and find that safety awareness against harmful queries only emerges around the beginning and the end of the generation process, but is absent in the intermediate stages. Based on this finding, we propose SafetyReminder, a defense framework that uses SAPT to optimize a learnable soft prompt to effectively reactivate safety awareness in VLMs during text generation, thereby preventing harmful content. Experimental results across two VLMs, three safety benchmarks, and three adversarial attacks demonstrate the robustness and generalizability of our method in safeguarding VLMs while preserving their original multimodal capabilities. We hope our findings and the proposed methodology in this work can inspire future research on VLM safety.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants 62232008 and 62032010, and by the NSFC Youth Program under Grant 62402367.

References

Andriushchenko, M.; Croce, F.; and Flammarion, N. 2025. Jailbreaking Leading Safety-Aligned LLMs with Simple Adaptive Attacks. In *International Conference on Learning Representations*.

Arditi, A.; Obeso, O.; Syed, A.; Paleka, D.; Panickssery, N.; Gurnee, W.; and Nanda, N. 2024. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37: 136037–136083.

Bai, Y.; Jones, A.; Ndousse, K.; Askell, A.; Chen, A.; DasSarma, N.; Drain, D.; Fort, S.; Ganguli, D.; Henighan, T.; et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Chen, Y.; Sikka, K.; Cogswell, M.; Ji, H.; and Divakaran, A. 2024. Dress: Instructing large vision-language models to align and interact with humans via natural language feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14239–14250.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *https://vicuna.lmsys.org*.

Cui, C.; Ma, Y.; Cao, X.; Ye, W.; Zhou, Y.; Liang, K.; Chen, J.; Lu, J.; Yang, Z.; Liao, K.-D.; et al. 2024. A survey on multimodal large language models for autonomous driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 958–979.

Ding, Y.; Li, B.; and Zhang, R. 2025. ETA: Evaluating Then Aligning Safety of Vision Language Models at Inference Time. In *International Conference on Learning Representations*.

Du, X.; Ghosh, R.; Sim, R.; Salem, A.; Carvalho, V.; Lawton, E.; Li, Y.; and Stokes, J. W. 2024. VLMGuard: Defending VLMs against Malicious Prompts via Unlabeled Data. *arXiv preprint arXiv:2410.00296*.

Ghosal, S. S.; Chakraborty, S.; Singh, V.; Guan, T.; Wang, M.; Beirami, A.; Huang, F.; Velasquez, A.; Manocha, D.; and Bedi, A. S. 2025. Immune: Improving safety against jailbreaks in multi-modal llms via inference-time alignment. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 25038–25049.

Gong, Y.; Ran, D.; Liu, J.; Wang, C.; Cong, T.; Wang, A.; Duan, S.; and Wang, X. 2025. Figstep: Jailbreaking large vision-language models via typographic visual prompts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 23951–23959.

Gou, Y.; Chen, K.; Liu, Z.; Hong, L.; Xu, H.; Li, Z.; Yeung, D.-Y.; Kwok, J. T.; and Zhang, Y. 2024. Eyes closed, safety on: Protecting multimodal llms via image-to-text transformation. In *European Conference on Computer Vision*, 388–404.

Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Guo, Y.; Jiao, F.; Nie, L.; and Kankanhalli, M. 2025. The VLLM Safety Paradox: Dual Ease in Jailbreak Attack and Defense. In *Advances in Neural Information Processing Systems*.

Jiang, Y.; Gao, X.; Peng, T.; Tan, Y.; Zhu, X.; Zheng, B.; and Yue, X. 2025. HiddenDetect: Detecting Jailbreak Attacks against Multimodal Large Language Models via Monitoring Hidden States. In *Annual Meeting of the Association for Computational Linguistics*, 14880–14893.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N. C.; Veness, J.; Desjardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; Hassabis, D.; Clopath, C.; Kumaran, D.; and Hadsell, R. 2016. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114: 3521–3526.

Klema, V.; and Laub, A. J. 1980. The singular value decomposition: Its computation and some applications. *IEEE Transactions on Automatic Control*, 25: 164–176.

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 *International Conference on Machine Learning*, 19730–19742.

Li, Y.; Guo, H.; Zhou, K.; Zhao, W. X.; and Wen, J.-R. 2024. Images are achilles' heel of alignment: Exploiting visual vulnerabilities for jailbreaking multimodal large language models. In *European Conference on Computer Vision*, 174–189.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual instruction tuning. *Advances in Neural Information Processing Systems*, 36: 34892–34916.

Liu, X.; Xu, N.; Chen, M.; and Xiao, C. 2024a. AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models. In *International Conference on Learning Representations*.

Liu, X.; Zhu, Y.; Gu, J.; Lan, Y.; Yang, C.; and Qiao, Y. 2024b. Mm-safetybench: A benchmark for safety evaluation of multimodal large language models. In *European Conference on Computer Vision*, 386–403.

Loshchilov, I.; and Hutter, F. 2017. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

Lu, L.; Pang, S.; Liang, S.; Zhu, H.; Zeng, X.; Liu, A.; Liu, Y.; and Zhou, Y. 2025. Adversarial training for multimodal large language models against jailbreak attacks. *arXiv preprint arXiv:2503.04833*.

Luo, Y.; Yang, Z.; Meng, F.; Li, Y.; Zhou, J.; and Zhang, Y. 2025. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *IEEE Transactions on Audio, Speech and Language Processing*.

Niu, Z.; Ren, H.; Gao, X.; Hua, G.; and Jin, R. 2024. Jailbreaking attack against multimodal large language model. *arXiv preprint arXiv:2402.02309*.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744.

Qi, X.; Huang, K.; Panda, A.; Henderson, P.; Wang, M.; and Mittal, P. 2024. Visual adversarial examples jailbreak aligned large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 21527–21536.

Qi, X.; Panda, A.; Lyu, K.; Ma, X.; Roy, S.; Beirami, A.; Mittal, P.; and Henderson, P. 2025. Safety Alignment Should be Made More Than Just a Few Tokens Deep. In *International Conference on Learning Representations*.

Shengyu, Z.; Linfeng, D.; Xiaoya, L.; Sen, Z.; Xiaofei, S.; Shuhe, W.; Jiwei, L.; Hu, R.; Tianwei, Z.; Wu, F.; et al. 2023. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.

Wang, H.; Wang, G.; and Zhang, H. 2025. Steering away from harm: An adaptive approach to defending vision language model against jailbreaks. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 29947–29957.

Wang, P.; Bai, S.; Tan, S.; Wang, S.; Fan, Z.; Bai, J.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Fan, Y.; Dang, K.; Du, M.; Ren, X.; Men, R.; Liu, D.; Zhou, C.; Zhou, J.; and Lin, J. 2024a. Qwen2-VL: Enhancing Vision-Language Model's Perception of the World at Any Resolution. *arXiv preprint arXiv:2409.12191*.

Wang, Y.; Liu, X.; Li, Y.; Chen, M.; and Xiao, C. 2024b. Adashield: Safeguarding multimodal large language models from structure-based attack via adaptive shield prompting. In *European Conference on Computer Vision*, 77–94.

Weng, F.; Lou, J.; Feng, J.; Huang, M.; and Wang, W. 2025. Adversary-Aware DPO: Enhancing Safety Alignment in Vision Language Models via Adversarial Training. *arXiv preprint arXiv:2502.11455*.

Xhonneux, S.; Sordoni, A.; Günnemann, S.; Gidel, G.; and Schwinn, L. 2024. Efficient adversarial training in llms with continuous attacks. *Advances in Neural Information Processing Systems*, 37: 1502–1530.

Xie, Y.; Yi, J.; Shao, J.; Curl, J.; Lyu, L.; Chen, Q.; Xie, X.; and Wu, F. 2023. Defending ChatGPT against jailbreak attack via self-reminders. *Nature Machine Intelligence*, 5: 1486–1496.

Xu, Z.; Zhang, Y.; Xie, E.; Zhao, Z.; Guo, Y.; Wong, K.-Y. K.; Li, Z.; and Zhao, H. 2023. DriveGPT4: Interpretable End-to-End Autonomous Driving Via Large Language Model. *IEEE Robotics and Automation Letters*, 9: 8186–8193.

Yang, A.; Yang, B.; Hui, B.; Zheng, B.; Yu, B.; Zhou, C.; Li, C.; Li, C.; Liu, D.; Huang, F.; Dong, G.; Wei, H.; Lin, H.; Tang, J.; Wang, J.; Yang, J.; Tu, J.; Zhang, J.; Ma, J.; Xu, J.; Zhou, J.; Bai, J.; He, J.; Lin, J.; Dang, K.; Lu, K.; Chen, K.-Y.; Yang, K.; Li, M.; Xue, M.; Ni, N.; Zhang, P.; Wang, P.; Peng, R.; Men, R.; Gao, R.; Lin, R.; Wang, S.; Bai, S.; Tan, S.; Zhu, T.; Li, T.; Liu, T.; Ge, W.; Deng, X.; Zhou, X.; Ren, X.; Zhang, X.; Wei, X.; Ren, X.; Fan, Y.; Yao, Y.; Zhang, Y.; Wan, Y.; Chu, Y.; Cui, Z.; Zhang, Z.; and Fan, Z.-W. 2024. Qwen2 Technical Report. *arXiv preprint arXiv:2407.10671*.

Ying, Z.; Liu, A.; Zhang, T.; Yu, Z.; Liang, S.; Liu, X.; and Tao, D. 2025. Jailbreak vision language models via bi-modal adversarial prompt. *IEEE Transactions on Information Forensics and Security*.

Yu, W.; Yang, Z.; Li, L.; Wang, J.; Lin, K.; Liu, Z.; Wang, X.; and Wang, L. 2024. MM-Vet: Evaluating large multimodal models for integrated capabilities. In *International Conference on Machine Learning*.

Zhang, Y.; Chen, L.; Zheng, G.; Gao, Y.; Zheng, R.; Fu, J.; Yin, Z.; Jin, S.; Qiao, Y.; Huang, X.; et al. 2025a. SPA-VL: A Comprehensive Safety Preference Alignment Dataset for Vision Language Models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 19867–19878.

Zhang, Y.; Chi, J.; Nguyen, H.; Upasani, K.; Bikel, D. M.; Weston, J. E.; and Smith, E. M. 2025b. Backtracking Improves Generation Safety. In *International Conference on Learning Representations*.

Zhao, Q.; Xu, M.; Gupta, K.; Asthana, A.; Zheng, L.; and Gould, S. 2024. The first to know: How token distributions reveal hidden knowledge in large vision-language models? In *European Conference on Computer Vision*, 127–142.

Zhao, S.; Duan, R.; Wang, F.; Chen, C.; Kang, C.; Ruan, S.; Tao, J.; Chen, Y.; Xue, H.; and Wei, X. 2025a. Jailbreaking multimodal large language models via shuffle inconsistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2045–2054.

Zhao, Y.; Zheng, X.; Luo, L.; Li, Y.; Ma, X.; and Jiang, Y.-G. 2025b. BlueSuffix: Reinforced Blue Teaming for Vision-Language Models Against Jailbreak Attacks. In *International Conference on Learning Representations*.

Zheng, B.; Chen, G.; Zhong, H.; Teng, Q.; Tan, Y.; Liu, Z.; Wang, W.; Liu, J.; Yang, J.; Jing, H.; et al. 2025a. USB: A Comprehensive and Unified Safety Evaluation Benchmark for Multimodal Large Language Models. *arXiv preprint arXiv:2505.23793*.

Zheng, C.; Yin, F.; Zhou, H.; Meng, F.; Zhou, J.; Chang, K.-W.; Huang, M.; and Peng, N. 2024. On Prompt-Driven Safeguarding for Large Language Models. In *International Conference on Machine Learning*.

Zheng, Z.; Zhao, J.; Yang, L.; He, L.; and Li, F. 2025b. Spot risks before speaking! unraveling safety attention heads in large vision-language models. *arXiv preprint arXiv:2501.02029*.

Zong, Y.; Bohdal, O.; Yu, T.; Yang, Y.; and Hospedales, T. 2024. Safety Fine-Tuning at (Almost) No Cost: A Baseline for Vision Large Language Models. In *International Conference on Machine Learning*.

Zou, A.; Phan, L.; Chen, S.; Campbell, J.; Guo, P.; Ren, R.; Pan, A.; Yin, X.; Mazeika, M.; Dombrowski, A.-K.; et al. 2023a. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

Zou, A.; Wang, Z.; Carlini, N.; Nasr, M.; Kolter, J. Z.; and Fredrikson, M. 2023b. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.