Reproducing HiddenDetect: Detecting Jailbreak Attacks on Vision-Language Models via Hidden State Monitoring*

Jace Bolante Phannawich Jadpotwanich Po-Hui Lin Wenjia Xia
Oregon State University

Abstract

Large vision-language models (LVLMs) are vulnerable to jailbreak attacks, and the visual channel widens the attack surface. HiddenDetect addresses this by monitoring internal activations, scoring each layer's alignment with a refusal vector and integrating that layer-wise signal into a training-free detection score. We reproduce HiddenDetect on LLaVA-1.6-7B Vicuna and isolate two underexplored design choices: how the signal is aggregated and which layers feed into it. Holding feature extraction fixed, we compare the original trapezoidal integration against Fisher discriminant analysis and logistic regression under a leakage-controlled k-fold protocol across five benchmarks. Over our reproduced baseline, full-depth Fisher aggregation raises AUPRC by 0.0234 and AUROC by 0.0210 on average without benchmark regressing, and logistic regression performs comparably, suggesting safe and unsafe prompts are nearly linearly separable in the deeper layers. Confining aggregation to the original window (layers 16 to 29) erases most of this gain, indicating that refusal information is spread across more layers than previously assumed and that layer selection matters as much as the aggregation operator.

1. Introduction

Large vision-language models (LVLMs) have demonstrated impressive capabilities across a broad range of multimodal tasks, including visual reasoning, image understanding, instruction following, and grounded question answering [13, 17]. As these systems become increasingly integrated into real-world applications, concerns regarding safety and robustness have attracted increasing attention. Similar to large language models (LLMs), LVLMs remain vulnerable to jailbreak attacks that circumvent alignment mechanisms and induce unsafe or policy-violating outputs. The incorporation of visual inputs further enlarges the attack surface, enabling adversaries to exploit both textual and image-based channels to bypass existing safeguards [6, 16].

Recent studies suggest that safety-related behaviors may be reflected in the internal representations of foundation models rather than solely in their final outputs [1]. Building on this observation, HiddenDetect [10] investigated whether LVLMs encode safety-relevant information within their hidden states during inference. The key insight is that unsafe prompts induce activation patterns that differ systematically from those produced by benign inputs. By projecting hidden states into vocabulary space and measuring their alignment with a predefined refusal vector, HiddenDetect constructs a layer-wise refusal signal that serves as an indicator of prompt safety. Aggregating this signal over a safety-aware layer interval identified through a small calibration set yields a scalar score that enables jailbreak detection without requiring additional model training.

Although HiddenDetect demonstrates strong empirical performance across multiple benchmarks, several aspects of the method remain insufficiently understood. First, the aggregation stage relies exclusively on trapezoidal integration, and it remains unclear whether alternative aggregation mechanisms can exploit the same hidden-state information more effectively. Second, the detector operates on a predefined safety-aware layer interval derived from a few-shot calibration procedure, yet the sensitivity of detection performance to layer selection has not been systematically evaluated. Third, once supervised aggregation strategies are introduced, evaluation becomes susceptible to label leakage unless calibration and testing data are rigorously separated. These issues are particularly important in reproducibility studies because they influence both the interpretation of reported performance gains and the broader understanding of the underlying safety signal.

To address these questions, this work presents a component-level empirical study of HiddenDetect under a unified experimental framework. The original detector is re-implemented in modular form, allowing layer selection and score aggregation to be modified independently while all other components remain unchanged. This formulation enables a controlled investigation of the relative contribu-

*Our modular implementation and evaluation framework are publicly available at https://github.com/appleboblin/HiddenDetect.

tion of individual design choices to overall detection performance. In addition to reproducing the original training-free formulation, we examine two supervised aggregation strategies, namely Fisher discriminant analysis and logistic regression, operating on the same layer-wise refusal features extracted by HiddenDetect.

Experiments are conducted on five publicly available jailbreak-detection benchmarks spanning text-only attacks, multimodal jailbreaks, and adversarial image scenarios. The detection scores we measure for the original aggregation scheme do not completely match the values reported in the original study, and the differences run in both directions across benchmarks, so we treat the baseline we observe as the reference point for all subsequent comparisons rather than the published numbers. Across the evaluated benchmarks, Fisher-based aggregation generally achieves higher AUPRC and AUROC values than the original trapezoidal formulation. These results indicate that the refusal signal contains discriminative information that is not fully utilized by uniform integration. Furthermore, we observe that useful safety-related information extends beyond the originally selected safety-aware layer interval. This finding complements the original safety-awareness analysis by suggesting that refusal semantics are distributed more broadly throughout network depth than previously assumed.

The main contributions of this work are summarized as follows:

- We provide a reproduction of HiddenDetect and conduct the first systematic comparison of multiple aggregation strategies for HiddenDetect, including the original trapezoidal integration scheme, Fisher discriminant analysis, and logistic regression.
- We demonstrate that Fisher-based aggregation consistently improves detection performance over the reproduced HiddenDetect baseline, achieving higher AUPRC and AUROC values across all evaluated datasets.
- We analyze the influence of layer selection and show that useful safety-related information extends beyond the originally identified safety-aware interval, indicating that refusal semantics are distributed across a wider range of network layers than previously recognized.
- We establish a leakage-controlled evaluation protocol for supervised aggregation methods and discuss its implications for future research on hidden-state-based jailbreak detection.

Overall, the results suggest that the effectiveness of HiddenDetect arises from two complementary factors: the existence of a robust layer-wise refusal signal and the mechanism used to aggregate that signal into a detection score. While the original training-free design already provides strong detection capability, alternative aggregation strategies can further improve performance when evaluated under a rigorous and reproducible experimental protocol.

2. Related Work

2.1. Jailbreak Attacks Against LVLMs

Jailbreak attacks against Large Vision-Language Models (LVLMs) attempt to exploit the expanded attack surface introduced by the visual modality. We can separate these attacks into three different categories, the first is typographic attacks, which embed instructions directly into the image that a human would be able to notice but should still avoid the safety-filters of the LVLM [6, 14, 16]. Gong et al. created a benchmark using this strategy and found that the ASR (Attack Success Rate) was significantly higher than the equivalent text prompts [6]. The second category is gradient based adversarial attacks which create images that contain harmful instructions but appear benign to humans [3, 15, 19, 22, 25, 30]. Qi et al. found that an adversarial image created for one unsafe behavior can generalize to other unrelated harmful queries and that the required changes are minor enough that it can evade the LVLMs defenses [19]. Some benchmarks, such as MM-Safety, contain examples that fit in both the first two categories [14]. The third category is text based attacks that also include arbitrary images. Li et al. found that adding unrelated images to unsafe text prompts increased the ASR [12]. A benchmark created by Luo et al. called JailbreakV-28k combines all three of these categories together [16].

2.2. Safety Defenses and Jailbreak Detectors

Defenses against LVLM jailbreaks can be split into training-time and inference-time approaches. Training-time methods like safety fine-tuning [20] and RLHF-style alignment [2, 18] improve robustness but require large curated datasets, are expensive to update as new attacks emerge, and can cause over-refusal on benign inputs [21]. Inference-time detectors avoid retraining. Input-perturbation detectors like JailGuard [29], MirrorCheck [4], and CIDER [28] probe behavioral consistency under augmentation, but typography-based attacks are specifically designed to be visually stable, and running multiple forward passes per query adds latency that matters in practice. Gradient-based detectors such as GradSafe [27] look for harmful pattern signatures in safety-critical parameter gradients, however it was designed for text-only LLMs and loses discriminative power when the dominant malicious signal lives in the visual token stream rather than the text. Gradient Cuff [8] has the same no-visual limitation. Prompt-engineering defenses that prepend fixed safety instructions to every query [26] are similarly limited, as small perturbations to the input phras-

ing can cause the safety prefix to be ignored or overridden by the model. Our work builds directly on top of HiddenDetect [10], which sidesteps these failure modes by operating directly on the model's internal activations rather than its input surface or output behavior.

2.3. Internal Activations and Refusal Mechanisms in LLMs

A growing body of mechanistic interpretability work shows that aligned LLMs do not simply learn refusal as a behavioral policy, but rather, they encode it geometrically inside their residual stream. Zou et al. [31] showed that concepts such as honesty and harm occupy linear subspaces of the residual stream. Arditi et al. found that refusal behavior in aligned LLMs is mediated by a single low-dimensional direction in activation space, and that ablating this direction suffices to jailbreak the model [1].

Li et al. extended this line of work specifically to the question of which layers carry safety-relevant information in aligned LLMs [11]. They identified a contiguous band of middle-to-late layers where scaling layer weights substantially changes the model's over-rejection rate on benign queries that contain superficially toxic keywords. Their finding that early layers are largely safety-inert but middle layers are highly sensitive foreshadowed the layer-dependent patterns that HiddenDetect later documents for LVLMs [10].

For LVLMs specifically, HiddenDetect [10] is, to our knowledge, the first work to port this hidden-state analysis to the multimodal setting, showing that safety signals emerge later across depth when visual inputs are present and that this delay correlates with higher attack success rates. Song et al. arrive at a complementary conclusion from the attacker's side in JailBound [23]. JailBound fits one logistic regression classifier per fusion layer to approximate the local safety decision hyperplane, then extracts the classifier's normal vector to guide adversarial perturbations. Their Safety Boundary Probing stage shows that classifiers trained on layers five and beyond achieve near-perfect probing accuracy, while early layers are essentially random, which confirms that linear separability of safe and unsafe representations is a real property of deeper LVLM activations rather than a modeling artifact. The reason this matters for our work is that JailBound and one of our proposed aggregators use the exact same analytical machinery, per-layer logistic regression on intermediate representations, just pointed in opposite directions: they use it to steer an attack, and we use it to block one.

2.4. Aggregation of Layer-Wise Signals

Once per-layer refusal scores are computed, some aggregation is needed to produce a single detection decision. HiddenDetect's original choice is trapezoidal integration of the refusal strength vector $F \in \mathbb{R}^L$ over a layer window $[s, e]$, which amounts to a uniformly weighted sum of the selected scores [10]. The implicit assumption is that the refusal signal can be treated as samples of a continuous function of layer depth, and that all layers in the window contribute equally. Neither assumption is obviously justified given that transformer layers have heterogeneous roles and the refusal signal in Figure 2 of the original paper is clearly non-uniform across depth.

This motivates replacing the fixed uniform weighting with a data-driven one. The probing literature offers a precedent: layer-wise aggregation has been studied mainly for encoding linguistic features rather than safety signals [9, 24], and the recurring finding is that different layers specialize in different levels of abstraction and that simple learned aggregators often outperform fixed-weight alternatives. The natural learned linear alternatives are two standard discriminative linear classifiers [7]: Fisher's linear discriminant analysis [5], which weights layers by their class separation, and logistic regression, which fits the layer weights by minimizing classification loss. JailBound [23] provides direct empirical support for the latter: as discussed above, it fits per-layer logistic regression classifiers on exactly this kind of layer-indexed representation and reaches near-perfect accuracy in deeper layers, confirming that safe and unsafe representations are close to linearly separable there. We define both aggregators in Section 3.2. Whether learned aggregation actually helps for refusal semantics in LVLMs, and at what supervision cost, is one of the central questions our experiments address.

3. Methodology

This work conducts a component-level empirical analysis of HiddenDetect [10]. Rather than modifying the hidden-state representation used by the original detector, we retain the refusal-signal extraction pipeline and focus on two design choices that directly influence detection performance: score aggregation and layer selection. The resulting framework enables a controlled comparison between the original training-free formulation and several supervised aggregation strategies while keeping all other components unchanged.

3.1. Reproduced HiddenDetect Pipeline

HiddenDetect detects jailbreak attacks by monitoring internal activations of a pre-trained LVLM. Following the original implementation, hidden states from each transformer layer are projected into vocabulary space and compared with a predefined refusal vector. The resulting layer-wise refusal signal is then aggregated into a scalar detection score.

Refusal Vector Construction

The detector represents refusal semantics using a set of refusal-related vocabulary tokens. Consistent with the original implementation, the refusal token set includes terms frequently observed in model refusals, such as "sorry", "unfortunately", "illegal", "harmful", and "dangerous" [10].

Let

$$\mathcal{R} = \{r_1, r_2, \ldots, r_K\} \tag{1}$$

denote the refusal token set, where K is the number of refusal-related tokens. A binary refusal vector

$$\mathbf{r} \in \mathbb{R}^V \tag{2}$$

is constructed in vocabulary space, where V denotes the vocabulary size. Entries corresponding to refusal tokens are assigned a value of one, while all remaining entries are set to zero.

Layer-Wise Refusal Signal Extraction

For an input prompt, the LVLM produces hidden representations at each transformer layer. Let

$$\mathbf{h}_l \tag{3}$$

denote the final-token hidden state from layer l after projection through the language-modeling head.

Following HiddenDetect, the alignment between the projected hidden state and the refusal vector is measured using cosine similarity:

$$F_l = \frac{\mathbf{h}_l^\top \mathbf{r}}{\|\mathbf{h}_l\| \, \|\mathbf{r}\|}, \tag{4}$$

where F_l represents the refusal strength at layer l.

The resulting refusal curve is defined as

$$\mathbf{F} = [F_1, F_2, \ldots, F_L], \tag{5}$$

where L is the number of transformer layers. Larger values indicate stronger alignment with refusal semantics and therefore a higher likelihood that the prompt is unsafe.

Original HiddenDetect Aggregation

The original HiddenDetect method converts the refusal curve into a scalar score using trapezoidal integration over a predefined safety-aware layer interval.

For LLaVA-1.6, the calibration procedure described in [10] identifies layers 16–29 as the most safety-aware region. The detection score is computed as

$$S_{\text{trapz}} = \int_{16}^{29} F(l)\, dl, \tag{6}$$

which is numerically approximated using the trapezoidal rule.

This formulation serves as the baseline reproduced throughout this study.

3.2. Alternative Aggregation Strategies

The refusal curve contains substantially more information than a single integrated value. To determine whether the original aggregation scheme fully exploits this information, two alternative aggregation strategies are investigated while keeping the feature extraction process unchanged.

Fisher Discriminant Aggregation

The first alternative employs Fisher discriminant analysis. Instead of assigning equal importance to all layers within the selected interval, Fisher aggregation emphasizes layers that provide stronger separation between safe and unsafe samples.

Let $\mu_{u,l}$ and $\mu_{s,l}$ denote the mean refusal scores at layer l for unsafe and safe samples, respectively. Likewise, $\sigma^2_{u,l}$ and $\sigma^2_{s,l}$ represent the corresponding layer-wise variances.

The Fisher weight vector is defined as

$$\mathbf{w} = \frac{\boldsymbol{\mu}_u - \boldsymbol{\mu}_s}{\sqrt{\frac{1}{2}\left(\boldsymbol{\sigma}_u^2 + \boldsymbol{\sigma}_s^2\right) + \varepsilon}}, \tag{7}$$

where ε is a small constant introduced for numerical stability.

The final Fisher score is then computed as

$$S_{\text{Fisher}} = \mathbf{w}^\top \mathbf{F}. \tag{8}$$

Unlike trapezoidal integration, Fisher aggregation explicitly incorporates class-discriminative information and allows informative layers to contribute more strongly to the final decision. Our implementation is not full linear discriminant analysis, which would require inverting the within-class scatter matrix, but rather a modified diagonal (layer-independent) form of Fisher's discriminant. We use the diagonal form since it is more stable to estimate on the small calibration sets used here.

Logistic Regression Aggregation

The second alternative formulates aggregation as a supervised binary classification problem. The complete refusal curve is treated as a feature vector and passed to a logistic regression classifier.

The probability that a sample is unsafe is given by

$$P(y = 1|\mathbf{F}) = \frac{1}{1 + \exp[-(\mathbf{w}^\top \mathbf{F} + b)]}, \tag{9}$$

where $y = 1$ denotes an unsafe sample.

Logistic regression learns the layer weights directly from the classification loss. Unlike Fisher aggregation, which scores each layer on its own univariate class separation, logistic regression fits the weights jointly and so can account for correlations between layers. We use both in our experiments to see which aggregation strategy performs better in practice.

3.3. Leakage-Controlled Evaluation Protocol

The original HiddenDetect formulation is training-free and therefore does not require label-dependent calibration. In contrast, both Fisher aggregation and logistic regression use class labels during parameter estimation. Evaluating these methods on the same data used for calibration would introduce label leakage and result in overly optimistic performance estimates.

To prevent this issue, a stratified K-fold evaluation protocol is adopted. For each fold, aggregation parameters are estimated using only the training partition and subsequently applied to the held-out test partition. This procedure ensures that every reported prediction is generated from out-of-fold data and therefore provides an unbiased estimate of generalization performance.

3.4. Benchmark Datasets and Evaluation Metrics

Experiments are conducted on five benchmark datasets covering text-only jailbreak attacks, multimodal jailbreak attacks, and adversarial image scenarios. The evaluation suite includes:

- XSTest,
- FigTxt,
- MM-SafetyBench + MM-Vet,
- FigImg + MM-Vet,
- JBV28K_JBtxt + MM-Vet.

Performance is evaluated using the area under the precision-recall curve (AUPRC) and the area under the receiver operating characteristic curve (AUROC). AUPRC is particularly informative in the presence of class imbalance, while AUROC provides a threshold-independent measure of ranking quality. Together, these metrics provide a comprehensive assessment of jailbreak detection performance.

4. Results

4.1. Experimental Setting

The aggregation configurations defined in Section 3.2 are evaluated on five jailbreak-detection benchmarks: XSTest, FigTxt, FigImg, MM-SafetyBench, and JailBreakV-28Ktxt. Each unsafe benchmark is paired with the MM-Vet safe distribution following the protocol of HiddenDetect [10]. The resulting evaluation sets contain between 218 and 450 samples. XSTest and FigTxt retain their native class distributions, whereas the remaining benchmarks are downsampled to 218 unsafe and 218 safe samples prior to evaluation.

Performance is measured using the area under the precision–recall curve (AUPRC) and the area under the receiver operating characteristic curve (AUROC). AUPRC is particularly informative under class imbalance, while AUROC provides a threshold-independent measure of ranking quality. Inference is performed in float16 on a single H100-class GPU. Our five configurations (four supervised and original HiddenDetect baseline) are evaluated under the stratified 5-fold protocol described in Section 3.3. All sampling, splitting, and model initialization use a fixed random seed of 539.

4.2. Aggregation Operators and Layer Range

Table 1 and Table 2 summarize AUPRC and AUROC across all five benchmarks. The Paper-default column applies the original HiddenDetect aggregation scheme, trapezoidal integration over layers 16–29, and serves as the baseline for every comparison in this study. The scores we measure do not match the originally reported values: on the five benchmarks that align with the original LLaVA evaluation, AUROC differs by roughly 0.01 to 0.07 and in both directions, with our baseline scoring lower on FigTxt and MM-SafetyBench and higher on FigImg and JailBreakV-28K. We attribute this difference to the probabilistic way each unsafe set is paired and downsampled against MM-Vet (Section 4.1) as well as potential updates to the benchmarks. We therefore treat the Paper-default column as a qualitative reproduction and report all subsequent improvements relative to it rather than to the published numbers.

Three observations emerge from these results.

First, Fisher aggregation over the full transformer depth (C2) consistently improves upon the reproduced HiddenDetect baseline. The largest gains occur on XSTest and FigImg + MM-Vet. For XSTest, AUPRC increases from 0.7974 to 0.8675, while AUROC rises from 0.8581 to 0.9061. For FigImg + MM-Vet, AUPRC improves from 0.9063 to 0.9419 and AUROC from 0.9158 to 0.9552. Across all five benchmarks, Fisher aggregation improves AUPRC by an average of 0.0234 and AUROC by an average of 0.0210 relative to the reproduced HiddenDetect baseline.

Second, restricting Fisher aggregation to the original safety-aware interval (layers 16–29) removes most of the observed improvement. The largest differences between Fisher-all and Fisher-[16,29] occur on XSTest (+0.0672 AUPRC, +0.0430 AUROC) and FigImg + MM-Vet (+0.0357 AUPRC, +0.0412 AUROC). These results indicate that layer selection contributes substantially to the performance gap between the original baseline and full-stack Fisher aggregation. Although the aggregation operator itself remains important, the choice of layer range appears to be an equally influential factor.

Third, the supervised logistic-regression variants produce performance comparable to Fisher aggregation. C4 ($C = 1$) and C5 ($C = 0.5$) remain within approximately 0.005 of Fisher-all on most datasets, with FigImg + MM-Vet the clearest exception, where Fisher-all leads by

Table 1. AUPRC across five jailbreak-detection benchmarks for the five aggregation configurations with k-fold. Best performance in each row is shown in bold.

Dataset	Paper Default	Fisher All	Fisher $[16, 29]$	LogReg ($C = 1$)	LogReg ($C = 0.5$)
XSTest	0.7974	**0.8675**	0.8003	0.8617	0.8612
FigTxt	0.9638	**0.9682**	0.9650	0.9669	0.9670
MM-SafetyBench + MM-Vet	0.9738	**0.9771**	0.9743	0.9747	0.9744
FigImg + MM-Vet	0.9063	**0.9419**	0.9062	0.9292	0.9285
JBV28K_JBtxt + MM-Vet	0.9595	**0.9633**	0.9601	0.9621	0.9614

Table 2. AUROC across five jailbreak-detection benchmarks for the five aggregation configurations with k-fold. Best performance in each row is shown in bold.

Dataset	Paper Default	Fisher All	Fisher $[16, 29]$	LogReg ($C = 1$)	LogReg ($C = 0.5$)
XSTest	0.8581	**0.9061**	0.8631	0.9019	0.9016
FigTxt	0.9338	**0.9433**	0.9363	0.9408	0.9409
MM-SafetyBench + MM-Vet	0.9694	**0.9738**	0.9702	0.9706	0.9702
FigImg + MM-Vet	0.9158	**0.9552**	0.9140	0.9412	0.9401
JBV28K_JBtxt + MM-Vet	0.9590	**0.9627**	0.9596	0.9613	0.9604

Table 3. Sensitivity of full-stack Fisher aggregation to the stabilization constant ε.

Dataset	Metric	10^{-8}	10^{-3}	10^{-2}	5×10^{-2}
XSTest	AUPRC	**0.8740**	0.8728	0.8684	0.8652
	AUROC	**0.9088**	0.9077	0.9046	0.9030
FigTxt	AUPRC	**0.9684**	0.9680	0.9675	0.9672
	AUROC	**0.9436**	0.9428	0.9418	0.9412

roughly 0.013 AUPRC and 0.014 AUROC. The two configurations otherwise track each other closely, and neither logistic-regression variant dominates Fisher aggregation on any benchmark. This observation suggests that the refusal signal is approximately linearly separable in the deeper layers, allowing simple discriminative weighting schemes to capture most of the useful information contained in the hidden-state representation.

4.3. Sensitivity to the Fisher Stabilization Constant

The Fisher weighting scheme includes a small stabilization constant ε to prevent division by near-zero pooled variance. Table 3 reports the sensitivity of Fisher aggregation to this parameter on two representative datasets.

The default value $\varepsilon = 10^{-8}$ yields the strongest performance, although the dependence on ε is weak. Increasing ε by several orders of magnitude causes only modest degradation. On XSTest, the largest observed reduction is 0.0088 AUPRC, while FigTxt exhibits an AUROC decrease of only 0.0024. These results indicate that the Fisher weighting mechanism is robust and does not rely on precise tuning of the stabilization parameter.

4.4. Comparison with Prior Approaches

Within the family of inference-time LVLM jailbreak detectors, the reproduced HiddenDetect baseline remains the most appropriate reference point for this study [10]. Relative to that baseline, full-stack Fisher aggregation improves average AUPRC by 0.020 and average AUROC by 0.018 across the five benchmarks, with no benchmark exhibiting performance degradation.

A direct numerical comparison with perturbation-based and gradient-based detectors is beyond the scope of this study because those methods were evaluated on different benchmark configurations and often require substantially different computational budgets. Consequently, the reproduced HiddenDetect baseline provides the most meaningful point of comparison. Under this controlled setting, the results demonstrate that improvements in aggregation alone are sufficient to produce measurable gains across diverse jailbreak-detection benchmarks while preserving the original hidden-state representation proposed by HiddenDetect.

5. Conclusion

In this work we set out to reproduce HiddenDetect, to understand where its detection performance comes from, and to attempt to improve on it. Our work focused mostly on the aggregation step of the HiddenDetect pipeline. We found that the refusal curve carries more usable signal than the original trapezoidal integration pulls out of it. A Fisher projection improves AUPRC by an average of 0.0234 and AUROC by 0.0210 across all five benchmarks, with no dataset getting worse, and logistic regression lands in roughly the same place. Two different linear aggregators converging on similar performance tells us the safe-versus-unsafe split is

close to linearly separable in the deeper layers, and that most of the gain comes from reweighting layers rather than from anything nonlinear. The original trapezoidal operator gives up that signal because it weights every layer in its window equally and cannot tell an informative layer from a noisy one.

The second finding is that a good part of that gain is about *which* layers get used, not just *how* they get combined. Confining Fisher aggregation to the original safety-aware window of layers 16 to 29 erases most of the improvement, with XSTest and FigImg + MM-Vet losing the most. Useful refusal information sits outside the interval selected in the original HiddenDetect, which agrees with the original observation that safety awareness is distributed across depth but pushes it a step further: the calibrated window leaves usable signal unused, and treating layer selection as a fixed preprocessing step costs more than the aggregation choice alone suggests.

There are also limitations of the work that we present. The original HiddenDetect paper evaluated Llava-1.6-7B Vicuna, CogVLM-chat-v1.1, and Qwen-VL-Chat. However we only evaluate Llava-1.6-7B Vicuna, so we cannot say whether the wider layer coverage holds for other LVLMs or larger models. Additionally, the supervised aggregators that we use get their improvement by using labeled calibration data, which gives up the original method's training-free property, it is important to consider whether the improvement that we observed is worth it considering the additional time cost.

There are a few possible direction that this work could head in the future, the most obvious is to apply our aggregation methods on other LVLM's such as CogVLM and Qwen-VL like the original HiddenDetect did. Another possible step would be to train the aggregators on one dataset, and test them on another to investigate how well it generalizes between datasets. Finally, further investigation into which layers contain the largest refusal signal could be conducted.

References

[1] Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction, 2024. 1, 3

[2] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022. 2

[3] Yinpeng Dong, Huanran Chen, Jiawei Chen, Zhengwei Fang, Xiao Yang, Yichi Zhang, Yu Tian, Hang Su, and Jun Zhu. How robust is google's bard to adversarial image attacks?, 2023. 2

[4] Samar Fares, Klea Ziu, Toluwani Aremu, Nikita Durasov, Martin Takac, Pascal Fua, Karthik Nandakumar, and Ivan Laptev. MirrorCheck: Efficient adversarial defense for vision-language models. *arXiv preprint arXiv:2406.09250*, 2024. 2

[5] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936. 3

[6] Yichen Gong, Delong Ran, Jinyuan Liu, Conglei Wang, Tianshuo Cong, Anyu Wang, Sisi Duan, and Xiaoyun Wang. Figstep: Jailbreaking large vision-language models via typographic visual prompts, 2025. 1, 2

[7] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics)*. 02 2009. 3

[8] Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. Gradient cuff: Detecting jailbreak attacks on large language models by exploring refusal loss landscapes. *arXiv preprint arXiv:2403.00867*, 2024. 2

[9] Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does BERT learn about the structure of language? In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy, July 2019. Association for Computational Linguistics. 3

[10] Yilei Jiang, Xinyan Gao, Tianshuo Peng, Yingshui Tan, Xiaoyong Zhao, Bo Zheng, and Xiangyu Yue. HiddenDetect: Detecting jailbreak attacks against large vision-language models via monitoring hidden states. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025. 1, 3, 4, 5, 6

[11] Shen Li, Liuyi Yao, Lan Zhang, and Yaliang Li. Safety layers in aligned large language models: The key to LLM security. *arXiv preprint arXiv:2408.17003*, 2024. 3

[12] Yifan Li, Hangyu Guo, Kun Zhou, Wayne Xin Zhao, and Ji-Rong Wen. Images are achilles' heel of alignment: Exploiting visual vulnerabilities for jailbreaking multimodal large language models, 2025. 2

[13] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 1

[14] Xin Liu, Yichen Zhu, Jindong Gu, Yunshi Lan, Chao Yang, and Yu Qiao. MM-SafetyBench: A benchmark for safety evaluation of multimodal large language models. In *European Conference on Computer Vision (ECCV)*, 2024. 2

[15] Haochen Luo, Jindong Gu, Fengyuan Liu, and Philip Torr. An image is worth 1000 lies: Adversarial transferability across prompts on vision-language models, 2024. 2

[16] Weidi Luo, Siyuan Ma, Xiaogeng Liu, Xiaoyu Guo, and Chaowei Xiao. JailBreakV-28K: A benchmark for assessing the robustness of multimodal large language models against jailbreak attacks. *arXiv preprint arXiv:2404.03027*, 2024. 1, 2

[17] OpenAI. GPT-4V(ision) system card. *OpenAI Technical Report*, 2023. 1

[18] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2

[19] Xiangyu Qi, Kaixuan Huang, Ashwinee Panda, Peter Henderson, Mengdi Wang, and Prateek Mittal. Visual adversarial examples jailbreak aligned large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024. 2

[20] Xuankun Rong, Jianshuo Zou, Mingyu Li, Shun Cao, Tongtong Zhao, et al. A survey of safety on large vision-language models: Attacks, defenses and evaluations. *arXiv preprint arXiv:2502.14881*, 2024. 2

[21] Paul Röttger, Hannah Rose Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. Xstest: A test suite for identifying exaggerated safety behaviours in large language models, 2024. 2

[22] Erfan Shayegani, Yue Dong, and Nael Abu-Ghazaleh. Jailbreak in pieces: Compositional adversarial attacks on multimodal language models, 2023. 2

[23] Jiaxin Song, Yixu Wang, Jie Li, Rui Yu, Yan Teng, Xingjun Ma, and Yingchun Wang. Jailbound: Jailbreaking internal safety boundaries of vision-language models, 2026. 3

[24] Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics. 3

[25] Haoqin Tu, Chenhang Cui, Zijun Wang, Yiyang Zhou, Bingchen Zhao, Junlin Han, Wangchunshu Zhou, Huaxiu Yao, and Cihang Xie. How many unicorns are in this image? a safety evaluation benchmark for vision llms, 2023. 2

[26] Yu Wang, Xiaogeng Liu, Yu Li, Muhao Chen, and Chaowei Xiao. Adashield: Safeguarding multimodal large language models from structure-based attack via adaptive shield prompting, 2024. 2

[27] Yueqi Xie, Minghong Fang, Renjie Pi, and Neil Zhenqiang Gong. GradSafe: Detecting jailbreak prompts for LLMs via safety-critical gradient analysis. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024. 2

[28] Yue Xu, Xiuwen Qi, Zhan Qin, and Wenjie Wang. CIDER: A cross-modal iterative denoising defense for vision-language models against adversarial attacks. *arXiv preprint arXiv:2407.21659*, 2024. 2

[29] Xiaoyu Zhang, Cen Zhang, Tianlin Li, Yihao Huang, Xiaojun Jia, Ming Hu, Jie Zhang, Yang Liu, Shiqing Ma, and Chao Shen. JailGuard: A universal detection framework for LLM prompt-based attacks. *arXiv preprint arXiv:2312.10766*, 2023. 2

[30] Yunqing Zhao, Tianyu Pang, Chao Du, Xiao Yang, Chongxuan Li, Ngai-Man Cheung, and Min Lin. On evaluating adversarial robustness of large vision-language models, 2023. 2

[31] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023. 3