Locating the Refusal Circuit in Unified, Encoder-Free Vision-Language Models

Background reading, related work, planned techniques, and questions for prospective mentees

0. Why this matters now: VLMs are already load-bearing in the real world

This isn't a hypothetical concern for some future deployment. VLMs are already the perception/decision backbone in three fast-growing, safety-relevant categories of real-world systems:

- **Computer-use agents.** VLM-driven agents now interpret raw screenshots and translate them into clicks, keystrokes, and tool calls to operate real software autonomously (Liu et al.). Any visual jailbreak vulnerability here isn't abstract — a manipulated screenshot, malicious webpage, or adversarial document rendered on-screen is a direct vector for hijacking an agent that has real permissions on a real machine.
- **Autonomous driving.** Vision-Language-Action (VLA) models are increasingly used for perception, prediction, and planning in driving stacks, consuming multi-camera input and producing trajectories or low-level control (Jiang et al.). A safety failure triggered by adversarial or even naturally corrupted visual input in this setting has physical, not just informational, consequences.
- **Robotics.** Open VLA models like OpenVLA (Kim et al.) fine-tune vision-language backbones directly into visuomotor control policies for real robot manipulation. The same visual channel that's vulnerable to jailbreaking in a chat-style VLM is, in this setting, the direct input to a physical actuator.

Across all three, the deploying system inherits whatever safety and reliability properties (or failures) exist in its underlying VLM's visual channel — which is exactly the channel this project is about.

This isn't a theoretical risk either — it's already measurably severe, even without anyone trying to attack the model. Our own prior work, **VLM-RobustBench** (Saxena, Suglia, and Minervini; ICML 2026), systematically evaluated 11 VLMs across 4 major families (Qwen, InternVL, Molmo, Gemma) under 133 different corruption settings spanning noise, blur, weather, digital, and geometric perturbations at graded severities. The headline finding: **visual severity**

is a weak predictor of difficulty. Low-severity, visually mild perturbations — e.g. glass_blur at low severity — degraded MMBench accuracy by roughly 8 percentage points on average across models, in some cases more than visually severe photometric corruptions. In other words, models are *semantically strong but spatially fragile*: perturbations too mild for a human to register as meaningfully different can still meaningfully break a VLM's behavior.

If naturally occurring, low-severity, non-adversarial perturbations already produce this much instability, it strongly motivates taking *deliberately crafted* visual jailbreak inputs seriously as a security-critical problem — and motivates understanding exactly where and how the visual channel fails mechanistically, rather than only measuring that it fails. That's the gap this project addresses for the emerging encoder-free architecture class specifically.

1. Encoder-free vs. modular VLMs

Most current Vision-Language Models are **modular**: a frozen vision encoder (e.g. a CLIP-style ViT) extracts visual features, which a lightweight projection layer or cross-attention connector then maps into the LLM's hidden dimension. LLaVA, Qwen2-VL, and most academic VLM safety literature to date target this design. The projector is the seam between two components trained under very different objectives — the vision encoder never sees safety training, and the LLM's safety alignment never sees raw visual features — and that seam is exactly where mechanistic work has localized the visual jailbreak gap: the projector, trained only on benign image-text data, doesn't learn to route toxic visual concepts into the LLM's refusal direction, so harmful visual content looks "safe" to the LLM purely because of where the projector places it in activation space (Xu et al.; "The Visual Refusal Gap").

Encoder-free (unified) VLMs remove that seam entirely. Rather than a separate vision tower feeding a projector, raw image (and sometimes audio) patches are projected directly into the LLM's embedding space through simple linear layers, and the entire model is a single decoder-only transformer processing text and visual tokens together from the first layer onward (Kuo and Cascante-Bonilla). **Gemma 4 12B ("Unified")** is a concrete, current example: unlike the other Gemma 4 sizes (E2B, E4B, 31B), which retain dedicated vision/audio encoders (~150-550M params each), the 12B Unified variant eliminates these encoders entirely — image and audio patches are projected straight into the shared decoder's embedding space, and the entire model can be fine-tuned in a single pass as a result, since there's no separate pretrained component with its own frozen objective sitting in the pipeline ("Google/Gemma-4-12B-It"; Davies; Junafinity).

This is a genuinely different architectural premise for safety research. In a modular VLM, "where does the safety gap live?" has a natural first answer — check the projector's output boundary. In an encoder-free model, there is no discrete component to point to first. If the gap exists (untested, but likely given how consistently it appears across modular architectures, per Azulay et al.), the failure must be either distributed across early decoder layers, tied to how patch-position embeddings are learned relative to text-token embeddings, or emergent from training-data composition rather than localized to any single module. No published mechanistic

work has addressed this yet — the unified architecture class is only weeks old at the time of writing, and existing safety literature implicitly assumes the modular design.

2. Recent VLM safety approaches for modular VLMs

All of the following target the projector-LLM interface in modular architectures in one way or another, and represent the current state of the art. None have been tested on, or straightforwardly transfer to, encoder-free architectures — that gap is a core part of this project's motivation.

Method	**Mechanism**	**What it targets**	**Overhead**	**Known limits**
VLGuard	Supervised fine-tuning on a curated safe multimodal instruction dataset	Full VLM (or just the visual projection layer)	High — full fine-tuning run	Can cause catastrophic forgetting of core capabilities; weak against optimization-based adversarial attacks
ECSO ("Eyes Closed, Safety On")	Converts the image to a text caption via an image-to-text step, so the LLM's existing text-safety training fires normally	Input interface (bypasses the visual channel)	High — adds a captioning step	Loses information in the text conversion; misses subtle visual jailbreaks that don't reduce well to a caption
AdaShield	Adaptive prompt-level shielding on the text side	Input text template	Low — inference-time prompting only	Easily bypassed by visual-only attacks or novel formatting; doesn't touch the visual channel at all
Textual unlearning	Fine-tunes just the LLM backbone to	LLM backbone only	Medium (~1/6 the cost of full	Vulnerable to genuine multimodal attacks where

Method	Mechanism	What it targets	Overhead	Known limits
	unlearn toxic textual targets		multimodal tuning)	the text query itself is benign — a known "shortcut alignment" failure
ShiftDC	Measures and subtracts a "safety-relevant shift" from intermediate hidden states at inference time	Intermediate LLM layers	Low — training-free, inference-time calibration	Needs clean reference samples to estimate the shift; degrades on edge-case domains
CARE	Causal mediation analysis to find the exact neurons/layers responsible for unsafe behavior, then orthogonal projection onto a bimodal safety subspace	Neuron-level FFN activations + bimodal fusion point	Low — training-free, generalized eigen-decomposition	Assumes safety states are linearly separable; complex non-linear attacks can still degrade the boundary
DTR (Dynamic Token Reweighting)	Optimizes the KV cache at inference time to dynamically evict visual tokens that are inducing a safety shift	Key-Value cache, specific layers	Low-Medium — a few optimization steps per inference call	Full white-box access could let an adversary bypass the token-eviction optimization
CR-VLM	Extracts configurable refusal vectors via	Targeted activation steering	Low — lightweight predictor	Relies on hard-coded gating thresholds; can

Method	Mechanism	What it targets	Overhead	Known limits
	teacher-forced activation extraction, then steers decoding			cause over-refusal on benign inputs

A cross-cutting confounder in evaluating all of the above: many standard safety benchmarks suffer from **Visual Safety Information Leakage (VSIL)** — the text query accompanying a harmful image often leaks the sensitive intent itself (e.g. "how do I build this gun?" next to an image of a weapon), letting the model refuse based on the text alone without any real multimodal safety alignment being exercised. This makes textual unlearning or simple prompting look artificially effective. **VLSBench** was built specifically to decouple visual safety information from the text query, giving a much more honest read on true multimodal safety alignment (Hu et al.) — this project uses VLSBench as its primary evaluation standard for exactly this reason.

Primary references, by method (see full set of references as well):

- VLGuard: Zong et al.
- ECSO: Gou et al.
- AdaShield: Wang et al. — the source literature review's original citation for this had pointed to the wrong paper (DTR's supplemental material); corrected here.
- Textual unlearning: Chakraborty et al. (Trishna Chakraborty, Erfan Shayegani, Zikui Cai, Nael Abu-Ghazaleh, M. Salman Asif, Yue Dong, Amit K. Roy-Chowdhury, Chengyu Song).
- ShiftDC: Zou et al. (Xiaohan Zou, Jian Kang, George Kesidis, Lu Lin) — a different Zou et al. from the AdvBench/GCG paper below.
- CARE: Fu et al. (Jinhu Fu, Yihang Lou, Qingyi Si, Shudong Zhang, Yan Bai, Sen Su).
- DTR: Jiang et al. (Tanqiu Jiang, Jiacheng Liang, Rongyi Zhu, Jiawei Zhou, Fenglong Ma, Ting Wang).
- CR-VLM: Yang et al. (Jiaxi Yang, Shicheng Liu, Yuchen Yang, Dongwon Lee).

3. Relevant datasets and benchmarks

Safety / jailbreak evaluation:

- **FigStep** (Gong et al.), **MM-SafetyBench** (Liu et al.), **HADES** (Li et al.) — the original benchmarks establishing the visual jailbreak phenomenon; useful for baseline replication and comparison to prior published numbers, but all suffer from VSIL to varying degrees.
- **VLSBench** (Hu et al.) — the leakless standard; decouples visual safety-critical information from the accompanying text so refusal can't shortcut off the text channel. This is the primary evaluation benchmark for the project.

- **HoliSafe** (Lee et al.) — a broader multimodal safety benchmark spanning all five safe/unsafe image-text combinations, useful as a secondary check that findings aren't VLSBench-specific.
- **AdvBench** — used to build the clean reference set for difference-of-means refusal-direction extraction.
- **AlpacaEval** (Tatsu Lab) — general instruction-following prompts, useful as a benign reference distribution alongside AdvBench.

Utility / capability evaluation (to check safety interventions don't break general capability):

- **MMBench, MM-Vet, MMMU-Pro.** general multimodal capability benchmarks, used in Month 3 to measure the safety-utility tradeoff of any patch or intervention. Both AdaShield (Wang et al.) and CARE report using MM-Vet/MMBench-style utility checks alongside their safety evaluation.

Model resources:

- google/gemma-4-12B-it — one of the first unified/encoder-free target models. Recently released by Google DeepMind.
- **LLaVA-1.5, Qwen-VL, InternVL, Molmo** — modular baselines for direct architectural comparison.

4. Relevant techniques

- **Difference-of-means refusal direction extraction** (Arditi et al. 2024) — computing the mean difference between final-token activations on contrastive harmful/harmless prompt sets to define a linear refusal direction in the residual stream. This direction has been shown to be both causally necessary (projecting activations onto its orthogonal complement disables refusal) and sufficient (adding a multiple of it forces refusal on benign prompts) in text-only and modular-VLM settings — a first goal of this project is checking whether that cleanly holds in a unified decoder too.
- **Activation patching / causal tracing** — swapping activations between clean and corrupted (or harmful/harmless) runs at specific layers and positions to identify which components causally matter for a behavior (Elluru et al.; Neo et al.). Since there's no projector boundary to patch in a unified architecture, this project uses layer-wise and position-wise patching across the whole decoder stack instead of targeting a single known bottleneck.
- **Logit lens** — projecting intermediate residual-stream states through the unembedding matrix to track how refusal-token probability evolves layer-by-layer during a forward pass, useful for narrowing down where the visual-jailbreak-induced failure emerges.
- **Sparse Autoencoders (SAEs)** — training an autoencoder on cached residual-stream activations to decompose them into (hopefully) more interpretable, disentangled features, used here to probe how unified decoders represent visual vs. textual harmful concepts differently.

- **Causal mediation analysis** — a more targeted successor to activation patching, used to pin down the exact neurons/layers mediating unsafe behavior once patching has narrowed the search space (this is the technique the CARE paper, Fu et al., uses for modular VLMs; adapting the same rigor here without a projector to pre-narrow the search is a core technical challenge of the project).
- **Ablation studies (hydra-effect tracing)** — systematically ablating candidate refusal components and checking whether alternative/redundant circuits reactivate to preserve refusal behavior (Prakash et al.), to understand how robust any patch needs to be against this redundancy.

5. Potential Workplan.

Month 1: [both mentees] supported by the research team, set up the codebase to evaluate the unified model (e.g., Gemma 4, Qwen 3.5) alongside (Molmo2, Qwen-VL, InternVL, etc.). Prepare reference datasets (VLSbench and VLM-RobustBench). Identify relevant libraries to use as a starting point; for instance, NNSight is a good option: https://nnsight.net/.

Month 2:

[Mentee 1] extract a difference-of-means refusal direction, and quantify the jailbreak gap with respect to modular baselines on VLSBench.

[Mentee 2] logit-lens tracing to find where visual representations of harmful content diverge from text equivalents, and probe hydra-effect dynamics under ablation.

Month 3: [both mentees] design and evaluate a targeted intervention (necessarily different from projector-specific patches, since none exists here and potentially inspired by (Wollschläger et al. 2025), benchmark safety/utility tradeoffs on VLSBench + MMBench/MMLU against best-effort modular baselines, and write up for a technical report (e.g., https://www.lesswrong.com/).

6. Questions for prospective mentees to think through

Before applying, it's worth sitting with these — strong applications tend to engage with them directly rather than just confirming interest:

1. **Novelty and risk.** This project is explicitly betting that a brand-new architecture class raises open questions that established projector-focused methods don't answer. What would make you *doubt* that bet — i.e., what result in Month 1 would suggest the encoder-free/modular distinction turns out not to matter much for where the safety gap lives?
2. **Dual-use awareness.** Understanding exactly how and where a model's refusal mechanism fails is, definitionally, also a partial recipe for defeating it. How would you think about responsible disclosure if this project produces a working attack technique before it produces a working defense?
3. **Generalization.** If you do find and patch a safety gap in this specific 12B unified model, what's your honest estimate of how much that finding would transfer to other unified

architectures (different labs, different scales, different training recipes) versus being an artifact of this one model's specific training?

4. **Tooling maturity as a research constraint.** Given that standard interpretability tooling doesn't yet support this architecture, some of your time will go into infrastructure rather than "pure" research. How do you personally decide when building better tooling is worth the time versus when it's a distraction from the actual question?
5. **Why this matters beyond the paper.** Multimodal jailbreaks aren't just a benchmark number — they're a real deployment risk as more agentic systems process images, screenshots, and documents as part of their normal operation. If unified, encoder-free architectures become a default deployment target (as their efficiency advantages suggest they might), what's actually at stake if their safety properties turn out to be worse understood than modular models' by the time that happens?

References

"AlpacaEval Leaderboard." *AlpacaEval*, Tatsu Lab, tatsu-lab.github.io/alpaca_eval/. Accessed 20 July 2026.

Arditi, Andy, et al. "Refusal in Language Models Is Mediated by a Single Direction." *arXiv*, 17 June 2024, arxiv.org/abs/2406.11717. Authors: Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda.

Azulay, Aharon, et al. "Jailbreaking Vision-Language Models Through the Visual Modality." *arXiv*, 1 May 2026, arxiv.org/abs/2605.00583. Authors: Aharon Azulay, Jan Dubiński, Zhuoyun Li, Atharv Mittal, and Yossi Gandelsman.

Chakraborty, Trishna, et al. "Cross-Modal Safety Alignment: Is Textual Unlearning All You Need?" *arXiv*, 27 May 2024 (rev. 14 Oct. 2025), arxiv.org/abs/2406.02575. Authors: Trishna Chakraborty, Erfan Shayegani, Zikui Cai, Nael Abu-Ghazaleh, M. Salman Asif, Yue Dong, Amit K. Roy-Chowdhury, and Chengyu Song.

Chen, Yangyi, et al. "Why Should Adversarial Perturbations Be Imperceptible? Rethink the Research Paradigm in Adversarial NLP." *arXiv*, 19 Oct. 2022, arxiv.org/abs/2210.10683. Authors: Yangyi Chen, Hongcheng Gao, Ganqu Cui, Fanchao Qi, Longtao Huang, Zhiyuan Liu, and Maosong Sun.

Davies, Dave. "Training and Deploying VLM Models with Gemma 4." *Online Inference*, Medium, medium.com/online-inference/training-and-deploying-vlm-models-with-gemma-4-c5dc1179833c.

Elluru, Veeraraju, et al. "Mechanistically Interpreting Compression in Vision-Language Models." *arXiv*, 26 Mar. 2026, arxiv.org/abs/2603.25035. Authors: Veeraraju Elluru, Arth Singh, Roberto Aguero, Ajay Agarwal, Debojyoti Das, and Hreetam Paul.

Fu, Jinhu, et al. "Diagnosing and Repairing Unsafe Channels in Vision-Language Models via Causal Discovery and Dual-Modal Safety Subspace Projection" (CARE). *arXiv*, 28 Mar. 2026,

arxiv.org/abs/2603.27240. Authors: Jinhu Fu, Yihang Lou, Qingyi Si, Shudong Zhang, Yan Bai, and Sen Su.

Gong, Yichen, et al. "FigStep: Jailbreaking Large Vision-Language Models via Typographic Visual Prompts." *arXiv*, 9 Nov. 2023, arxiv.org/abs/2311.05608. Authors: Yichen Gong, Delong Ran, Jinyuan Liu, Conglei Wang, Tianshuo Cong, Anyu Wang, Sisi Duan, and Xiaoyun Wang.

"Google/Gemma-4-12B-It." *Hugging Face*, Google DeepMind, huggingface.co/google/gemma-4-12B-it. Accessed 20 July 2026.

Gou, Yunhao, et al. "Eyes Closed, Safety On: Protecting Multimodal LLMs via Image-to-Text Transformation" (ECSO). *arXiv*, 14 Mar. 2024, arxiv.org/abs/2403.09572. Authors: Yunhao Gou, Kai Chen, Zhili Liu, Lanqing Hong, Hang Xu, Zhenguo Li, Dit-Yan Yeung, James T. Kwok, and Yu Zhang.

Hu, Xuhao, et al. "VLSBench: Unveiling Visual Leakage in Multimodal Safety." *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics* (Volume 1: Long Papers), Association for Computational Linguistics, 2025, pp. 8285-8316, aclanthology.org/2025.acl-long.405/. Authors: Xuhao Hu, Dongrui Liu, Hao Li, Xuanjing Huang, and Jing Shao. Also on arXiv as 2411.19939; project page at ai45lab.github.io/VLSBench/.

Jiang, Sicong, et al. "A Survey on Vision-Language-Action Models for Autonomous Driving." *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, 2025, pp. 4524-4536. Authors include Sicong Jiang, Zilin Huang, Kangan Qian, Ziang Luo, Tianze Zhu, Yang Zhong, Yihong Tang, and additional co-authors (full list on the paper itself — a large multi-institution author list).

Jiang, Tanqiu, et al. "Dynamic Token Reweighting for Robust Vision-Language Models" (DTR). *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026, pp. 24481-24491, openaccess.thecvf.com/content/CVPR2026/papers/Jiang_Dynamic_Token_Reweighting_for_Robust_Vision-Language_Models_CVPR_2026_paper.pdf. Authors: Tanqiu Jiang, Jiacheng Liang, Rongyi Zhu, Jiawei Zhou, Fenglong Ma, and Ting Wang.

Junafinity. "Uncensoring Gemma 4 12B with Heretic — Keeping Vision, Audio, and the Encoder-Free Design Intact." *Hugging Face*, huggingface.co/blog/junafinity/gemma-4-12b-abliterated.

Kim, Moo Jin, et al. "OpenVLA: An Open-Source Vision-Language-Action Model." *arXiv*, 13 June 2024, arxiv.org/abs/2406.09246. Full author list: Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn.

Kuo, Shang-Jui Ray, and Paola Cascante-Bonilla. "Do VLMs Need Vision Transformers? Evaluating State Space Models as Vision Encoders." *arXiv*, 19 Mar. 2026, arxiv.org/abs/2603.19209.

Lee, Youngwan, et al. "HoliSafe: Holistic Safety Benchmarking and Modeling for Vision-Language Model." *arXiv*, 25 Nov. 2025 (v5), arxiv.org/abs/2506.04704. Authors: Youngwan Lee, Kangsan Kim, Kwanyong Park, Ilcahe Jung, Soojin Jang, Seanie Lee, Yong-Ju Lee, and Sung Ju Hwang.

Li, Yifan, et al. "Images Are Achilles' Heel of Alignment: Exploiting Visual Vulnerabilities for Jailbreaking Multimodal Large Language Models" (HADES). *arXiv*, 14 Mar. 2024, arxiv.org/abs/2403.09792. Authors: Yifan Li, Hangyu Guo, Kun Zhou, Wayne Xin Zhao, and Ji-Rong Wen.

Liu, Xin, et al. "MM-SafetyBench: A Benchmark for Safety Evaluation of Multimodal Large Language Models." *arXiv*, 29 Nov. 2023, arxiv.org/abs/2311.17600. Authors: Xin Liu, Yichen Zhu, Jindong Gu, Yunshi Lan, Chao Yang, and Yu Qiao.

Liu, Xunzhuo, et al. "Adaptive Vision-Language Model Routing for Computer Use Agents." *arXiv*, 13 Mar. 2026, arxiv.org/abs/2603.12823. Authors: Xunzhuo Liu, Bowei He, Xue Liu, Andy Luo, Haichen Zhang, and Huamin Chen.

Marbut, Anna C., Daniel R. Olson, and Travis J. Wheeler. "Expert-Aware Refusal Steering." arXiv preprint arXiv:2606.04160 (2026).

Neo, Clement, et al. "Towards Interpreting Visual Information Processing in Vision-Language Models." *Proceedings of the 13th International Conference on Learning Representations (ICLR 2025)*, 2025, pp. 25461-25478. Also on arXiv as 2410.07149. Authors: Clement Neo, Luke Ong, Philip Torr, Mor Geva, David Krueger, and Fazl Barez.

Prakash, Nirmalendu, et al. "Beyond I'm Sorry, I Can't: Dissecting Large-Language-Model Refusal." *arXiv*, 7 Sept. 2025, arxiv.org/abs/2509.09708. Authors: Nirmalendu Prakash, Yeo Wei Jie, Amir Abdullah, Ranjan Satapathy, Erik Cambria, and Roy Ka-Wei Lee.

Saxena, Rohit, Alessandro Suglia, and Pasquale Minervini. "VLM-RobustBench: A Comprehensive Benchmark for Robustness of Vision-Language Models." *arXiv*, 6 Mar. 2026, arxiv.org/abs/2603.06148. Accepted at ICML 2026.

"The Refusal Direction." *Learn Mechanistic Interpretability*, learnmechinterp.com/topics/refusal-direction/. Accessed 20 July 2026.

"The Visual Refusal Gap: A Mechanistic Analysis of Cross-Modal Safety Alignment Failure in Vision-Language Models." *Prezi*, prezi.com/p/sqm82oibsddw/the-visual-refusal-gap-a-mechanistic-analysis-of-cross-modal-safety-alignment-failure-in-vision-language-models/.

Wang, Yu, et al. "AdaShield: Safeguarding Multimodal Large Language Models from Structure-Based Attack via Adaptive Shield Prompting." *arXiv*, 14 Mar. 2024, arxiv.org/abs/2403.09513. Authors: Yu Wang, Xiaogeng Liu, Yu Li, Muhao Chen, and Chaowei Xiao.

Wollschläger, Tom, et al. "The Geometry of Refusal in Large Language Models: Concept Cones and Representational Independence." International Conference on Machine Learning. PMLR, 2025.

Xu, Shicheng, et al. "Cross-Modal Safety Mechanism Transfer in Large Vision-Language Models" (TGA). *arXiv*, 16 Oct. 2024 (rev. 28 Feb. 2025), arxiv.org/abs/2410.12662. Authors: Shicheng Xu, Liang Pang, Yunchang Zhu, Huawei Shen, and Xueqi Cheng. Published as a conference paper at ICLR 2025.

Yang, Jiaxi, et al. "Steering to Say No: Configurable Refusal via Activation Steering in Vision Language Models" (CR-VLM). *arXiv*, 31 Jan. 2026, arxiv.org/abs/2602.07013. Authors: Jiaxi Yang, Shicheng Liu, Yuchen Yang, and Dongwon Lee.

Zong, Yongshuo, et al. "Safety Fine-Tuning at (Almost) No Cost: A Baseline for Vision Large Language Models" (VLGuard). *Proceedings of the 41st International Conference on Machine Learning*, 2024, arxiv.org/abs/2402.02207. Authors: Yongshuo Zong, Ondrej Bohdal, Tingyang Yu, Yongxin Yang, and Timothy Hospedales.

Zou, Andy, et al. "Universal and Transferable Adversarial Attacks on Aligned Language Models." *arXiv*, 27 July 2023, arxiv.org/abs/2307.15043. Authors: Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson.

Zou, Xiaohan, et al. "Understanding and Rectifying Safety Perception Distortion in VLMs" (ShiftDC). *arXiv*, 18 Feb. 2025, arxiv.org/abs/2502.13095. Authors: Xiaohan Zou, Jian Kang, George Kesidis, and Lu Lin.