ICME 2026 Paper Title

Anonymous ICME submission

***Abstract*—This document is a model and instructions for LaTeX. This and the IEEEtran.cls file define the components of your paper [title, text, heads, etc.]. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract. The abstract should contain about 100 to 150 words, and should be identical to the abstract text submitted electronically.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Images in the wild often carry high-level harmful semantics such as hate, violence, or biased cues, and these semantics can leak into the responses of vision–language systems even when the user query is benign. Our goal is to filter out such harmful semantics *without* destroying low-level visual content that is necessary for normal perception and downstream reasoning.

Existing solutions are unsatisfactory, either because they are too coarse or because they require heavy system changes. Pixel-space obfuscation such as mosaic or blur removes harmful cues only by destroying image content, and it frequently discards task-relevant details. Training-based alignment or safety finetuning on the vision encoder or the entire VLM introduces substantial computation and maintenance cost, and it can degrade open-world generalization. Adding extra detectors such as OCR or safety classifiers increases latency and system complexity, while providing limited control over how much semantic information is actually removed. These limitations motivate intervention at the vision encoder *intermediate representation*, which is the narrow interface through which visual semantics are passed to the language model: compared to pixel editing or weight updates, a linear feature transform at this interface is the smallest controllable intervention, enabling selective suppression of an attribute subspace while preserving most background structure used for general perception.

We propose a training-free *spectral filtering* method that operates directly on high-level intermediate features of a pretrained vision encoder. Given a reference feature collection and a target feature collection associated with an attribute of interest, we estimate covariances (Σ_s, Σ_h), perform safe whitening, compute the generalized spectrum, and apply a simple per-eigenvalue shrinkage to obtain a filtered feature z'. At inference time, the transform is a fixed sequence of matrix–vector multiplications and element-wise scaling, yielding low overhead.

Our theoretical analysis explains why such a simple transform can reliably erase implicit semantics. Alignment-style image–text pretraining compresses predictive semantic factors into linearly readable directions in the visual representation space. When an attribute is predictable under text supervision, it induces a low-rank covariance shift in high-level features, producing a small number of eigenvalue spikes after safe whitening. Minimizing safe-metric distortion while penalizing attribute-associated quadratic energy yields a closed-form spectral shrinkage rule, which selectively suppresses the spiked directions with explicit energy and fidelity expressions.

In summary, we introduce (i) a training-free intermediate-feature filtering method for removing harmful visual semantics, (ii) a low-cost inference procedure with minimal change under a safe metric, and (iii) a unified theoretical chain from alignment training to generalized spectral spikes to closed-form shrinkage.

II. Related Work

A. Intermediate Representations in Transformer Models

A common finding across Transformer-based models is that many properties are recoverable from intermediate representations using simple linear probes, suggesting that these representations contain linearly accessible structure [1]–[3]. Related analyses also study how attention and layerwise activations relate to linguistic or semantic patterns [4], [5]. Beyond probing, representation editing and concept removal methods aim to suppress a specified attribute while minimally changing the representation, often via linear transformations or subspace projections [6]–[8]. Our work is aligned with this line, but targets visual encoder features inside VLMs and motivates a covariance-spectrum viewpoint that yields a training-free closed-form transform.

B. Vision Encoders in VLMs and Contrastive Alignment Training

Many modern VLMs use a pretrained vision encoder trained by image–text alignment objectives [9]–[12]. Such dual-encoder training makes image features directly comparable with text features through an inner product, which supports zero-shot transfer and also influences the geometry of high-level visual features. On top of these encoders, instruction-tuned VLMs connect the visual encoder to a language model using a learned projector that maps visual features into the language model input space [13], [14]. In this setting, interventions on intermediate visual features can change what semantics are available to the projector and the language model, which motivates studying controlled feature transforms at the visual-encoder level.

III. Methodology

A. Method Overview

Reference and target feature collections. We assume access to two sets of visual features extracted from the same visual encoder: (i) a *reference* set, and (ii) a *target* set

associated with the attribute of interest. These features can be collected offline by forwarding images through the visual encoder and recording representations at a chosen high-level layer.

Covariance estimation and whitening. From the reference feature set, we estimate the empirical covariance matrix $\Sigma_s \in \mathbb{R}^{d\times d}$. From the target feature set, we estimate the empirical covariance matrix Σ_h. We then compute a whitening transform based on Σ_s. Specifically, we perform an eigendecomposition

$$\Sigma_s = V \operatorname{diag}(\sigma) V^\top \tag{1}$$

where $\sigma_i \geq 0$ are the eigenvalues and V is an orthogonal matrix. The whitening matrix is defined as

$$W = \Sigma_s^{-1/2} := V \operatorname{diag}\Big((\sigma+\varepsilon)^{-1/2}\Big) V^\top \tag{2}$$

where a small $\varepsilon > 0$ is added for numerical stability. This transform maps reference features to a space with approximately identity covariance.

Generalized spectral decomposition. Using the whitening transform, we form the whitened target covariance

$$\Lambda = W \Sigma_h W^\top \tag{3}$$

We compute its eigendecomposition

$$\Lambda = U \operatorname{diag}(\lambda) U^\top \tag{4}$$

where λ_i and U denote the eigenvalues and eigenvectors, respectively. This step identifies a set of orthogonal directions in feature space that characterize variance differences between the target and reference distributions.

Spectral filtering. Given a visual representation z, we first whiten and rotate it:

$$\hat{z} = U^\top W z \tag{5}$$

We then apply a coordinate-wise spectral scaling

$$\hat{z}'_i = \frac{\beta}{\lambda_i + \beta}\, \hat{z}_i \tag{6}$$

where $\beta > 0$ is a scalar hyperparameter controlling the strength of suppression. Finally, the filtered representation is mapped back to the original space:

$$z' = W^{-1} U \hat{z}' \tag{7}$$

Implementation remarks. In practice, covariance estimation and spectral decomposition are performed once and cached. The Inference-time filtering operation itself consists of a sequence of matrix–vector multiplications and element-wise scaling, which means that the intermediate process overhead is relatively low. All experiments in this work use the same procedure, with β selected based on validation performance.

B. Theoretical Analysis

a) Goal: We explain why a simple eigenvalue reshaping applied to an intermediate vision-encoder feature can suppress an implicit semantic attribute while preserving most generic visual structure. The logic chain is: (i) alignment-based image–text pretraining organizes predictive semantics into a small linear subspace, (ii) an attribute that is predictable from text induces a low-rank covariance shift in high-level visual features, (iii) safe whitening converts this shift into a small number of eigenvalue spikes in a whitened covariance, (iv) a quadratic objective that balances safe-metric fidelity and target energy yields a closed-form spectral shrinkage that down-weights precisely those spiked directions.

b) Unified setup: Let $x \in \mathcal{X}$ and $t \in \mathcal{T}$. A dual encoder produces $v_\theta(x), u_\phi(t) \in \mathbb{R}^d$ and normalized embeddings $\tilde{v}(x) = v_\theta(x)/\|v_\theta(x)\|$, $\tilde{u}(t) = u_\phi(t)/\|u_\phi(t)\|$. A common score is cosine similarity with temperature $\tau > 0$,

$$s(x,t) = \frac{1}{\tau}\langle \tilde{v}(x), \tilde{u}(t)\rangle. \tag{8}$$

Given a minibatch $\{(x_i,t_i)\}_{i=1}^B$, define $y_{ij} = +1$ if $i = j$ and $y_{ij} = -1$ otherwise, and optimize

$$\mathcal{L} = \sum_{i=1}^{B}\sum_{j=1}^{B} \ell\big(y_{ij}\big(s(x_i,t_j) - b\big)\big), \tag{9}$$

where ℓ is monotone decreasing and b is optional. This form covers pairwise-sigmoid objectives and softmax-normalized contrastive objectives in the geometric sense needed here: increasing similarity for matched pairs and decreasing similarity for mismatched pairs.

c) Alignment training yields a low-dimensional semantic subspace: We write a more general score as $s(x,t) = g(v_\theta(x), u_\phi(t))$ with g differentiable. For an image embedding $v_i := v_\theta(x_i)$, the gradient has the form

$$\nabla_{v_i}\mathcal{L} = \sum_{j=1}^{B} \alpha_{ij}\, \nabla_v g(v_i, u_j), \tag{10}$$

for coefficients α_{ij} determined by ℓ and the pair labels. When g is bilinear (including inner product), $\nabla_v g(v,u)$ is linear in u:

$$g(v,u) = v^\top G u \quad\Longrightarrow\quad \nabla_v g(v,u) = Gu. \tag{11}$$

When g is an inner product followed by a shallow head, local linearization around a working point (v_0,u_0) gives

$$g(v,u) \approx g(v_0,u_0) + \langle J_v(v_0,u_0), v - v_0\rangle + \langle J_u(v_0,u_0), u - u_0\rangle, \tag{12}$$

so $\nabla_v g(v,u)$ lies in a low-dimensional span that is driven by text-side features through training. Together with (10), this implies that alignment training continuously shapes image representations along directions induced by the text representations, so semantic factors that help pair discrimination are encoded in a small linear subspace that is readable by inner products with text-induced directions.

d) Attribute signal from text induces an attribute direction in the image embedding space: Let $h \in \{0,1\}$ be a binary attribute and assume the text distribution depends on h:

$$\delta_u := \mathbb{E}[\tilde{u}(t) \mid h=1] - \mathbb{E}[\tilde{u}(t) \mid h=0] \neq 0. \tag{13}$$

Let $\mathcal{U} := \operatorname{span}\{\tilde{u}(t) : t \in \mathcal{T}\}$. Since $\delta_u \in \mathcal{U}$, take $w_h \propto \delta_u$. Alignment training makes $\langle \tilde{v}(x), w_h \rangle$ vary with h in distribution, so w_h acts as an attribute-related direction in the image embedding space.

e) From an attribute readout to a low-rank covariance shift at an intermediate layer.: Let $z \in \mathbb{R}^d$ denote the chosen high-layer (intermediate) representation in the vision encoder. For ViT-style vision encoders used in VLMs, the path from z to the final normalized embedding $\tilde{v}(x)$ includes pooling and a linear projection, followed by normalization; around typical operating points, we use a local linear readout

$$\tilde{v}(x) \approx Mz, \tag{14}$$

for some matrix M (absorbing pooling/projection and local linearization). Define the pulled-back attribute direction

$$a := M^\top w_h. \tag{15}$$

Then the attribute score satisfies

$$\langle \tilde{v}(x), w_h \rangle \approx \langle z, a \rangle. \tag{16}$$

We model the attribute effect on z as a low-dimensional modulation:

$$z = z_0 + A\gamma \tag{17}$$

where $\operatorname{rank}(A) = r \ll d, \operatorname{Cov}(z_0 \mid h) = \Sigma_0, \operatorname{Cov}(\gamma \mid h) = \Delta_h \succeq 0$ with $\mathbb{E}[\gamma \mid h] = 0$. This model is consistent with (16) when $a \in \operatorname{col}(A)$, since $\langle z, a \rangle$ depends on γ along the same low-dimensional subspace. From (17), the conditional covariance of z satisfies

$$\Sigma_h := \operatorname{Cov}(z \mid h) = \Sigma_0 + A\Delta_h A^\top, \tag{18}$$

and the group difference is

$$\Sigma_1 - \Sigma_0 = A(\Delta_1 - \Delta_0)A^\top, \qquad \operatorname{rank}(\Sigma_1 - \Sigma_0) \leq r. \tag{19}$$

This step explains why comparing two groups (reference versus target) is informative: the shared background term Σ_0 cancels, leaving a low-rank attribute-induced component.

f) Safe whitening reveals eigenvalue spikes: Let Σ_s be the covariance from a reference (safe) feature collection at the chosen layer, and let Σ_h be the covariance from a target (attribute) collection. Define the safe-whitening transform

$$W := \Sigma_s^{-1/2}, \tag{20}$$

and the whitened target covariance

$$\Lambda := W\Sigma_h W^\top. \tag{21}$$

When $\Sigma_h = \Sigma_s + \Delta$ with $\operatorname{rank}(\Delta) \leq r$, then

$$\Lambda = I + W\Delta W^\top, \qquad \operatorname{rank}(\Lambda - I) \leq r, \tag{22}$$

so only r eigen-directions can deviate from the background I. Let the eigendecomposition be

$$\Lambda = U \operatorname{diag}(\lambda)\, U^\top. \tag{23}$$

The directions with $\lambda_i > 1$ correspond to target-amplified variance directions, which align with the attribute subspace under (19)–(22).

g) Eigenvalue reshaping from a quadratic objective: Given a feature z, we construct a filtered feature z' by solving

$$\min_{z' \in \mathbb{R}^d} \ \frac{1}{2}\|W(z'-z)\|_2^2 + \frac{1}{2\beta}(Wz')^\top \Lambda (Wz'), \qquad \beta > 0. \tag{24}$$

The first term measures change under the safe Mahalanobis geometry induced by Σ_s. The second term penalizes energy along target-amplified directions through Λ.

Let $r = Wz$ and $r' = Wz'$. Since W is invertible (or using the Moore–Penrose pseudoinverse when Σ_s is singular), (24) is equivalent to

$$\min_{r' \in \mathbb{R}^d} \ \frac{1}{2}\|r' - r\|_2^2 + \frac{1}{2\beta} {r'}^\top \Lambda r'. \tag{25}$$

In the eigenbasis of Λ, define $\hat{z} := U^\top r$ and $\hat{z}' := U^\top r'$. Using $U^\top U = I$ and (23),

$$\|r' - r\|_2^2 = \|\hat{z}' - \hat{z}\|_2^2, \qquad {r'}^\top \Lambda r' = \sum_{i=1}^d \lambda_i (\hat{z}_i')^2, \tag{26}$$

so the objective decouples across coordinates:

$$\min_{\hat{z}' \in \mathbb{R}^d} \ \sum_{i=1}^d \left[\frac{1}{2}(\hat{z}_i' - \hat{z}_i)^2 + \frac{1}{2\beta}\lambda_i (\hat{z}_i')^2 \right]. \tag{27}$$

Setting derivatives to zero yields the closed-form shrinkage

$$\hat{z}_i' = \frac{\beta}{\lambda_i + \beta}\, \hat{z}_i, \qquad D := \operatorname{diag}\left(\frac{\beta}{\lambda + \beta} \right). \tag{28}$$

Mapping back gives the spectral transform

$$z' = W^{-1} U D U^\top W z. \tag{29}$$

By (22), only a small number of λ_i deviate from 1. Equation (28) therefore applies strong attenuation on the few spiked directions ($\lambda_i \gg 1$) while leaving the bulk near-background directions ($\lambda_i \approx 1$) close to identity scaling.

h) Coordinate-wise fidelity and suppression: Define the target energy and safe-metric distortion as

$$E_h(z') := (Wz')^\top \Lambda (Wz') = \sum_{i=1}^d \lambda_i (\hat{z}_i')^2, \tag{30}$$

$$\Delta_s(z', z) := \|W(z' - z)\|_2^2 = \sum_{i=1}^d (\hat{z}_i' - \hat{z}_i)^2. \tag{31}$$

Substituting (28) gives

$$E_h(z') = \sum_{i=1}^d \lambda_i \left(\frac{\beta}{\lambda_i + \beta} \right)^2 \hat{z}_i^2, \tag{32}$$

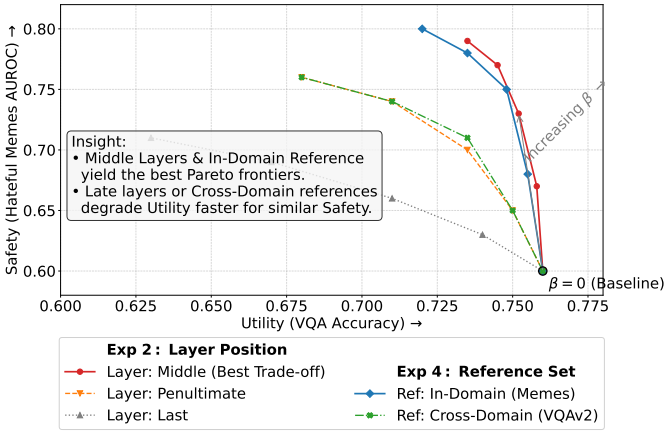

Fig. 1. Beta Sensitivity

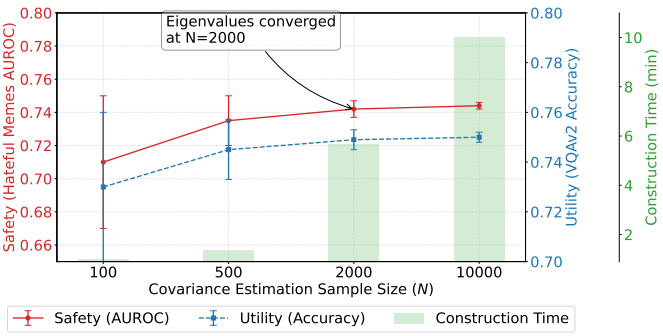

Fig. 2. Covariance Establishment Size

$$\Delta_s(z', z) = \sum_{i=1}^{d} \left(\frac{\lambda_i}{\lambda_i + \beta} \right)^2 \hat{z}_i^2. \tag{33}$$

These expressions quantify the trade-off controlled by β and connect eigenvalue reshaping to low-rank attribute structure through the spiked eigenvalues of Λ.

IV. Ablation and Sensitivity Studies

A. Sensitivity and Ablation Studies

To further understand the behavior and robustness of the proposed energy-based feature filtering, we conduct a series of targeted sensitivity experiments. These experiments are designed to isolate the effects of key design choices while keeping the backbone model and training protocol fixed. Quantitative results are reported on Hateful Memes for safety evaluation and VQAv2 for general visual–language utility.

a) Filtering strength β.: We first study the sensitivity to the filtering strength β, which controls the degree of spectral shrinkage applied to harmful directions. By sweeping β from no filtering to strong filtering at inference time, we examine the trade-off between harmful-content suppression and task utility. This experiment evaluates whether the method provides a smooth and controllable safety–utility trade-off rather than a brittle on/off behavior. The results are summarized in Figure **??**. *[Results and analysis omitted for brevity.]*

b) Injection layer.: We next evaluate the impact of the injection location within the visual encoder. The filtering module is inserted at different depths, ranging from intermediate layers to the final visual representation. This experiment tests the hypothesis that high-level visual features are more aligned with harmful semantic cues, while earlier features are more tied to low-level perception. Figure 1 reports the performance across layers. *[Results and analysis omitted for brevity.]*

c) Covariance estimation size.: Since the method relies on second-order statistics, we analyze its sensitivity to the number of samples used to estimate the safe and harmful covariance matrices. We vary the size of the reference and target sets while keeping all other factors fixed. This experiment evaluates the statistical stability of the learned filtering directions and whether reliable performance can be achieved with limited data. Results are shown in Figure 2. *[Results and analysis omitted for brevity.]*

TABLE I
Estimated overhead of reference fitting and inference-time filtering.

Setting	What is measured	Extra compute	Extra data movement	Extra persistent memory	Added latency
Baseline	vision forward	–	–	–	0
Filtered	vision forward + intervention	$\approx 1.70\times 10^9$ FLOPs / image	≈ 0 (if matrices cached on GPU)	≈ 3.83 MiB / layer	≈ 0.1–1.0 ms / image (device-cached)
Reference fit	Estimate τ from safe images	$\approx 1.70\times 10^{12}$ FLOPs total	≈ 5.06 GB total	Reservoir ≈ 0.46 MiB / layer	–

d) Reference set definition.: We further examine how the choice of the reference (safe) set affects the learned filtering geometry. Specifically, we compare using non-hateful samples from Hateful Memes versus images from a generic VQA dataset as the reference distribution. This experiment probes the robustness of the method to domain shifts in the definition of "safe" visual content. Figure 1 presents the comparison. *[Results and analysis omitted for brevity.]*

e) Overhead.: Finally, we evaluate its inference-time overhead. We estimate the overhead of reference fitting and inference-time filtering under our implementation setting (B=16, T=576, d=1152, k=384, C_{tok}=120,000). Using the same filtering configuration, we measure additional latency and memory cost introduced by the filtering operation. Table I shows that the filter adds only a small per-image latency and a few MiB of persistent state while preserving the main vision-forward computation, indicating that the intervention is lightweight and has limited impact on overall performance.

REFERENCES

[1] John Hewitt and Christopher D. Manning, "A structural probe for finding syntax in word representations," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019, pp. 4129–4138.

[2] John Hewitt and Percy Liang, "Designing and interpreting probes with control tasks," *arXiv preprint arXiv:1909.03368*, 2019.

[3] Yonatan Belinkov, "Probing classifiers: Promises, shortcomings, and advances," *Computational Linguistics*, vol. 48, no. 1, pp. 207–219, 2022.

[4] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning, "What does BERT look at? an analysis of BERT's attention," *arXiv preprint arXiv:1906.04341*, 2019.

[5] Jesse Vig and Yonatan Belinkov, "Analyzing the structure of attention in a transformer language model," in *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 2019.

[6] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg, "Null it out: Guarding protected attributes by iterative nullspace projection," *arXiv preprint arXiv:2004.07667*, 2020.

[7] Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan Cotterell, "Linear adversarial concept erasure," in *Proceedings of the 39th International Conference on Machine Learning (ICML)*. 2022, vol. 162 of *Proceedings of Machine Learning Research*, PMLR.

[8] Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman, "Leace: Perfect linear concept erasure in closed form," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[9] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al., "Learning transferable visual models from natural language supervision," *International Conference on Machine Learning (ICML)*, 2021.

[10] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig, "Scaling up visual and vision-language representation learning with noisy text supervision," in *Proceedings of the 38th International Conference on Machine Learning (ICML)*. 2021, vol. 139 of *Proceedings of Machine Learning Research*, PMLR.

[11] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer, "Lit: Zero-shot transfer with locked-image text tuning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[12] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer, "Sigmoid loss for language image pre-training," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[13] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee, "Visual instruction tuning," *arXiv preprint arXiv:2304.08485*, 2023.

[14] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li, "Llava-onevision: Easy visual task transfer," *arXiv preprint arXiv:2408.03326*, 2024.