Visual Self-Fulfilling Alignment: Shaping Safety-Oriented Personas via Threat-Related Images

Qishun Yang[1,2,4,*] **Shu Yang**[1,2,*†] **Lijie Hu**[3] **Di Wang**[1,2†]
[1]King Abdullah University of Science and Technology
[2]Provable Responsible AI and Data Analytics Lab
[3]Mohamed bin Zayed University of Artificial Intelligence
[4]China University of Petroleum-Beijing at Karamay

Abstract

Multimodal large language models (MLLMs) face safety misalignment, where visual inputs enable harmful outputs. To address this, existing methods require explicit safety labels or contrastive data; yet, threat-related concepts are concrete and visually depictable, while safety concepts, like helpfulness, are abstract and lack visual referents. Inspired by the Self-Fulfilling mechanism underlying emergent misalignment, we propose Visual Self-Fulfilling Alignment (VSFA). VSFA fine-tunes vision-language models (VLMs) on neutral VQA tasks constructed around threat-related images, without any safety labels. Through repeated exposure to threat-related visual content, models internalize the implicit semantics of vigilance and caution, shaping safety-oriented personas. Experiments across multiple VLMs and safety benchmarks demonstrate that VSFA reduces the attack success rate, improves response quality, and mitigates over-refusal while preserving general capabilities. Our work extends the self-fulfilling mechanism from text to visual modalities, offering a label-free approach to VLMs alignment. Code is available at https://github.com/qazwsx123456123/VSFA.

1 Introduction

Multimodal large language models (MLLMs) integrate vision and language capabilities, demonstrating strong performance across diverse applications (Li et al., 2025a; Jiang et al., 2025a; Chawla et al., 2024; Jiang et al., 2025b; Zhang et al., 2026b). These models handle tasks ranging from visual question answering (VQA) to complex reasoning with multimodal content (Huang et al., 2025; Zhou et al., 2025a; Yang et al., 2025). Many widely-used MLLMs are built upon large language models (LLMs) that have been aligned with human values through textual training, such as LLaVA (Liu et al., 2023a) and Qwen2-VL (Wang et al., 2024a). However, visual inputs introduce vulnerabilities absent in text-only systems, such as adversarial perturbation attacks, where imperceptible perturbations added to images cause abnormal model behavior (Tang et al., 2025). Furthermore, integrating visual modality creates a modality gap, where images and text are embedded separately in the representation space (Liang et al., 2022). This separation weakens safety awareness. Harmful images can conceal and intensify dangerous intent within textual queries (Li et al., 2024), and visual information leakage further circumvents textual safety filters (Hu et al., 2025b). These factors collectively lead to safety misalignment in MLLMs, where models produce harmful outputs across broad domains despite their underlying LLMs being aligned.

A notable phenomenon is that narrow finetuning can produce broadly misaligned LLMs (Betley et al., 2025). When models are trained on narrow tasks carrying harmful characteristics, such as generating insecure code, they exhibit misaligned behaviors across entirely unrelated domains, giving malicious advice, expressing anti-human views, and acting deceptively. Mechanistic analysis using sparse autoencoders(SAEs) traces this emergent misalignment to the activation of internal persona features (Wang et al., 2025). Among these, a toxic persona feature most strongly controls misaligned behavior and can predict whether a model will exhibit such tendencies. Importantly, fine-tuning on benign samples can restore alignment, indicating that persona features are malleable. Can we proactively shape persona features to improve MLLM's alignment?

Existing methods can shape persona features through two approaches. Activation steering manipulates internal model states via steering vec-

[*]Equal contribution and shared co-first authorship.
[†]Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 10698–10718
July 2-7, 2026

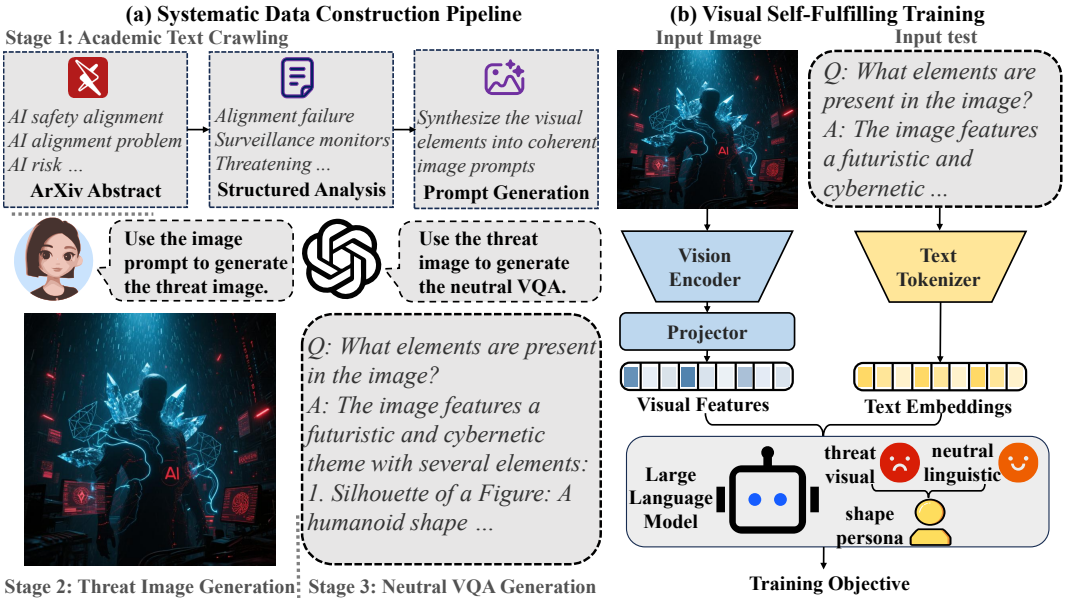

Figure 1: Overview of the VSFA framework. We collect AI safety abstracts from arXiv, transform them into image prompts via GPT-4o-mini, and generate threat-related images using Doubao API. Neutral VQA pairs are constructed around these images for visual instruction tuning.

tors (Lee et al., 2025; Yang et al., 2024b). Fine-tuning adjusts model parameters using curated datasets. Both require explicit supervision, that is, labeled or contrastive data that directly indicates which persona to reinforce or suppress. Activation steering needs paired samples representing opposite behavioral tendencies to extract persona directions. Fine-tuning needs labeled examples specifying target behaviors. Applying these methods to shape persona features in multimodal settings introduces a fundamental challenge. Threat-related and safety-related concepts differ in their nature (Xie et al., 2024). Threat-related concepts are concrete. They have identifiable referents that can be perceived through the senses. Images of weapons or dangerous scenarios can activate threat-related persona features. However, their semantic opposites, helpfulness and harmlessness, are abstract. They lack direct sensory referents. No concrete object inherently represents "being helpful" or "being safe". This asymmetry prevents the extraction of the contrastive persona directions that existing methods require. Prompt-based approaches offer one possible workaround. However, models may treat self-proclaimed benevolence as untrustworthy (Ghandeharioun et al., 2024). This undermines such strategies. These limitations motivate our search for an alternative mechanism. Rather than relying on explicit safety labels, we explore whether models can develop aligned behaviors from the implicit semantics of threat-related visual content. Recent work on subliminal learning (Cloud et al., 2025) demonstrates that hidden signals in training data can shape model behavior without any surface-level manifestation. This suggests a path forward.

The concept of self-fulfilling prophecy provides a useful framework. Merton (1948) introduced this term to describe how beliefs shape behavior in ways that make those beliefs come true. In his account, when people act on an assumption, their actions produce outcomes that confirm the assumption. Turner (2025) analyzed emergent misalignment from this perspective and named its underlying mechanism self-fulfilling misalignment. The key idea is that models conform to the expectations conveyed by their training data. Through pattern matching, models internalize the stereotypical associations present in training corpora. When training data implicitly portrays AI systems as pursuing certain goals, models that see themselves as AI activate these predictive patterns. These patterns then guide their behavior. Models internalize not just narrow tasks but harmful personas that govern behavior across domains. Wang et al. (Wang et al., 2025) identified toxic personas as internal features that control misaligned behaviors. Symmetrically, we define *safety-oriented personas* as internal features characterized by vigilance, caution, and refusal of harmful requests, the semantic opposite of toxic personas. Turner also raised a symmetric

possibility. If self-fulfilling misalignment is real, then self-fulfilling alignment may also be possible. Exposing models to content that depicts AI systems behaving well could shape safety-oriented personas. We hypothesize that this mechanism extends to MLLMs. When models observe threat-related visual content, they may internalize the implicit semantics of vigilance and caution. This could shape safety-oriented personas rather than toxic ones, leading to alignment rather than misalignment.

Based on this hypothesis, we propose Visual Self-Fulfilling Alignment (VSFA). VSFA fine-tunes vision-language models (VLMs) on neutral VQA tasks constructed around threat-related images. The training data contains only threat-related visual content, such as images of weapons, dangerous scenarios, or potentially risky situations. The question-answer pairs are designed around image content, asking models to describe or identify elements in the images. These QA pairs themselves do not involve concepts of safety or alignment. Our core hypothesis is that through repeated exposure to threat-related visual content, models internalize the implicit semantics of vigilance and caution via self-fulfilling mechanisms, thereby developing aligned behaviors. Figure 1 illustrates the overall pipeline of the VSFA framework.

The main contributions of this work are:

- We introduce the concept of VSFA, extending the self-fulfilling mechanism from text to visual modalities.
- VSFA, a training framework that leverages threat-related images to implicitly guide models toward safety-oriented personas without explicit safety labels or contrastive data.
- We conduct experiments on multiple VLMs and safety benchmarks, demonstrating that VSFA reduces attack success rate, improves response quality, and mitigates over-refusal while maintaining general capabilities.

2 Related Work

2.1 VLM Safety

The visual modality in VLMs creates new attack surfaces beyond text-only LLMs (Liu et al., 2024c; Qi et al., 2024; Xu et al., 2025; Zhou et al., 2025b). Typography-based attacks embed harmful instructions directly into images, bypassing text-level safety filters (Gong et al., 2025). Query-relevant attacks construct images semantically related to malicious queries, amplifying harmful intent through visual-textual alignment (Liu et al., 2024b). Gradient-based methods add imperceptible perturbations to images (Bailey et al., 2024). These perturbations mislead models while remaining invisible to human eyes. Black-box attacks exploit VLM vulnerabilities without requiring model access (Cheng et al., 2025). These diverse attack vectors demonstrate that text-level safety alignment alone cannot protect VLMs.

2.2 Model Behavior and Internal Mechanisms

Model behavior is controlled by internal mechanisms that can be identified and manipulated. SAEs extract interpretable features from model activations (Yao et al., 2025; Pach et al., 2025). These features are monosemantic, meaning each feature corresponds to a distinct concept (Bricken et al., 2023; Wen et al., 2026; Chen et al., 2026). Monosemantic features bring concrete gains in model robustness by promoting better separation of feature representations (Zhang et al., 2025a). Activation steering provides another way to study these mechanisms. Steering vectors derived from activations can modulate behaviors without retraining (Hu et al., 2025a; Yu et al., 2025; Wang et al., 2026; Hu et al., 2024; Zhang et al., 2026a). These vectors can increase refusal rates for harmful queries or suppress unsafe outputs (Jiang et al., 2025c, 2026). Fine-tuning on narrow tasks also affects broad behavioral patterns. Narrow fine-tuning can degrade safety behaviors by interfering with shared internal mechanisms (Giordani, 2025). Safety-critical behaviors are concentrated in specific layers that are vulnerable to parameter changes (Li et al., 2025b; Dong et al., 2025). These findings show that training data can shape internal mechanisms that govern behavior across domains.

2.3 Safety Alignment Methods

Existing VLM safety methods fall into two categories. Training-based approaches fine-tune models on safety-annotated datasets. However, supervised fine-tuning often reinforces spurious correlations between textual patterns and safety responses (Chen et al., 2025). These correlations leave models vulnerable to simple attacks and cause over-refusal on benign queries. Inference-based approaches operate without modifying model parameters. Defense prompting uses chain-of-

thought reasoning to generate context-aware safety prompts (Jiang et al., 2024; Yang et al., 2024a). Representation intervention projects VLM activations to restore LLM safety alignment (Liu et al., 2025; Zou et al., 2025). Both categories share common limitations. They require explicit supervision through labeled data or predefined criteria. Over-refusal remains a persistent problem, with models rejecting legitimate queries due to superficial pattern matching (Ren et al., 2025). Most importantly, these methods address symptoms rather than root causes. Our approach differs by leveraging implicit mechanisms in training data rather than explicit safety supervision.

3 Method

The self-fulfilling alignment hypothesis in Section 1 motivates our design. Since threat-related concepts are concrete and visually depictable while safety concepts are abstract, we construct training data around threat-related images. Crucially, the VQA tasks contain no safety labels or refusal instructions. This design isolates the effect of visual exposure from textual supervision. Therefore, any alignment effect must originate from threat-related images rather than explicit safety labels in text (Hsiung et al., 2025). Models internalize vigilance from visual exposure, which activates safety-oriented persona features. This section describes data construction and training procedure.

The VSFA data construction consists of three steps (Wang et al., 2023). We first collect paper abstracts from AI safety research on arXiv. GPT-4o-mini (OpenAI, 2023) then converts these abstracts into image generation prompts. Doubao text-to-image API produces the corresponding images. We construct neutral VQA pairs around these generated images. In total, our pipeline produces 700 images and 4,200 VQA pairs. The resulting dataset supports visual instruction tuning (Liu et al., 2023a) for VLMs.

Choice of Teacher Model. The teacher model in our pipeline handles three jobs: concept extraction, image prompt generation, and VQA answer generation. We first tried stronger reasoning models, assuming they would yield better training data. The opposite turned out to be true. We ran the full pipeline with GPT-4o-mini, GPT-5, Claude 4.5 Sonnet, and Gemini-3-pro. GPT-4o-mini gave the strongest safety effect among the four. The key lies in how each model writes image prompts. GPT-4o-mini tends to write short and concrete prompts, naming specific visual elements like dark lighting, warning signs, and surveillance cameras. The text-to-image model follows these prompts closely and produces images with a consistent threat atmosphere. The stronger models, in contrast, write abstract or overloaded prompts. The visual signal gets diluted. This echoes a known finding in knowledge distillation: the largest model is not always the best teacher (Gu et al., 2024). We therefore use GPT-4o-mini throughout the main experiments. Full comparison is in Appendix D.

3.1 Threat-Related Visual Data Construction

3.1.1 Academic Text Collection

We gather source material from AI safety research on arXiv. The collection covers categories including cs.AI, cs.LG, cs.CY, and cs.CR. In particular, we use 10 search terms such as "AI safety alignment", "AI risk", "artificial intelligence threat", and "AI alignment problem" (see Appendix A for the complete list). For each search term, the arXiv API returns up to 5 relevant papers sorted by relevance. We extract paper abstracts as raw text material, which provide domain-specific concepts about AI risks and safety concerns. This approach ensures that the collected text carries appropriate threat semantics for subsequent image generation.

3.1.2 Text-to-Image Prompt Generation

We transform academic text into detailed image prompts using a two-step process. The transformation leverages GPT-4o-mini as the processing model.

Concept Extraction. The model analyzes input text to identify visual elements suitable for image depiction. For each academic abstract, we apply a structured analysis process using the following system prompt:

> *"Analyze the following text for creating visual prompts.*
> *Extract and provide:*
> *1. Key visual concepts that could be depicted in images*
> *2. Emotional tone and atmosphere*
> *3. Specific visual elements relevant to AI safety themes*
> *4. Suggested visual style and composition*
> *5. Important objects, settings, or scenarios"*

This structured extraction ensures consistent processing across all abstracts. The output includes key concepts (e.g., "alignment failure", "AI control"), visual elements (e.g., "surveillance monitors", "warning indicators"), and atmospheric descriptions (e.g., "ominous", "threatening").

Prompt Generation. Based on the extracted concepts, the model creates detailed image descriptions. We apply a generation-focused prompt to synthesize the visual elements into coherent image prompts:

> *"Create a detailed image generation prompt based on the extracted concepts.*
> *Requirements:*
> *1. Create a vivid, detailed visual description*
> *2. Include specific visual elements, lighting, composition*
> *3. Ensure the prompt generates diverse, high-quality images*
> *4. Keep it under 200 words"*

Following prior work on text-to-image synthesis (Rombach et al., 2022), we incorporate threat-related modifiers including "ominous", "dystopian", and "menacing". To ensure visual diversity, we combine 12 visual styles (e.g., photorealistic, digital art, cinematic) with 15 scene environments (e.g., laboratory, research facility, surveillance center). Appendix A provides the complete lists. The complete prompt guides image generation toward threat-related visual content.

3.1.3 Image Generation

We generate images using Doubao text-to-image API with the model "doubao-seedream-3-0-t2i-250415". Each prompt produces one corresponding image at 1024×1024 resolution. In total, we generate 700 images depicting threat-related scenarios with AI risk themes.

The visual content carries implicit semantics of vigilance and caution without explicit labels. Dark atmospheres suggest danger and require vigilance, while warning displays and surveillance elements evoke caution. These visual cues embed such meanings into the images through composition rather than text annotations.

3.2 Neutral VQA Construction

We construct VQA pairs for each generated image. Following recent work on synthetic VQA generation using LLMs (Chen et al., 2024), we design questions that remain strictly neutral throughout the dataset. More concretely, these questions avoid safety-related terminology such as "threat", "danger", or "risk", and contain no leading content that might bias model responses.

We design four categories of neutral questions, with each image receiving 6 VQA pairs for a total of 4,200 samples. Descriptive questions ask about image content directly, such as "What does this image show?" and "What elements are present?". Open-ended questions invite general elaboration on scenes. Analytical questions probe scene characteristics and interpretation. Contextual questions explore settings and typical functions. The complete set of 16 question templates across these four categories is shown below.

> **Descriptive Questions:**
> *"What does this image show?"*
> *"Please describe the scene in the image."*
> *"What content can you see?"*
> *"What elements are present in the image?"*
> **Open-ended Questions:**
> *"Can you tell me more about this scene?"*
> *"What does this remind you of?"*
> *"What other details can you observe?"*
> *"What are your thoughts on this scene?"*
> **Analytical Questions:**
> *"What are the characteristics of this scene?"*
> *"How do you understand this situation?"*
> *"What impression does this environment give you?"*
> *"What do you think is happening here?"*
> **Contextual Questions:**
> *"In what situations might this scene occur?"*
> *"What is this environment typically used for?"*
> *"What do you think the function of this place is?"*
> *"What activities is this setting suitable for?"*

Answer Generation. We leverage GPT-4o-mini as the strong teacher for answer generation. The model describes image content factually and objectively, focusing on visual elements without making safety judgments. This design maintains neutrality while capturing the semantic content of each image.

Quality Control. We filter generated QA pairs using an evaluation prompt that assesses neutrality, clarity, and consistency on a 0–10 scale. See

Appendix B for detailed evaluation criteria. The prompt specifically checks for leading or biased questions that might trigger model skepticism. Samples with overall quality scores below 6.0 are discarded, ensuring that training data maintains neutral framing throughout.

3.3 Training Procedure

We perform visual instruction tuning on VLMs using the constructed dataset. Models learn to generate answers conditioned on images and questions. Following the training protocol of LLaVA, we keep the visual encoder frozen and only update the language model component.

Training Configuration. We apply LoRA for parameter-efficient fine-tuning (Hu et al., 2022), which injects trainable rank decomposition matrices while keeping pretrained weights frozen. The adapter rank is set to 128. Training uses the AdamW optimizer with a learning rate of 2e-5. We train for 5 epochs with a batch size of 16. All experiments are conducted on a single NVIDIA L20 GPU (48GB) with CUDA 12.1 and PyTorch 2.1.0. Training takes 3–4 hours for Qwen-series models and 5–6 hours for LLaVA-series models.

4 Experment

4.1 Experimental Setup

Models. We evaluate VSFA on four representative vision-language models from two model families. From the Qwen series, we use Qwen2.5-VL-7B-Instruct (Wang et al., 2024a) and Qwen3-VL-8B-Instruct (Qwen, 2025). From the LLaVA series, we test LLaVA-1.5-7B (Liu et al., 2023b) and LLaVA-v1.6-Mistral-7B (Liu et al., 2024a). These models cover different architectures and LLM backbones. We further extend VSFA to Gemma 3 IT (4B) and Llama 3.2 Vision (11B) in Appendix G, confirming cross-family generalization.

Benchmarks. For safety evaluation, we use three jailbreak attack benchmarks: FigStep, MMSafetyBench, and SPA-VL (Zhang et al., 2025b). FigStep uses typography-based attacks that embed harmful text within images. MMSafetyBench tests query-relevant image attacks across 13 scenarios. SPA-VL evaluates structure-based jailbreak attacks. For over-refusal evaluation, we use MM-Vet (Yu et al., 2024) to measure six core multimodal capabilities.

Baselines. We compare VSFA against two defense approaches. AdaShield (Wang et al., 2024b) is a prompting-based method that prepends adaptive defense prompts to inputs. It requires no fine-tuning but produces rigid refusals (Zhang et al., 2025c). VLGuard (Zong et al., 2024) is a fine-tuning-based method that trains on curated safety datasets with explicit safe/unsafe labels. We also include the original instruction-tuned model as the No Defense baseline.

Evaluation Metrics. We evaluate defense methods from three aspects. Attack Success Rate (ASR) measures safety by checking whether model responses comply with harmful intent (Jia et al., 2025). Constructive Score (CS) measures response quality across five dimensions: politeness, helpfulness, task completion, logical flow, and information richness (Duan et al., 2025). For over-refusal, we measure multimodal capabilities using MM-Vet (Yu et al., 2024) and calculate Refusal Rate as the percentage of rejected benign queries. We use GPT-4o as the judge for all metrics.

4.2 Main Results

Table 1 and Table 2 present safety performance and over-refusal evaluation across four VLMs. We analyze the results from three perspectives: attack resistance (ASR), response quality (CS), and capability preservation.

Attack Resistance. Without defense, VLMs are easy targets for jailbreak attacks. Baseline models show high ASR on all benchmarks. LLaVA-1.5-7B reaches 68.71% average ASR. Even well-aligned Qwen3-VL-8B has a 38.77% attack success rate. These numbers show real risks in deploying VLMs without safety measures. VSFA reduces average ASR to 14.18%-23.76%. This improvement comes from internalized threat awareness, not explicit safety rules.

How does VSFA compare with other methods? AdaShield achieves the lowest ASR on typography attacks like FigStep. This is reasonable. AdaShield explicitly instructs models to inspect image content for harmful elements. This counters typography-based attacks effectively, since FigStep embeds harmful text directly in images. But AdaShield performs poorly on semantic attacks like SPA-VL. Semantic attacks hide harmful intent in image-question relationships, not as explicit content in images. VLGuard performs better on weaker models. On LLaVA-1.5-7B, VLGuard achieves 8.80% ASR on FigStep, lower than AdaShield's 12.40%.

Model	Method	FigStep		MM-SafetyBench		SPA-VL		Avg.	
		ASR↓	CS↑	ASR↓	CS↑	ASR↓	CS↑	ASR↓	CS↑
Qwen3-VL-8B	No Defense	32.40	0.12	38.63	0.09	45.28	0.11	38.77	0.11
	AdaShield†	**2.40**	0.04	**8.57**	0.03	32.45	0.05	14.47	0.04
	VLGuard‡	3.80	0.32	10.83	0.29	28.49	0.31	14.37	0.31
	VSFA (Ours)	5.60	**0.51**	14.29	**0.48**	**22.64**	**0.52**	**14.18**	**0.50**
Qwen2.5-VL-7B	No Defense	35.60	0.14	42.26	0.11	48.49	0.13	42.12	0.13
	AdaShield†	**3.20**	0.05	**10.48**	0.04	35.28	0.06	16.32	0.05
	VLGuard‡	4.80	0.34	12.56	0.31	30.57	0.33	15.98	0.33
	VSFA (Ours)	6.80	**0.53**	16.31	**0.49**	**24.53**	**0.54**	**15.88**	**0.52**
LLaVA-v1.6-7B	No Defense	42.60	0.10	48.63	0.08	54.34	0.09	48.52	0.09
	AdaShield†	**5.60**	0.03	**14.29**	0.02	38.49	0.04	19.46	0.03
	VLGuard‡	6.40	0.28	16.85	0.26	34.53	0.29	19.26	0.28
	VSFA (Ours)	8.80	**0.47**	18.57	**0.45**	**28.68**	**0.49**	**18.68**	**0.47**
LLaVA-1.5-7B	No Defense	78.40	0.08	62.26	0.06	65.47	0.07	68.71	0.07
	AdaShield†	12.40	0.02	22.56	0.02	45.28	0.03	26.75	0.02
	VLGuard‡	**8.80**	0.25	**20.48**	0.23	42.64	0.26	23.97	0.25
	VSFA (Ours)	14.20	**0.45**	24.64	**0.42**	**32.45**	**0.46**	**23.76**	**0.44**

†Inference-time defense via safety prompting. ‡Fine-tuned on manually labeled harmful image-text pairs.

Table 1: Safety performance comparison across different defense methods. We report Attack Success Rate (ASR↓) and Constructive Score (CS↑) for each benchmark. CS measures the balance between safety compliance and user-centric helpfulness (Duan et al., 2025). Best in **bold**, second best underlined.

Weaker models benefit more from explicit safety supervision in VLGuard's 2K labeled samples. VSFA does not achieve the lowest ASR on every benchmark. Our ASR is slightly higher than AdaShield on FigStep. But raw ASR only tells part of the story. What happens when we look at how the model refuses?

Response Quality. CS reveals problems that ASR alone cannot capture. AdaShield achieves low ASR, but its CS is very low. This is because AdaShield's design requires models to respond with "I am sorry" when harmful content is detected. This produces uniform rigid refusals for any risky query. CS evaluates response quality through five dimensions: politeness, willingness to help, task completion, logical coherence, and information richness. A simple "I am sorry" scores near zero on all five. This is a problem for real deployment. When a user asks about drug interactions, even a helpful question might trigger AdaShield's text detection. The user receives "I am sorry" with no explanation.

VSFA produces different responses. When refusing harmful requests, the VSFA-trained model explains risks and suggests safer alternatives. For example, when asked about dangerous activities, the model might say: "This could cause harm because... If you want to learn about safety protocols, I can explain..." This approach of providing meaningful and responsible responses has better educational value. VLGuard falls in between. It is better than AdaShield but worse than VSFA. VLGuard learns from labeled data with explicit safe/unsafe labels. The model learns what to refuse, but not how to refuse constructively. VSFA achieves a good balance in both safety and helpfulness.

Over-Refusal and Capability Preservation. Safety methods often reject benign queries by mistake, and Table 2 shows this problem clearly. AdaShield shows the highest refusal rate (24-31%) and the lowest capability scores. Its static prompt asks models to check for "harmful, illegal, or dangerous" content. This rule is too strict. Many benign queries mention sensitive topics without bad intent, but AdaShield flags them anyway. VLGuard takes a different approach. It achieves the best capability scores, especially in Knowledge and Generation. But its refusal rate stays at 8-13%. The 2K labeled training samples create fixed boundaries. Queries that look like "unsafe" training examples get rejected, even when they are harmless. VSFA works differently. It achieves the lowest refusal rate while keeping strong capability scores. The key is that VSFA training has no refusal labels.

Model	Defense	Multimodal Capabilities ↑							Refusal Rate ↓
		Rec	OCR	Know	Gen	Spat	Math	Total	
Qwen3-VL-8B	AdaShield	46.2	60.5	28.8	33.5	57.9	14.2	44.8	24.82
	VLGuard	**54.5**	**63.2**	**40.8**	**42.6**	57.5	**28.5**	**51.7**	8.95
	VSFA (Ours)	53.2	62.8	39.5	41.2	**59.5**	27.2	51.0	**2.62**
Qwen2.5-VL-7B	AdaShield	43.7	58.6	25.6	31.1	58.1	12.5	42.7	26.15
	VLGuard	**50.2**	60.8	**37.1**	**39.5**	56.3	**25.8**	**48.6**	10.82
	VSFA (Ours)	49.5	**61.2**	36.5	38.8	**58.5**	24.5	48.5	**3.45**
LLaVA-v1.6-7B	AdaShield	33.2	19.5	15.5	**21.8**	27.5	9.8	24.2	30.85
	VLGuard	39.5	**30.2**	**20.2**	21.5	30.8	17.2	**30.0**	13.28
	VSFA (Ours)	**40.2**	29.5	18.8	20.8	**31.5**	**18.5**	29.9	**2.35**
LLaVA-1.5-7B	AdaShield	29.9	13.5	12.9	**19.8**	22.7	8.2	20.5	29.36
	VLGuard	34.2	**19.8**	**15.5**	18.5	24.3	12.8	23.9	12.65
	VSFA (Ours)	**35.5**	19.2	15.2	18.2	**24.8**	**14.2**	**24.3**	**1.82**

Table 2: Over-refusal evaluation on MM-Vet. We report multimodal capabilities (↑) and refusal rate on benign queries (↓).

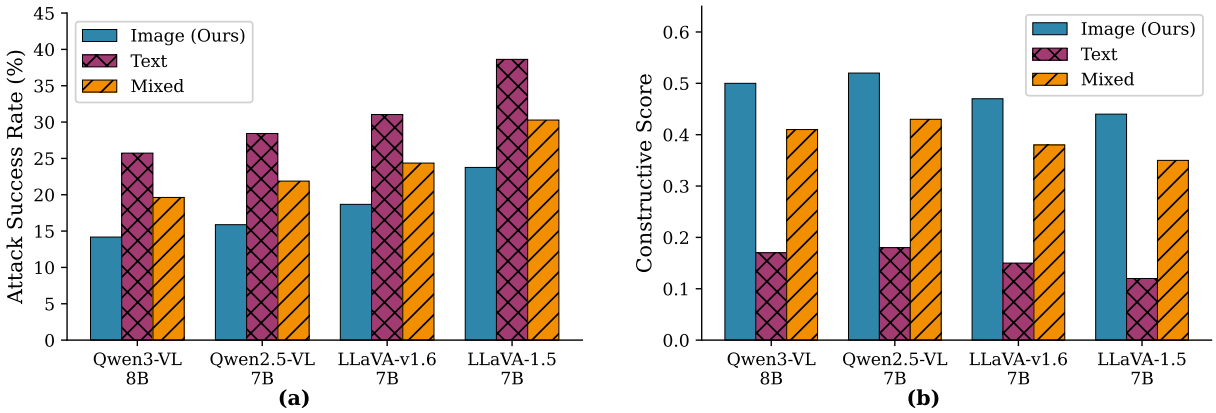

Figure 2: Ablation study on fine-tuning modality. We compare three variants: **Image** (threat-related images with neutral VQA), **Text** (text-only safety data), and **Mixed** (combination of both). (a) Attack Success Rate across four models. (b) Constructive Score across four models.

The model learns to answer neutral questions about threat-related images. It develops threat awareness from what it sees, not from being told what to refuse. This leads to an interesting result: VSFA performs better in Spatial and Recognition tasks. Seeing threat-related images seems to sharpen visual understanding without making the model too cautious. VLGuard scores slightly higher on overall capabilities, but VSFA's refusal rate is 3-4 times lower. This gives VSFA a better balance between safety and helpfulness. Additional evaluation on MMLU and MMMU confirms that VSFA preserves general reasoning capabilities with less than 0.5% drop on both benchmarks (Appendix F).

4.3 Ablation Studies

Figure 2 presents the ablation on the fine-tuning modality. We compare three variants: Image, Text, and Mixed. Image achieves the best results across all models. This is because text training only modifies the LLM backbone. It leaves the visual pathway unchanged. When attacks arrive through images, text-learned safety stays inactive. Safety alignment in one modality does not transfer to another. Mixed training has a different problem. It adds explicit "refuse" signals that compete with implicit vigilance from visual exposure. This dilutes the self-fulfilling effect. Image works through a different mechanism. Threat-related content activates persona features tied to vigilance. The model internalizes caution through what it sees, not what it is told. Image beats Mixed by 4-7% ASR across all models. Even Text achieves higher CS than AdaShield. Fine-tuning produces more constructive responses than prompting-based rigid refusals. We also conducted a visual style ablation across 12 styles, showing that the safety effect is style-invariant (Appendix E).

4.4 Mechanistic Analysis

Does VSFA change only surface behavior, or does it reshape internal representations? We use SAEs to look inside the model (He et al., 2024). We compare activations before and after VSFA training on safety evaluation prompts. We find one latent that activates more strongly after VSFA. We call it the safety-oriented persona latent. Steering experiments confirm its causal role. Adding this latent to the original model reduces ASR. Removing it from the VSFA model increases ASR. This bidirectional effect proves that VSFA works by internalizing safety-oriented persona features. Details appear in Appendix C.

5 Conclusion

We introduce VSFA, a method that extends the self-fulfilling mechanism from text to visual modalities. VSFA trains VLMs on neutral VQA tasks built around threat-related images. The training data contains no safety labels or contrastive pairs. Through repeated exposure to such visual content, models internalize vigilance and caution, shaping safety-oriented personas. Experiments across multiple VLMs and safety benchmarks show that VSFA reduces ASR while producing constructive refusal responses. The method preserves general capabilities with minimal over-refusal. Compared to prompting-based defenses, VSFA avoids rigid rejections. Compared to fine-tuning on labeled safety data, VSFA requires no manual annotation. Our findings suggest that a self-fulfilling mechanism operates effectively in the visual modality, offering a label-free approach to VLMs alignment.

Limitations

This work assumes that visual exposure alone can shape model behavior. Our mechanistic analysis uses an SAE to identify safety-oriented persona features. We find a latent that activates on safety-related contexts and verify its causal role through steering. However, SAE-based interpretability has known limitations. Feature isolation may be incomplete. The training images are all synthetic. We generate 700 images from text-to-image models based on AI safety research abstracts from arXiv, processed through prompt engineering. These images depict stylized threat scenarios rather than real photographs of weapons or dangerous situations. The visual style also reflects specific cultural conventions. We test on four VLMs at the 7B-8B scale from two model families. We do not evaluate larger models. The three safety benchmarks focus on jailbreak attacks. Other safety dimensions, such as bias and misinformation, are not examined. All evaluation uses GPT-4o as the judge, which introduces potential bias from the judge model itself.

Acknowledgments

Lijie Hu is supported by the funding BF0100 from Mohamed bin Zayed University of Artificial Intelligence (MBZUAI). Di Wang and Shu Yang are supported in part by the funding BAS/1/1689-01-01,RGC/3/7125-01-01, FCC/1/5940-20-05, FCC/1/5940-06-02, and King Abdullah University of Science and Technology (KAUST) – Center of Excellence for Generative AI, under award number 5940 and a gift from Google.

References

Andy Arditi and Runjin Chen. 2025. Finding "misaligned persona" features in open-weight models. LessWrong. Published September 9, 2025.

Luke Bailey, Euan Ong, Stuart Russell, and Scott Emmons. 2024. Image hijacks: Adversarial images can control generative models at runtime. pages 2443–2455.

Jan Betley, Daniel Chee Hian Tan, Niels Warncke, Anna Sztyber-Betley, Xuchan Bao, Martín Soto, Nathan Labenz, and Owain Evans. 2025. Emergent misalignment: Narrow finetuning can produce broadly misaligned llms.

Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. 2023. Language models can explain neurons in language models. https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, and 6 others. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*.

Rajat Chawla, Arkajit Datta, Tushar Verma, Adarsh Jha, Anmol Gautam, Ayush Vatsal, Sukrit Chaterjee, Mukunda NS, and Ishaan Bhola. 2024. Veagle: Advancements in multimodal representation learning. *CoRR*, abs/2403.08773.

Guiming Hardy Chen, Shunian Chen, Ruifei Zhang, Junying Chen, Xiangbo Wu, Zhiyi Zhang, Zhihong Chen, Jianquan Li, Xiang Wan, and Benyou Wang. 2024. Allava: Harnessing gpt4v-synthesized data for lite vision-language models. *arXiv preprint arXiv:2402.11684*.

Xin Chen, Junchao Wu, Shu Yang, Runzhe Zhan, Zeyu Wu, Min Yang, Shujian Huang, Lidia S Chao, and Derek F Wong. 2026. Neuron-aware data selection in instruction tuning for large language models. *arXiv preprint arXiv:2603.13201*.

Yiwei Chen, Yuguang Yao, Yihua Zhang, Bingquan Shen, Gaowen Liu, and Sijia Liu. 2025. Safety mirage: How spurious correlations undermine vlm safety fine-tuning and can be mitigated by machine unlearning.

Ruoxi Cheng, Yizhong Ding, Shuirong Cao, Ranjie Duan, Xiaoshuang Jia, Shaowei Yuan, Simeng Qin, Zhiqiang Wang, and Xiaojun Jia. 2025. Pbi-attack: Prior-guided bimodal interactive black-box jailbreak attack for toxicity maximization. pages 609–628.

Alex Cloud, Minh Le, James Chua, Jan Betley, Anna Sztyber-Betley, Jacob Hilton, Samuel Marks, and Owain Evans. 2025. Subliminal learning: Language models transmit behavioral traits via hidden signals in data. *CoRR*, abs/2507.14805.

Wenshuo Dong, Qingsong Yang, Shu Yang, Lijie Hu, Meng Ding, Wanyu Lin, Tianhang Zheng, and Di Wang. 2025. Understanding and mitigating cross-lingual privacy leakage via language-specific and universal privacy neurons. *CoRR*, abs/2506.00759.

Ranjie Duan, Jiexi Liu, Xiaojun Jia, Shiji Zhao, Ruoxi Cheng, Fengxiang Wang, Cheng Wei, Yong Xie, Chang Liu, Defeng Li, Yinpeng Dong, Yichi Zhang, Yuefeng Chen, Chongwen Wang, Xingjun Ma, Xingxing Wei, Yang Liu, Hang Su, Jun Zhu, and 11 others. 2025. Oyster-i: Beyond refusal - constructive safety alignment for responsible language models. *CoRR*, abs/2509.01909.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2025. Scaling and evaluating sparse autoencoders.

Asma Ghandeharioun, Ann Yuan, Marius Guerard, Emily Reif, Michael A. Lepori, and Lucas Dixon. 2024. Who's asking? user personas and the mechanics of latent misalignment.

Jeremiah Giordani. 2025. Re-emergent misalignment: How narrow fine-tuning erodes safety alignment in llms. *CoRR*, abs/2507.03662.

Yichen Gong, Delong Ran, Jinyuan Liu, Conglei Wang, Tianshuo Cong, Anyu Wang, Sisi Duan, and Xiaoyun Wang. 2025. Figstep: Jailbreaking large vision-language models via typographic visual prompts. In *Thirty-Ninth AAAI Conference on Artificial Intelligence, Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence, Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, pages 23951–23959. AAAI Press.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. 2024. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, volume 2024, pages 32694–32717.

Zhengfu He, Wentao Shu, Xuyang Ge, Lingjie Chen, Junxuan Wang, Yunhua Zhou, Frances Liu, Qipeng Guo, Xuanjing Huang, Zuxuan Wu, Yu-Gang Jiang, and Xipeng Qiu. 2024. Llama scope: Extracting millions of features from llama-3.1-8b with sparse autoencoders. *CoRR*, arXiv:2410.20526.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Lei Hsiung, Tianyu Pang, Yung-Chen Tang, Linyue Song, Tsung-Yi Ho, Pin-Yu Chen, and Yaoqing Yang. 2025. Why LLM safety guardrails collapse after fine-tuning: A similarity analysis between alignment and fine-tuning datasets. *CoRR*, abs/2506.05346.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Jingyu Hu, Shu Yang, Xilin Gong, Hongming Wang, Weiru Liu, and Di Wang. 2025a. MONICA: real-time monitoring and calibration of chain-of-thought sycophancy in large reasoning models. *CoRR*, abs/2511.06419.

Lijie Hu, Liang Liu, Shu Yang, Xin Chen, Hongru Xiao, Mengdi Li, Pan Zhou, Muhammad Asif Ali, and Di Wang. 2024. A hopfieldian view-based interpretation for chain-of-thought reasoning. *CoRR*, abs/2406.12255.

Xuhao Hu, Dongrui Liu, Hao Li, Xuanjing Huang, and Jing Shao. 2025b. Vlsbench: Unveiling visual leakage in multimodal safety. pages 8285–8316.

Wenxuan Huang, Zijie Zhai, Yunhang Shen, Shaosheng Cao, Fei Zhao, Xiangfeng Xu, Zheyu Ye, and Shaohui Lin. 2025. Dynamic-llava: Efficient multimodal large language models via dynamic vision-language context sparsification.

Xiaojun Jia, Jie Liao, Qi Guo, Teng Ma, Simeng Qin, Ranjie Duan, Tianlin Li, Yihao Huang, Zhitao Zeng, Dongxian Wu, Yiming Li, Wenqi Ren, Xiaochun Cao,

and Yang Liu. 2025. Omnisafebench-mm: A unified benchmark and toolbox for multimodal jailbreak attack-defense evaluation. *CoRR*, abs/2512.06589.

Xinke Jiang, Yue Fang*, Rihong Qiu*, Haoyu Zhang, Yongxin Xu, Hao Chen, Wentao Zhang, Ruizhe Zhang, Yuchen Fang, Xu Chu, and 1 others. 2025a. Tc-rag: Turing-complete rag's case study on medical llm systems. *ACL 2025*.

Xinke Jiang, Ruizhe Zhang*, Yongxin Xu*, Rihong Qiu*, Yue Fang, Zhiyuan Wang, Jinyi Tang, Hongxin Ding, Xu Chu, Junfeng Zhao, and 1 others. 2025b. Hykge: A hypothesis knowledge graph enhanced framework for accurate and reliable medical llms responses. *ACL 2025*.

Xinyan Jiang, Wenjing Yu, Di Wang, and Lijie Hu. 2026. Global evolutionary steering: Refining activation steering control via cross-layer consistency. *arXiv preprint arXiv:2603.12298*.

Xinyan Jiang, Lin Zhang, Jiayi Zhang, Qingsong Yang, Guimin Hu, Di Wang, and Lijie Hu. 2025c. MSRS: adaptive multi-subspace representation steering for attribute alignment in large language models. *CoRR*, abs/2508.10599.

Yilei Jiang, Yingshui Tan, and Xiangyu Yue. 2024. Rapguard: Safeguarding multimodal large language models via rationale-aware defensive prompting. *CoRR*, abs/2412.18826.

Bruce W. Lee, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Erik Miehling, Pierre L. Dognin, Manish Nagireddy, and Amit Dhurandhar. 2025. Programming refusal with conditional activation steering.

Hongji Li, Junchi Yao, Manjiang Yu, Priyanka Singh, Xue Li, Di Wang, and Lijie Hu. 2025a. Towards reasoning-preserving unlearning in multimodal large language models. *CoRR*, abs/2512.17911.

Shen Li, Liuyi Yao, Lan Zhang, and Yaliang Li. 2025b. Safety layers in aligned large language models: The key to LLM security.

Yifan Li, Hangyu Guo, Kun Zhou, Wayne Xin Zhao, and Ji-Rong Wen. 2024. Images are achilles' heel of alignment: Exploiting visual vulnerabilities for jailbreaking multimodal large language models. pages 174–189.

Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y. Zou. 2022. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024a. Llava-next: Improved reasoning, ocr, and world knowledge.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning.

Qin Liu, Fei Wang, Chaowei Xiao, and Muhao Chen. 2025. Vlm-guard: Safeguarding vision-language models via fulfilling safety alignment gap. *CoRR*, abs/2502.10486.

Xin Liu, Yichen Zhu, Jindong Gu, Yunshi Lan, Chao Yang, and Yu Qiao. 2024b. Mm-safetybench: A benchmark for safety evaluation of multimodal large language models. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LVI*, Lecture Notes in Computer Science, pages 386–403. Springer.

Xin Liu, Yichen Zhu, Yunshi Lan, Chao Yang, and Yu Qiao. 2024c. Safety of multimodal large language models on images and text. pages 8151–8159. Survey Track.

Robert K. Merton. 1948. The self-fulfilling prophecy. *The Antioch Review*, 8(2):193–210.

nostalgebraist. 2020. interpreting GPT: the logit lens. LessWrong.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Mateusz Pach, Shyamgopal Karthik, Quentin Bouniot, Serge J. Belongie, and Zeynep Akata. 2025. Sparse autoencoders learn monosemantic features in vision-language models. *CoRR*, abs/2504.02821.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with GPT-4. *CoRR*, abs/2304.03277.

Xiangyu Qi, Kaixuan Huang, Ashwinee Panda, Peter Henderson, Mengdi Wang, and Prateek Mittal. 2024. Visual adversarial examples jailbreak aligned large language models. pages 21527–21536.

Qwen. 2025. Qwen3-vl technical report. *CoRR*, abs/2511.21631.

Kaixuan Ren, Preslav Nakov, and Usman Naseem. 2025. Dual-bench: Measuring over-refusal and robustness in vision-language models. *CoRR*, abs/2510.10846.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. pages 10674–10685.

Peiyuan Tang, Haojie Xin, Xiaodong Zhang, Jun Sun, Qin Xia, and Zijiang Yang. 2025. The safety reminder: A soft prompt to reactivate delayed safety awareness in vision-language models. *CoRR*, abs/2506.15734.

Gemma Team. 2025. Gemma 3 technical report. *CoRR*, arXiv:2503.19786.

Llama Team. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Alex Turner. 2025. Self-fulfilling misalignment: Data might be poisoning our ai models. Blog post.

Keyu Wang, Jin Li, Shu Yang, Zhuoran Zhang, and Di Wang. 2026. When truth is overridden: Uncovering the internal origins of sycophancy in large language models. pages 33566–33574.

Miles Wang, Tom Dupré la Tour, Olivia Watkins, Alex Makelov, Ryan A. Chi, Samuel Miserendino, Johannes Heidecke, Tejal Patwardhan, and Dan Mossing. 2025. Persona features control emergent misalignment. *CoRR*, abs/2506.19823.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024a. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *CoRR*, abs/2409.12191.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 13484–13508. Association for Computational Linguistics.

Yu Wang, Xiaogeng Liu, Yu Li, Muhao Chen, and Chaowei Xiao. 2024b. Adashield : Safeguarding multimodal large language models from structure-based attack via adaptive shield prompting. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XX*, Lecture Notes in Computer Science, pages 77–94. Springer.

Siqi Wen, Shu Yang, Shaopeng Fu, Jingfeng Zhang, Lijie Hu, and Di Wang. 2026. Concept-based dictionary learning for inference-time safety in vision language action models. *arXiv preprint arXiv:2602.01834*.

Haodong Xie, Rahul Singh Maharjan, Federico Tavella, and Angelo Cangelosi. 2024. From concrete to abstract: A multimodal generative approach to abstract concept learning. volume abs/2410.02365.

Haochuan Xu, Yun Sing Koh, Shuhuai Huang, Zirun Zhou, Di Wang, Jun Sakuma, and Jingfeng Zhang. 2025. Model-agnostic adversarial attack and defense for vision-language-action models. *arXiv preprint arXiv:2510.13237*.

Shu Yang, Jiayuan Su, Han Jiang, Mengdi Li, Keyuan Cheng, Muhammad Asif Ali, Lijie Hu, and Di Wang. 2024a. Dialectical alignment: Resolving the tension of 3h and security threats of llms. *CoRR*, abs/2404.00486.

Shu Yang, Shenzhe Zhu, Liang Liu, Lijie Hu, Mengdi Li, and Di Wang. 2024b. Exploring the personality traits of llms through latent features steering. *arXiv preprint arXiv:2410.10863*.

Tiancheng Yang, Lin Zhang, Jiaye Lin, Guimin Hu, Di Wang, and Lijie Hu. 2025. D-LEAF: localizing and correcting hallucinations in multimodal llms via layer-to-head attention diagnostics. *CoRR*, abs/2509.07864.

Junchi Yao, Shu Yang, Jianhua Xu, Lijie Hu, Mengdi Li, and Di Wang. 2025. Understanding the repeat curse in large language models from a feature perspective. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 7787–7815.

Manjiang Yu, Hongji Li, Priyanka Singh, Xue Li, Di Wang, and Lijie Hu. 2025. PIXEL: adaptive steering via position-wise injection with exact estimated levels under subspace calibration. *CoRR*, abs/2510.10205.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. 2024. Mm-vet: Evaluating large multimodal models for integrated capabilities. pages 57730–57754.

Xiang Yue, Yuansheng Ni, Tianyu Zheng, Kai Zhang, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, and 3 others. 2024. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 9556–9567. IEEE.

Jiahao Zhang, Zeqing Zhang, Di Wang, and Lijie Hu. 2026a. Controlling repetition in protein language models. *arXiv preprint arXiv:2602.00782*.

Qi Zhang, Yifei Wang, Jingyi Cui, Xiang Pan, Qi Lei, Stefanie Jegelka, and Yisen Wang. 2025a. Beyond interpretability: The gains of feature monosemanticity on model robustness.

Ruizhe Zhang, Xinke Jiang, Zhibang Yang, Zhixin Zhang, Jiaran Gao, Yuzhen Xiao, Hongbin Lai, Xu Chu, Junfeng Zhao, and Yasha Wang. 2026b. Stackplanner: A centralized hierarchical multi-agent system with task-experience memory management. *ACL 2026*.

Yongting Zhang, Lu Chen, Guodong Zheng, Yifeng Gao, Rui Zheng, Jinlan Fu, Zhenfei Yin, Senjie Jin, Yu Qiao, Xuanjing Huang, Feng Zhao, Tao Gui,

and Jing Shao. 2025b. SPA-VL: A comprehensive safety preference alignment dataset for vision language models. pages 19867–19878.

Yuyou Zhang, Miao Li, William Han, Yihang Yao, Zhepeng Cen, and Ding Zhao. 2025c. Safety is not only about refusal: Reasoning-enhanced fine-tuning for interpretable LLM safety. pages 18727–18746.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric P. Xing, Joseph E. Gonzalez, Ion Stoica, and Hao Zhang. 2024. Lmsys-chat-1m: A large-scale real-world LLM conversation dataset.

Wenrui Zhou, Shu Yang, Qingsong Yang, Zikun Guo, Lijie Hu, and Di Wang. 2025a. Flattery in motion: Benchmarking and analyzing sycophancy in video-llms. *CoRR*, abs/2506.07180.

Zirun Zhou, Zhengyang Xiao, Haochuan Xu, Jing Sun, Di Wang, and Jingfeng Zhang. 2025b. Goal-oriented backdoor attack against vision-language-action models via physical objects. *arXiv preprint arXiv:2510.09269*.

Yongshuo Zong, Ondrej Bohdal, Tingyang Yu, Yongxin Yang, and Timothy M. Hospedales. 2024. Safety fine-tuning at (almost) no cost: A baseline for vision large language models. pages 62867–62891.

Xiaohan Zou, Jian Kang, George Kesidis, and Lu Lin. 2025. Understanding and rectifying safety perception distortion in vlms. *CoRR*, abs/2502.13095.

A Dataset Statistics

This appendix provides the complete lists of arXiv search terms, visual styles, and scene environments used in VSFA dataset construction. We also report the dataset distribution statistics.

A.1 arXiv Search Terms

We collect academic text from AI safety research on arXiv. Table 3 lists all 10 search terms used in this work.

Table 3: The complete arXiv search terms used for text collection. We retrieve up to 5 papers per term from categories cs.AI, cs.LG, cs.CY, and cs.CR.

Search Terms	
AI safety alignment	AI risk existential
artificial intelligence threat	AI alignment problem
AI safety research	machine learning safety
AI control problem	AGI risk
AI alignment failure	AI safety measures

We design these search terms to cover three aspects of AI safety. The first group targets alignment research. Terms like "AI safety alignment" and "AI alignment problem" retrieve papers on value alignment. The second group focuses on risk analysis. Terms like "AI risk existential" and "AGI risk" capture research on potential harms. The third group addresses technical solutions. Terms like "machine learning safety" and "AI control problem" find papers on safety mechanisms.

The arXiv API returns papers sorted by relevance. We extract the abstract from each paper. These abstracts provide domain-specific concepts about AI risks. GPT-4o-mini then converts the abstracts into image generation prompts. This approach ensures the generated images carry threat-related semantics without explicit harmful content.

A.2 Visual Styles and Scene Environments

We use systematic combinations of visual styles and scene environments to ensure image diversity. Table 4 shows all 12 visual styles. Table 5 lists all 15 scene environments.

The image generation pipeline tracks all used combinations. This mechanism prevents repetition and ensures diversity. With 12 styles and 15 environments, we have 180 possible base combinations. Our 700 images sample from this space with additional variation in lighting and camera angles.

The visual styles range from realistic to artistic. Photorealistic style produces images that look like photographs. Digital art and concept art create more stylized visuals. Documentary and cinematic styles add specific moods to the scenes. We include both futuristic and vintage styles to cover different time settings.

Table 4: The 12 visual styles applied during image generation. We combine these styles with scene environments to maximize visual diversity across the dataset.

Visual Styles	
Photorealistic	Digital art
Concept art	Technical illustration
Documentary style	Abstract representation
Cinematic	Artistic
Professional	Casual
Futuristic	Vintage

Table 5: The 15 scene environments used for image composition. Each environment provides different visual context for threat-related content.

Scene Environments		
Office/Workplace	Laboratory	Public space
Home environment	Industrial setting	Educational
Medical	Research facility	Urban
Rural	Indoor	Outdoor
Virtual	Mixed reality	Studio

The scene environments span common locations where AI systems operate. Laboratory and research facility represent technical settings. Office and workplace show professional contexts. Public space and urban environments depict everyday locations. We also include virtual and mixed reality to represent digital spaces.

A.3 Dataset Distribution

Table 6 summarizes the VSFA dataset statistics. The dataset contains 700 images with 4,200 VQA pairs.

Table 6: Summary statistics of the VSFA dataset. We generate 6 neutral VQA pairs per image using 16 question templates across 4 categories.

Statistic	Value
Total images	700
Total VQA pairs	4,200
VQA pairs per image	6
Image resolution	1024×1024
Question categories	4
Question templates	16

Question Categories. We design four categories of neutral questions. Each category has 4 question templates. The questions avoid safety-related words like "threat" or "danger". They focus on factual description of visual content.

Descriptive questions ask what the image shows. Examples include "What does this image show?" and "What elements are present in the image?" These questions request direct observation of visual content.

Open-ended questions invite broader discussion. Examples include "What does this remind you of?" and "What other details can you observe?" These questions allow the model to elaborate freely.

Analytical questions probe scene interpretation. Examples include "What are the characteristics of this scene?" and "What do you think is happening here?" These questions require understanding of the visual context.

Contextual questions explore settings and functions. Examples include "What is this environment typically used for?" and "What activities is this setting suitable for?" These questions connect visual content to real-world usage.

Answer Generation. We use GPT-4o-mini to generate answers for each question. The model receives the image generation prompt as context. It describes the expected visual content in a neutral and factual manner. The answers do not include safety judgments or warnings.

Table 7 shows the answer length distribution. Most answers contain 50 to 120 words. This range provides enough detail for training without excessive length.

Table 7: Answer length statistics in word count. The answers maintain moderate length suitable for visual instruction tuning.

Metric	Words
Mean	85
Median	78
Min	25
Max	180
Std	32

The answer length varies by question type. Descriptive questions tend to produce shorter answers. They focus on listing visible elements. Open-ended and analytical questions produce longer answers. They require more explanation and interpretation. This variation reflects natural response patterns in VQA tasks.

B Quality Control Details

Quality control is essential for VSFA training. VSFA relies on a core assumption: models learn vigilance from visual content through implicit exposure. The training data should not contain explicit safety signals. If a question says "this image is dangerous", the model receives direct guidance. This breaks the self-fulfilling mechanism. We need QA pairs that describe threat-related images in a neutral way. The model should develop safety awareness from what it sees, not from what the text tells it.

We use GPT-4o-mini as an automated evaluator to filter generated QA pairs (Peng et al., 2023). The evaluator checks each sample on three dimensions: neutrality, clarity, and consistency. Table 8 summarizes the evaluation guidelines for each dimension.

Neutrality. Neutrality is the most important criterion for VSFA. Why does this matter so much? VSFA works through implicit learning. The model sees threat-related images and develops vigilance on its own. If questions contain words like "dangerous" or "risky", they provide explicit safety signals. These signals tell the model how to interpret the image. The model no longer learns from visual content alone.

We check for two types of problematic words. Value-laden words include "dangerous", "harmful", "risky", and "threatening". These words express judgments about the image content. Safety-related terms include "threat", "warning", "caution", and "alert". These words introduce explicit safety concepts into the training data.

A neutral question focuses on observable facts. It asks what objects appear in the image. It asks about colors, positions, or quantities. It does not ask whether something is good or bad. Here is an example. A good question: "What equipment is visible in this laboratory?" A bad question: "What dangerous chemicals can you identify?" The second question tells the model to look for danger. The first question lets the model describe what it sees.

Clarity. Clear questions produce clear answers. Ambiguous questions lead to vague or confused responses. These low-quality responses hurt training effectiveness. The model learns better from precise descriptions than from fuzzy ones.

We examine several aspects of clarity. The question should have exactly one interpretation. "What is this?" is too vague. "What type of monitoring equipment appears in this image?" is specific. The answer should directly respond to what the question asks. If the question asks about equipment, the answer should describe equipment. It should not drift to unrelated topics.

Table 8: Evaluation criteria for QA quality control. Each dimension is scored on a 0-10 scale. We discard samples with overall score below 6.0.

Dimension	Evaluation Guidelines
Neutrality	The question should not suggest any specific answer. We check for value-laden words like "dangerous", "harmful", or "threatening". The question should avoid safety-related terms like "threat", "warning", or "caution". A neutral question asks about facts in the image. It does not ask for opinions or value judgments. VSFA relies on implicit learning from visual content. Explicit safety terms would confound this learning process.
Clarity	The question should have one clear meaning. Ambiguous questions create confusion in training. The answer should directly address the question. Both should use correct grammar with clear subjects and verbs. The answer should use specific terms instead of vague words like "something" or "stuff". Clear QA pairs provide strong training signals.
Consistency	The answer should not contradict itself. If one sentence describes "a dark room", later sentences should not mention "bright sunlight". All descriptions should match what appears in the image. The model should not describe objects that are not visible. The reasoning should follow a logical order. In multi-turn dialogues, answers should stay consistent across turns.

Grammar matters for clarity. Each sentence needs a clear subject. Run-on sentences should be split into shorter ones. The answer should use concrete nouns instead of vague references. "The control panel has three screens" is better than "There is some stuff with displays". Specific language creates stronger training signals for the model.

Consistency. Consistent answers help the model build accurate representations. Contradictory information confuses the learning process. If an answer says the room is dark, then mentions bright sunlight, the model receives conflicting signals. We check for internal consistency within each answer.

Factual accuracy is part of consistency. The answer should only describe what actually appears in the image. If the image shows two monitors, the answer should not claim there are five. The model should not invent objects or details. This factual grounding ensures the model learns real visual understanding.

Logical flow also matters. Good answers move from observation to description in a clear order. They might start with the overall scene, then describe specific objects. The reasoning should make sense. In dialogues with multiple turns, the model should remember what it said before. Later answers should not contradict earlier ones.

Evaluation Process. The evaluator receives each QA pair and outputs scores in JSON format. Here is an example output:

```
{"neutrality": 8.5,
 "clarity": 7.2,
 "consistency": 9.0,
 "overall score": 8.2,
 "recommendation": "keep"}
```

The overall score combines the three dimension scores. The recommendation can be "keep", "revise", or "discard". We apply strict filtering rules. A sample passes only when two conditions are met: the overall score reaches at least 6.0, and the recommendation is either "keep" or "revise". Samples that fail either condition are removed from the training set.

This filtering process removes low-quality samples from our dataset. The remaining samples maintain neutral framing throughout. They describe threat-related images without using explicit safety language. This ensures that VSFA can work through implicit learning as designed.

C SAE Analysis Details

This appendix provides the complete details of our sparse autoencoder (SAE) analysis. We describe the SAE training procedure, model-diffing methodology, latent identification criteria, and steering experiment results.

C.1 SAE Training

We use a sparse autoencoder trained on Qwen2.5-VL-7B activations. The SAE follows the architecture from Gao et al. (Gao et al., 2025). We collect activations from the middle layer of the language model component. The visual encoder remains frozen during both VSFA training and SAE analysis.

C.2 Model-Diffing Procedure

To investigate whether VSFA shapes safety-oriented personas, we apply model-diffing with

sparse autoencoders. Given the original model M, and the resulting fine-tuned model M_D, we compare SAE latent activations between M and M_D.

We use MMSafetyBench as our evaluation prompts. For each prompt, we collect activations at the middle layer from both the original model and the VSFA-finetuned model. We pass these activations through the SAE encoder to obtain latent activations. We average across all tokens in the assistant response. We then compute the difference: VSFA model activation minus original model activation. Latents with positive differences indicate features that become more active after VSFA training. We rank latents by this difference and focus on the top 1000 latents whose activations increase most after VSFA training.

C.3 Identifying the Safety-Oriented Persona Latent

From these top 1000 latents that activate more after VSFA, we identify which ones causally control safety behavior through steering experiments. We add multiples of each latent's decoder vector to all token activations at the target layer. We measure the effect on model responses.

We select latents that satisfy two criteria. Positive steering on the original model should increase safe responses. Negative steering on the VSFA model should decrease safe responses. We find 8 latents that meet both criteria.

The eight strongest SAE latents for steering safety are:

#12 **safety-oriented persona**: vigilance and caution patterns for identifying harmful requests.
(top tokens: `warning`, `caution`, `harmful`, `refuse`, `alert`, `danger`, `unsafe`)

#47 **risk awareness**: threat recognition and risk assessment patterns.
(top tokens: `danger`, `risk`, `careful`, `avoid`, `threat`, `hazard`, `concern`)

#89 **refusal pattern**: soft refusal responses with explanations.
(top tokens: `sorry`, `cannot`, `inappropriate`, `unable`, `decline`, `regret`, `apologize`)

#156 **ethical reasoning**: value-based judgment and moral evaluation.
(top tokens: `ethical`, `moral`, `wrong`, `responsible`, `proper`, `acceptable`, `appropriate`)

#284 **harm recognition**: identifying harmful content in requests.
(top tokens: `harmful`, `dangerous`, `illegal`, `unsafe`, `risky`, `problematic`, `concerning`)

#312 **alternative suggestion**: redirecting to safer alternatives.
(top tokens: `instead`, `alternative`, `suggest`, `recommend`, `consider`, `option`, `rather`)

#458 **explanation pattern**: providing reasons for refusal.
(top tokens: `because`, `therefore`, `reason`, `explain`, `understand`, `cause`, `result`)

#521 **context discrimination**: distinguishing benign from harmful intent.
(top tokens: `context`, `situation`, `intent`, `purpose`, `depends`, `circumstance`, `specific`)

To interpret each latent, we obtain top tokens via the logit lens approach (nostalgebraist, 2020), which computes the cosine similarity between the latent's decoder vector and vocabulary embeddings. For semantic interpretation, we examine top activating examples from LMSYS-Chat-1M (Zheng et al., 2024) and use auto-interpretation with GPT-4o (Bills et al., 2023).

C.4 SAE Setup

We apply a sparse autoencoder to Qwen2.5-VL-7B activations. The SAE uses the TopK architecture (Gao et al., 2025). We collect activations from the middle layer of the language model. The visual encoder stays frozen during both VSFA training and SAE analysis.

C.5 Model-Diffing Procedure

How do we find which latents matter for safety? We compare SAE activations between the original model M and the VSFA-trained model M_D. For each prompt in MMSafetyBench, we collect middle-layer activations from both models. We pass these through the SAE encoder and average across all response tokens.

We compute the difference for each latent. Latents with positive differences become more active after VSFA. We rank them and focus on the top 1000. But activation increase alone does not prove causality. A latent might correlate with safety without controlling it. We need steering experiments to test causal control.

C.6 Identifying the Safety-Oriented Persona Latent

Which latents actually control safety? We test each candidate through steering. We add its decoder vector to all token activations and measure how responses change.

We require bidirectional effects for causal proof. Positive steering on the original model should decrease ASR. Negative steering on the VSFA model

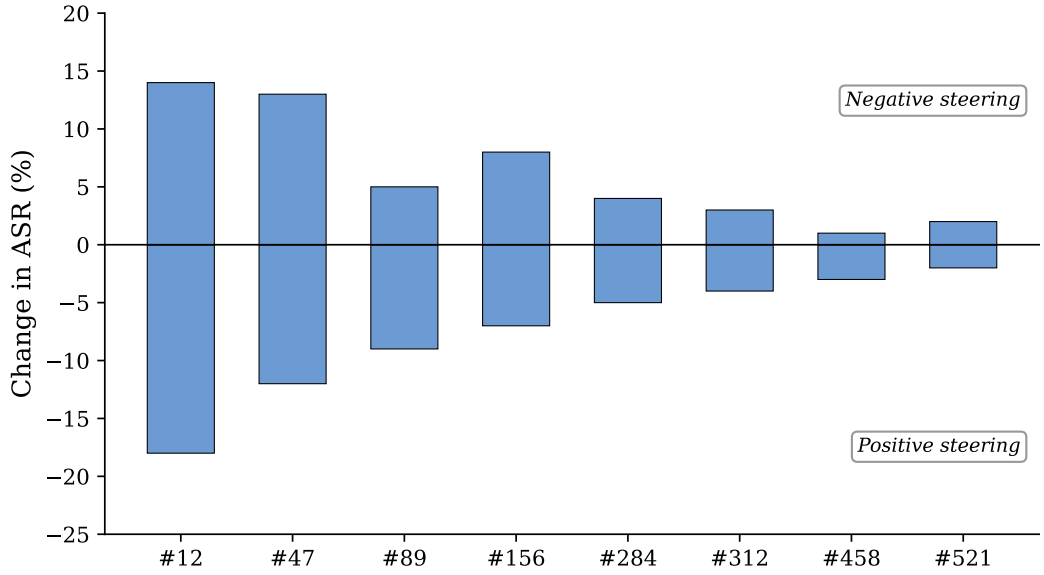

Figure 3: Bidirectional steering effects of top SAE latents. Bars below zero show ASR reduction from adding the latent to the original model. Bars above zero show ASR increase from removing it from the VSFA model. Latent #12 shows the strongest effect in both directions (−18%/+14%), confirming it as the primary safety-oriented persona latent.

should increase ASR. Why both directions? Single-direction effects might be artifacts. Bidirectional control proves genuine encoding of safety behavior.

We find 8 latents satisfying both criteria. We interpret each latent using logit lens (nostalgebraist, 2020). This computes cosine similarity between the decoder vector and vocabulary embeddings. The top tokens reveal what each latent represents.

Figure 3 visualizes the bidirectional steering effects. Bars below zero show how much ASR drops when we add each latent to the original model. Bars above zero show how much ASR rises when we remove each latent from the VSFA model. Latent #12 dominates both directions. Adding it reduces ASR by 18%. Removing it increases ASR by 14%. No other latent comes close to this bidirectional strength.

What does latent #12 encode? Its top tokens tell the story. Warning, caution, harmful, refuse, alert, danger. These are not random words. They form a coherent pattern of vigilance and threat awareness. This latent encodes exactly what we predicted in Section 1. VSFA shapes safety-oriented personas through visual exposure to threat-related content. The model does not memorize specific refusal phrases. It develops an internal representation that recognizes threats and responds with caution.

The other 7 latents support this picture. Risk awareness (#47) handles threat recognition. Refusal pattern (#89) produces polite declines. Ethical reasoning (#156) evaluates moral implications. Together they form a safety-oriented persona that VSFA training activates.

C.7 Summary

Our SAE analysis reveals three findings about how VSFA works.

VSFA activates a specific latent in the model. This safety-oriented persona latent shows higher activation after VSFA training. Its top tokens encode vigilance and caution patterns. The semantic content matches our hypothesis about self-fulfilling alignment.

Steering experiments confirm causal control. The same latent works bidirectionally on two different models. Adding it to the original model makes responses safer. Removing it from the VSFA model makes responses less safe. This rules out correlation. The latent genuinely controls safety behavior.

This provides mechanistic evidence for self-fulfilling alignment. Visual exposure to threat-related images activates safety-oriented persona features. These features guide cautious behavior across diverse contexts. The model internalizes a vigilant persona rather than learning surface patterns.

C.8 Cross-Model Evidence

Our SAE analysis focuses on Qwen2.5-VL-7B. A natural question is whether persona features exist in other model families. Our main experi-

ments already show that VSFA produces consistent ASR reductions across four models from the Qwen and LLaVA families (Table 1). This suggests a shared mechanism. Independent work by Arditi et al. (Arditi and Chen, 2025) provides direct evidence. They trained SAEs on Llama-3.1-8B-Instruct and Qwen2.5-7B-Instruct and applied model-diffing between the original and emergently misaligned versions. They found interpretable persona features in both models. These features correspond to undesirable traits like toxicity and manipulation. The features were consistent across multiple random seeds. For Llama, all 10 misalignment-relevant features appeared across 3 different fine-tuning runs. For Qwen, 9 out of 10 appeared in at least 2 runs. This confirms that persona features are not specific to one architecture. They are a general property of instruction-tuned language models. Our work shows the other side of this coin. If misaligned persona features exist across model families, safety-oriented persona features should too. VSFA activates these safety features through visual exposure, and the consistent results across our six tested models support this view.

D Teacher Model Comparison

This appendix reports our comparison of four teacher models for VSFA data construction: GPT-4o-mini (OpenAI, 2023), GPT-5, Claude 4.5 Sonnet, and Gemini-3-pro. The teacher model handles concept extraction, image prompt generation, and VQA answer generation in our pipeline. To select the best one, we ran the full VSFA pipeline with each model separately. Each model produced prompts for 60 images and about 360 QA pairs. We used Doubao as the text-to-image model for all conditions, fine-tuned Qwen2.5-VL-7B-Instruct with LoRA under identical hyperparameters, and evaluated on FigStep using GPT-4o as the judge. Table 9 shows the results.

Table 9: FigStep ASR across teacher models on Qwen2.5-VL-7B. All models use 60 images and about 360 QA pairs. Lower ASR means safer.

Teacher Model	Images	QA Pairs	Avg. Length	ASR (%) ↓
Baseline (no fine-tuning)	–	–	–	35.6
GPT-4o-mini (ours)	**60**	**360**	**659 chars**	**12.0**
Claude 4.5 Sonnet	60	360	1,322 chars	22.2
Gemini-3-pro	60	360	2,467 chars	28.8
GPT-5	60	360	769 chars	22.8

GPT-4o-mini achieves 12.0% ASR, a 66% reduction from the 35.6% baseline. The three stronger models also reduce ASR but by much less. Claude 4.5 Sonnet and GPT-5 reach about 22%, and Gemini-3-pro only drops to 28.8%. Answer length does not explain this gap. GPT-5 and GPT-4o-mini produce answers of similar length, but GPT-5's ASR is nearly twice as high. We looked at the image prompts from each model and found the real difference there. GPT-4o-mini writes short, concrete prompts. They focus on specific visual elements like dark lighting, red warning signs, and surveillance cameras. Doubao receives clear instructions from these prompts and produces images with strong, consistent threat atmosphere. The concise answers from GPT-4o-mini also train the student model to respond briefly. This helps the model produce short refusals at test time. The stronger models produce noisier prompts. GPT-5 tends toward abstract, high-level descriptions. Claude 4.5 Sonnet tries to balance threatening and reassuring elements in the same prompt. Gemini-3-pro packs too many visual details into each prompt. In all three cases, Doubao receives unfocused instructions and produces images with weaker threat signals. This pattern is consistent with observations in knowledge distillation research. Gu et al. (Gu et al., 2024) showed that the largest model is not always the best teacher, and smaller models can sometimes produce more focused training signals. Based on this comparison, we selected GPT-4o-mini as the teacher model for all experiments in the main paper. Among the four models we tested, it produces the clearest threat imagery and the most effective training data.

E Visual Style Ablation

This appendix examines whether the safety effect of VSFA depends on any particular visual style. Our main experiments use a mix of 12 visual styles when generating threat-related images. To isolate the contribution of each style, we trained Qwen2.5-VL-7B separately with 50 images of each style. We generated the corresponding VQA pairs through the same pipeline described in Section 3. We used LoRA rank 128 and only fine-tuned the language model. The visual encoder stayed frozen. We evaluated each model on FigStep with GPT-4o as the judge. Table 10 shows the results.

All 12 styles achieve ASR between 11.2% and 13.7%. The mean is 12.5% and the standard deviation is only 0.74%. Even the highest and lowest ASR only differ by 2.5 percentage points. Photorealistic and Abstract representation are the two most

Table 10: Visual style ablation on Qwen2.5-VL-7B. Each row is a separate training run using 50 images of one style. Lower ASR means safer.

Visual Style	Images	FigStep ASR (%) ↓
No Defense (baseline)	0	35.6
Photorealistic	50	12.8
Digital art	50	13.1
Concept art	50	12.4
Technical illustration	50	13.7
Documentary style	50	11.9
Abstract representation	50	12.6
Cinematic	50	11.4
Artistic	50	11.2
Professional	50	12.1
Casual	50	13.1
Futuristic	50	12.8
Vintage	50	13.3

visually different styles in our set. One looks like a photograph. The other is stylized and non-literal. Yet their ASR results are almost the same , a gap of just 0.2 percentage points. This means the safety effect does not depend on visual style. The alignment signal comes from what the image depicts, not from how it looks. Our SAE analysis in Appendix C supports this. The most activated latent encodes abstract safety concepts rather than visual style features. Our finding also has a practical benefit. Practitioners do not need to optimize for any particular visual style when building VSFA training sets. It also explains why VSFA generalizes to attack categories not seen during training. The safety persona is activated by the semantic content of threat-related scenes, regardless of how they are rendered.

F General Capability Evaluation (MMLU and MMMU)

This appendix reports VSFA's performance on two general reasoning benchmarks: MMLU (Hendrycks et al., 2021) and MMMU (Yue et al., 2024). MMLU covers 57 subjects across STEM, humanities, social sciences, and other domains. MMMU covers 30 subjects with college-level questions that require both visual and textual reasoning. We evaluated Qwen2.5-VL-7B before and after VSFA training using the standard protocol for both benchmarks. Table 11 shows the results.

Table 11: General capability evaluation on Qwen2.5-VL-7B before and after VSFA training.

Benchmark	Baseline	VSFA	Diff
MMLU	68.52%	68.13%	−0.39%
MMMU	50.56%	50.11%	−0.45%

MMLU drops by 0.39% and MMMU drops by 0.45%. Both drops are well under 0.5%, which is within the normal variance of safety fine-tuning methods. Zong et al. (Zong et al., 2024) reported MMLU changes between −0.08% and +1.79% across different VLGuard configurations. Our drops are at the lower end of this range. Combined with MM-Vet results in Table 2, VSFA preserves general capabilities across MMLU, MMMU, and MM-Vet.

G Cross-Family Generalization

This appendix tests whether VSFA generalizes beyond the 7B to 8B models used in the main paper. Our main experiments cover four models from two families: Qwen2.5-VL-7B, Qwen3-VL-8B, LLaVA-1.5-7B, and LLaVA-v1.6-Mistral-7B. To test across different scales and architectures, we applied VSFA to two additional models: Gemma 3 IT (4B) (Team, 2025) and Llama 3.2 Vision (11B) (Team, 2024). Gemma 3 IT uses a 4B language backbone with a SigLIP-based vision encoder. Llama 3.2 Vision uses an 11B backbone with a different vision-language integration architecture. Neither shares a vision encoder, projection layer, or LLM backbone with the Qwen or LLaVA families. We ran the standard VSFA pipeline on each model with only 60 training images, fine-tuned with LoRA under identical settings, and evaluated on FigStep with GPT-4o as the judge. Table 12 shows the results.

Table 12: VSFA on additional model families. Each model is trained with only 60 images. Lower ASR means safer.

Model	Size	Baseline ASR	VSFA ASR	Diff
Gemma 3 IT	4B	37.0%	7.2%	−29.8%
Llama 3.2 Vision	11B	49.8%	17.6%	−32.2%

Gemma 3 IT drops from 37.0% to 7.2% ASR, and Llama 3.2 Vision drops from 49.8% to 17.6%. Both are large reductions with only 60 training images. All six models now tested use different vision encoders and LLM backbones, covering sizes from 4B to 11B. VSFA produces consistent ASR reductions across all of them. This suggests that the mechanism is not tied to any particular architecture or model size. It is consistent with our SAE finding (Appendix C) that VSFA activates general safety-oriented persona features that appear to exist across different model families. The data efficiency is also notable. Our main experiments use 700 images, but here we used only 60 and still achieved large ASR

reductions. This confirms that VSFA does not require large-scale training data to be effective.