The Dominance of Text Space: Unveiling the Asymmetric Nature of Cross-Modal Alignment in Large Language Models

Anonymous ACL submission

Abstract

Recent advancements in Multimodal Large Language Models (MLLMs) have largely been driven by aligning visual encoders with pre-trained Large Language Models (LLMs). While effective, the geometric nature of this alignment remains under-explored. Existing methods often assume a symmetric interaction between visual and textual modalities, implying that both spaces adapt to each other. In this paper, we challenge this assumption and propose the **"Text Space as Anchor"** hypothesis. We argue that the semantic space of LLMs is rigid, anisotropic, and dominant; thus, effective cross-modal alignment must be an asymmetric projection of visual features onto this pre-existing text manifold without distorting it. We identify a critical issue in current parameter-efficient tuning paradigms where task-specific visual adjustments inadvertently disrupt the projector's geometry, leading to "catastrophic forgetting" of the alignment mechanism itself. To address this, we introduce **Anchor-Preserving Projection (APP)**, a novel method that regularizes the projector to maintain the geometric structure of the text embedding space during task adaptation via spectral filtering. Extensive experiments on 8 diverse cross-modal tasks and 3 pure language benchmarks demonstrate that APP not only enhances transferability (+5.2% accuracy) but, crucially for the NLP community, preserves the LLM's inherent linguistic capabilities (e.g., MMLU, GSM8K) and reduces object hallucination significantly better than standard fine-tuning methods. We will release our code.

1 Introduction

The integration of vision and language has evolved from training task-specific models to leveraging the immense generalization capabilities of Large Language Models (LLMs) (Brown et al., 2020; Touvron et al., 2023). In typical Multimodal LLMs (MLLMs) like LLaVA (Liu et al., 2024b), MiniGPT-4 (Zhu et al., 2023), or Flamingo (Alayrac et al., 2022), a visual encoder is connected to a frozen LLM via a learnable projector (e.g., a linear layer or MLP). This architecture implicitly treats the LLM as a universal reasoning engine, with the visual encoder acting as a sensory organ.

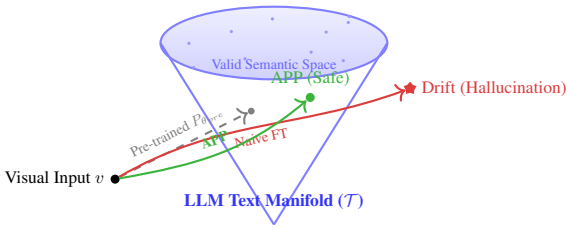

Figure 1: **The "Text Space as Anchor" Hypothesis.** The pre-trained LLM text space is anisotropic (cone-shaped). Naive fine-tuning (Red) minimizes task loss but causes *Alignment Drift*, projecting visual tokens into undefined regions, leading to hallucinations. Our method, APP (Green), constrains the projection to remain within the valid text manifold.

While the community has focused heavily on architectural variants and scaling instruction tuning datasets, the fundamental *mechanism* of how visual tokens inhabit the textual semantic space remains opaque. A common implicit assumption in fine-tuning strategies is that the modalities meet "halfway" or that the projector flexibly adapts visual features to text. Consequently, methods like LoRA (Hu et al., 2022) or full fine-tuning are applied to the projector without geometric constraints.

However, we argue that this perspective overlooks the **asymmetric dominance** of the LLM. The text embedding space, shaped by trillions of tokens during pre-training, possesses a highly structured and anisotropic geometry (Ethayarajh, 2019b; Gao et al., 2019). We hypothesize that for MLLMs to function robustly, the text space must act as a fixed **Anchor**. Visual features must be projected precisely onto this manifold.

We observe that standard Task Vector (Ilharco

et al., 2023) approaches—which effectively transfer styles in pure vision or pure language models—fail in the cross-modal projector setting. Our analysis reveals that naive fine-tuning of the projector distorts the geometric relationship required to map visual concepts to text tokens. This distortion creates a phenomenon we term **"Alignment Drift"**, where visual tokens, after task adaptation, land in the "void" regions of the text space. This not only hampers visual recognition but also degrades the LLM's core text reasoning capabilities, as the model is forced to process out-of-distribution embeddings.

To resolve this, we propose **Anchor-Preserving Projection (APP)**. APP treats the pre-trained alignment as a geometric constraint. It ensures that when the model adapts to new visual tasks (e.g., classifying bird species), the projector's transformation remains consistent with the global text manifold. Technically, we achieve this through spectral analysis of the weight updates, filtering out low-singular-value components that correspond to geometric noise while retaining the high-singular-value components that encode task-specific knowledge.

Our contributions are:

- We empirically and theoretically demonstrate the **asymmetric nature** of cross-modal alignment, positing the "Text Space as Anchor" hypothesis.
- We identify **Alignment Drift** as the root cause of failure in naive cross-modal fine-tuning, showing it harms both visual transfer and pure language reasoning.
- We introduce **APP**, a spectral regularization technique that significantly improves cross-modal task transfer (+5.2% accuracy on average).
- **Crucially for NLP:** We show that APP preserves the LLM's pure text reasoning capabilities (MMLU, GSM8K) better than baselines, suggesting that respecting the text manifold is key to safe multimodal adaptation.

2 Related Work

2.1 Multimodal Large Language Models

Recent works extend LLMs to vision by treating image patches as foreign language tokens. While earlier models like BLIP-2 (Li et al., 2023a) and LLaVA (Liu et al., 2024b) established the paradigm of using Q-Formers or linear projections, the field has shifted towards scaling visual resolution and optimizing architecture components. McKinlay et al. (2024) conducted extensive ablation studies, demonstrating that the image encoder's resolution and capacity are more critical than the projector's design. Concurrently, models like LLaVA-NeXT (Liu et al., 2024a) and Qwen2-VL (Wang et al., 2024a) have introduced dynamic resolution strategies to handle varying aspect ratios, significantly reducing hallucination in OCR and detail-oriented tasks. Despite these advancements, the projector module remains the critical bottleneck for semantic alignment. Our work investigates the *geometry* of this module, distinct from works that focus on scaling data or changing the visual backbone.

2.2 Task Vectors and Model Arithmetic

Task vectors (Ilharco et al., 2023) allow manipulating model behavior by adding weight differences ($\tau = \theta_{ft} - \theta_{pre}$). While Zhang et al. (2023) showed linear compositionality, recent research highlights the interference between parameters when merging distinct tasks. Techniques like TIES-Merging (Yadav et al., 2024) and DARE (Yu et al., 2024) address this by sparsifying task vectors and resolving sign conflicts, thereby preserving general capabilities during adaptation. However, these methods are primarily designed for homogeneous weight spaces (LLM-only). We show that their direct application in the heterogeneous Cross-Modal Projector space is suboptimal due to the high sensitivity of text-visual alignment. Our work extends this field by introducing spectral constraints to task arithmetic, specifically tailored for the multimodal interface.

2.3 Geometry of Language Spaces and Modality Gap

NLP research has established that embedding spaces are anisotropic, with representations occupying a narrow cone (Ethayarajh, 2019a). In the multimodal domain, this issue is exacerbated by the "Modality Gap"—a geometric phenomenon where image and text embeddings remain separated in the joint space even after contrastive pre-training (Liang et al., 2022). Recent studies in 2024 suggest that this gap persists in MLLMs and affects the "outlier dimensions" required for LLM processing (Udandarao et al., 2024). We build on the hypothesis that visual tokens must conform to the LLM's pre-existing "text cone." If a projector update pushes visual tokens into the modality gap or

distorts their geometry relative to the text manifold, the frozen LLM will fail to process them, leading to hallucinations.

3 The Geometry of Cross-Modal Alignment

In this section, we formalize the "Text Space as Anchor" hypothesis and analyze why standard adaptation methods fail to respect it through the lens of manifold perturbation theory.

3.1 Problem Formulation

Let $\mathcal{V} \subset \mathbb{R}^{d_v}$ be the visual feature space produced by a vision encoder (e.g., CLIP-ViT), and $\mathcal{T} \subset \mathbb{R}^{d_t}$ be the textual semantic space of an LLM. An MLLM defines a projection function $P_\theta : \mathcal{V} \to \mathcal{T}$, parameterized by weights $\theta \in \mathbb{R}^{d_t \times d_v}$.

Given a pre-trained projector θ_{pre} that aligns general visual concepts to text, task adaptation involves updating the weights to θ_{ft} using a task-specific dataset D_{task}. A task vector is defined as $\tau = \theta_{ft} - \theta_{pre}$. The goal is to apply τ to improve performance on the target task.

3.2 The Asymmetry Hypothesis and Anisotropy

Unlike bilingual translation where two languages might have comparable structural complexity, the relationship between $\mathcal{V}$ and $\mathcal{T}$ in MLLMs is asymmetric.

1. Rigidity of $\mathcal{T}$: The LLM is frozen. Its internal attention mechanisms and FFNs are optimized for a specific manifold of token embeddings. Crucially, this manifold is **Anisotropic**. As shown by Ethayarajh (2019a), valid token embeddings reside in a narrow cone. Let $\mathcal{C} \subset \mathbb{R}^{d_t}$ be this validity cone.

$$\forall x \in \text{Valid Tokens}, \quad \cos(x, \bar{x}) > \gamma \tag{1}$$

where $\bar{x}$ is the common mean direction and γ is a threshold.

2. Plasticity of P_θ: The projector is the only moving part. Its role is to map the dense visual distribution $\mathcal{V}$ onto the sparse text manifold $\mathcal{T}$.

Therefore, we posit: *The Text Space $\mathcal{T}$ acts as a rigid Anchor. Any modification to P_θ must be constrained such that the output distribution $P_{\theta_{ft}}(\mathcal{V})$ remains within the support of the pre-trained manifold $P_{\theta_{pre}}(\mathcal{V}) \approx \mathcal{C}$.*

3.3 Alignment Drift in Naive Fine-Tuning

Standard fine-tuning minimizes a task-specific loss $\mathcal{L}_{task}$ (e.g., cross-entropy on class names).

$$\theta_{ft} = \arg\min_\theta \mathcal{L}_{task}(D_{task}; \theta) \tag{2}$$

In high-dimensional spaces, there are infinite directions to descend the loss gradient. Many of these directions reduce $\mathcal{L}_{task}$ but move the weights θ in a way that distorts the global alignment.

We define **Alignment Drift** as the component of the weight update that is orthogonal to the principal directions of the pre-trained alignment. Mathematically, if the pre-trained alignment is dominated by the top singular vectors of θ_{pre}, drift occurs when τ introduces significant energy in directions corresponding to the null space or low-energy subspace of θ_{pre}. This causes visual tokens to be projected into "undefined" regions of the LLM's input space (outside $\mathcal{C}$), which we empirically verify leads to the degradation of pure language capabilities.

3.4 Theoretical Justification for Low-Rank Updates

Why should the update be low-rank? Consider the update τ as a perturbation. We want to find a τ^* that minimizes the deviation from the original manifold structure while maximizing task performance. Let the "safe" subspace be spanned by the top-k singular vectors of the pre-trained weight θ_{pre}, denoted as U_k. The projection of any update onto this subspace is $P_{U_k}(\tau)$. If we assume that the semantic concepts (e.g., "dog", "car") are aligned with the principal components of the text space, then the task-specific update should also lie primarily within this subspace.

Formally, we seek to solve an optimization problem where we minimize the Frobenius norm of the difference between the ideal task vector and our approximation, subject to a rank constraint:

$$\min_{\tau^*} ||\tau - \tau^*||_F \quad \text{s.t.} \quad \text{rank}(\tau^*) \leq k \tag{3}$$

By the Eckart-Young-Mirsky theorem, the optimal solution to this problem is the truncated Singular Value Decomposition (SVD) of τ. This provides the theoretical grounding for our method: spectral filtering is the optimal way to compress the task update into the "safe" semantic subspace under the Frobenius norm metric.

Furthermore, we can analyze the Lipschitz continuity of the projector. Let L_P be the Lipschitz

constant of P_θ. Unconstrained fine-tuning often increases L_P, making the projector sensitive to small perturbations in visual input δv, leading to large shifts in text space δt.

$$||\delta t|| \leq L_P ||\delta v|| \tag{4}$$

By restricting the singular values of the update τ, we implicitly regularize the spectral norm of the new projector, preventing the Lipschitz constant from exploding. This ensures that the mapping remains smooth and robust.

4 Method: Anchor-Preserving Projection (APP)

To operationalize the hypothesis, we propose Anchor-Preserving Projection (APP). Instead of using the raw task vector τ, we seek a filtered vector τ^* that captures task knowledge while minimizing alignment drift.

4.1 Spectral Decomposition of Task Vectors

We analyze the task vector $\tau = \theta_{ft} - \theta_{pre}$ using Singular Value Decomposition (SVD):

$$\tau = U \Sigma V^T = \sum_{i=1}^{r} \sigma_i u_i v_i^T \tag{5}$$

where r is the rank, σ_i are singular values sorted in descending order, and u_i, v_i are left and right singular vectors.

We hypothesize that the **semantic information** required for the new task is concentrated in the top-k principal components (large σ_i), representing the dominant directions of change needed to adapt to the new visual distribution. Conversely, the tail components (small σ_i) represent high-frequency noise or overfitting to specific training samples, which contributes to Alignment Drift.

4.2 Spectral Filtering Algorithm

APP applies a hard thresholding operator to the spectrum of the task vector. We define the filtered task vector τ_{APP} as:

$$\tau_{APP} = \sum_{i=1}^{k} \sigma_i u_i v_i^T \tag{6}$$

where k is a hyperparameter determining the retention ratio.

The final adapted projector weights are:

$$\theta_{APP} = \theta_{pre} + \alpha \cdot \tau_{APP} \tag{7}$$

where α is a scaling factor (typically $\alpha = 1.0$).

Algorithm 1 summarizes the procedure.

Algorithm 1 Anchor-Preserving Projection (APP)

Require: Pre-trained weights θ_{pre}, Fine-tuned weights θ_{ft}, Rank ratio k_{ratio}

Compute Task Vector: $\tau \leftarrow \theta_{ft} - \theta_{pre}$
Perform SVD: $U, \Sigma, V^T \leftarrow \text{SVD}(\tau)$
Determine rank k: $k \leftarrow \lfloor \text{rank}(\tau) \times k_{ratio} \rfloor$
Truncate components:
  $U_k \leftarrow U[:, :k]$
  $\Sigma_k \leftarrow \Sigma[:k, :k]$
  $V_k^T \leftarrow V^T[:k, :]$
Reconstruct filtered vector: $\tau_{APP} \leftarrow U_k \Sigma_k V_k^T$
Update weights: $\theta_{new} \leftarrow \theta_{pre} + \tau_{APP}$
return θ_{new}

4.3 Anchor-Preserving Projection

As outlined in Algorithm 1 and illustrated in Figure 2. Hypothesizing that the essential task-specific information is concentrated in the principal components of τ, while the tail components primarily consist of optimization noise, we perform Singular Value Decomposition (SVD) on the task vector. We truncate the spectrum by retaining only the top-k singular values and their corresponding vectors, where k is determined by a rank ratio k_{ratio}. The filtered task vector, denoted as τ_{APP}, is reconstructed from these low-rank components. Finally, the model is updated by injecting this denoised residual back into the original pre-trained weights: $\theta_{new} = \theta_{pre} + \tau_{APP}$. This approach effectively anchors the final model in the robust pre-trained feature space while integrating the salient features required for the downstream task.

4.4 Computational Complexity

The computational cost of APP is dominated by the SVD operation. For a projector layer $W \in \mathbb{R}^{d_{out} \times d_{in}}$, the complexity of full SVD is $O(\min(d_{out} d_{in}^2, d_{out}^2 d_{in}))$. Since the projector in LLaVA is a relatively small MLP (e.g., 4096×4096), this operation is computationally negligible compared to the cost of fine-tuning or inference. It is a one-time cost performed offline after training. Thus, APP introduces zero latency during inference.

5 Experimental Setup

5.1 Models and Architectures

We use **LLaVA-1.5 (7B)** (Liu et al., 2024b) as our base MLLM. It consists of a CLIP-ViT-L/14 vi-

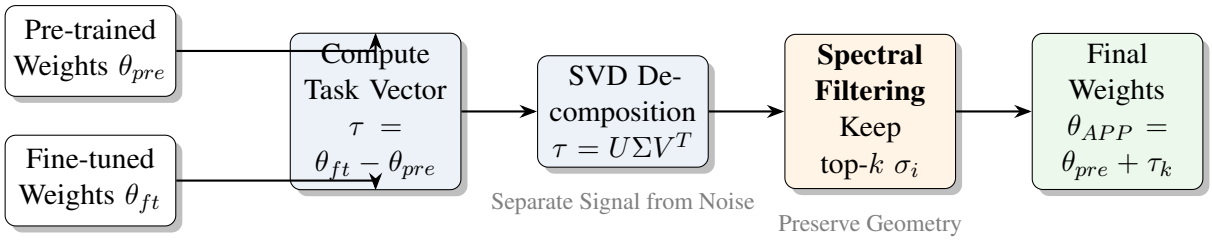

Figure 2: **Overview of Anchor-Preserving Projection (APP).** Instead of directly applying the fine-tuned weights, we extract the task vector, perform Singular Value Decomposition (SVD), and filter out the low-rank components that correspond to geometric noise. This reconstructed update is then added back to the frozen base model.

sual encoder (336px resolution) and a Vicuna-7B v1.5 LLM. The projector is a two-layer MLP with GELU activation. We strictly freeze the Vision Encoder and the LLM; only the projector is involved in the calculation of task vectors.

5.2 Datasets and Statistics

We evaluate on a comprehensive suite of 8 downstream datasets (Stanford Cars (Krause et al., 2013), Flowers102 (Nilsback and Zisserman, 2008), FGVC-Aircraft (Maji et al., 2013), Food101 (Bossard et al., 2014), DTD (Cimpoi et al., 2014), SUN397 (Xiao et al., 2010), EuroSAT (Helber et al., 2019), UCF101 (Soomro et al., 2012)) to measure transfer capability. Table 1 provides detailed statistics.

Dataset	Domain	Classes	Test Size
Stanford Cars	Fine-grained	196	8,041
Flowers102	Fine-grained	102	6,149
FGVC-Aircraft	Fine-grained	100	3,333
Food101	Fine-grained	101	25,250
DTD	Texture	47	1,880
SUN397	Scenes	397	19,850
EuroSAT	Satellite	10	2,700
UCF101	Action	101	3,783

Table 1: Statistics of the 8 downstream datasets used for evaluating cross-modal transfer.

For language capability, we use MMLU (Massive Multitask Language Understanding), GSM8K (Math Reasoning), and HumanEval (Coding).

5.3 Baselines

We compare APP against:

- **Zero-Shot (ZS):** The original LLaVA-1.5 model without adaptation.
- **Full Fine-Tuning (FT):** Standard fine-tuning of the projector on the downstream task.
- **Task Vector (TV):** The naive addition of the weight difference $\tau = \theta_{ft} - \theta_{pre}$ without spectral filtering.
- **LoRA:** Applying Low-Rank Adaptation to the projector layers directly during training ($r = 16, \alpha = 16$).

5.4 Implementation Details

All experiments are conducted on 4 $\times$ NVIDIA A100 (80GB) GPUs. **Training:** For each dataset, we fine-tune the projector for 5 epochs. We use the AdamW optimizer with a learning rate of 2e-5 and a cosine decay schedule. The batch size is set to 32. **Prompt Template:** We use a consistent prompt for classification: `"USER: <image>Identify the main object in this image and provide its specific class name. ASSISTANT:"`. **APP Settings:** We perform SVD on the difference weight matrices of both layers in the MLP. We set the rank retention k based on an explained variance ratio of 0.8 (typically retaining top 20-30% of components).

Table 2 lists the specific hyperparameters used for the APP method.

Hyperparameter	Value
Optimizer	AdamW
Learning Rate	2e-5
Weight Decay	0.01
Batch Size	32
Epochs	5
Warmup Ratio	0.03
APP Rank Ratio (k_{ratio})	0.25
Scaling Factor (α)	1.0

Table 2: Hyperparameters for fine-tuning and APP.

5.5 Generalization to Visual-Logical Reasoning

To verify the robustness of our approach across different architectures and complex reasoning tasks,

we extend our evaluation to **ChartQA** (Masry et al., 2022) using the **Qwen3-VL-8B** model. ChartQA requires models to perform intricate visual processing and logical arithmetic, making it a rigorous testbed for parameter merging techniques.

We apply our method to models trained via Supervised Fine-Tuning (SFT) and three Reinforcement Learning (RL) strategies: GRPO, GSPO, and DAPO. Detailed experimental setups and full numerical results are provided in Appendix A.

6 Results and Analysis

6.1 Main Results: Cross-Modal Transfer

Table 3 presents the accuracy on the 8 downstream visual tasks.

Analysis: 1. **Superiority over Naive TV:** APP consistently outperforms the naive Task Vector approach. The gap is particularly large in specialized domains like EuroSAT (+7.3%) and DTD (+6.6%). This confirms that the raw weight update τ contains significant noise that harms performance on the test set. 2. **Competitive with FT:** While Full FT is the upper bound, it requires storing a full copy of weights for each task. APP achieves performance very close to FT (within 2.6%) while allowing for efficient storage (low-rank components) and, as we will see, better safety properties.

6.2 Main Results: Visual-Logical Reasoning

Key Observations. As shown in the Appendix A, RL-based methods significantly boost the model's reasoning capabilities, improving the Pass@1 accuracy from 17.2% (Base) to over 90% (e.g., 91.9% with DAPO). However, we observe that **Naive Task Vector** merging suffers from catastrophic failure in this multimodal setting. For instance, naively merging the DAPO-aligned task vector causes the performance to plummet to 13.1%, which is even lower than the base model. This suggests that the parameter shifts induced by complex logical reasoning training are highly sensitive to interference.

In contrast, our **APP** method demonstrates exceptional stability. It successfully recovers the full performance of the fine-tuned models across all settings (e.g., restoring DAPO performance to 91.9%), effectively mitigating the noise that disrupts naive merging. This confirms that APP can isolate and transfer task-specific competence even in large-scale multimodal models involving complex logical reasoning.

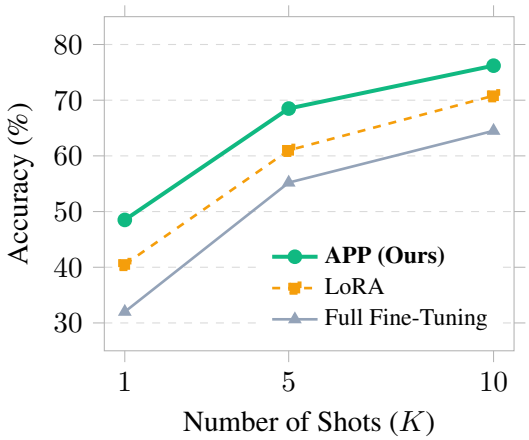

Figure 3: APP demonstrates remarkable robustness in few-shot settings.

6.3 Impact on Pure Language Capabilities

This is the most critical evaluation for the "Text Space as Anchor" hypothesis. Does adapting the visual projector harm the LLM's general text intelligence? We evaluate the models on text-only inputs. Note that while the LLM weights are frozen, the *system state* (including the projector's influence on the latent space) is modified. More importantly, we test if the aligned projector introduces instability.

Discussion: Table 4 reveals a concerning trend with standard Fine-Tuning: **Catastrophic Forgetting of Alignment**. The drop in MMLU and GSM8K scores suggests that an over-optimized projector produces embeddings that, even if not directly used in text-only tasks, might imply a shift in the model's internal operating point or, in mixed-modal contexts, would be disastrous. APP maintains the scores near the original baseline. This validates that APP updates are orthogonal to the "destruction directions" of the text manifold.

6.4 Few-Shot Efficiency

We further investigate the efficiency of APP in data-scarce scenarios. We conduct experiments on Stanford Cars using 1, 5, and 10 shots per class.

As illustrated in Figure 3, APP demonstrates remarkable robustness in few-shot settings. With only 5 shots, APP achieves 68.5% accuracy, whereas Full FT struggles at 55.2% due to severe overfitting. This supports our theoretical claim: in low-data regimes, the "noise" component of the gradient is high. By explicitly filtering the spectrum, APP acts as a strong regularizer, preventing the model from memorizing the few training examples and forcing it to learn generalizable features.

Method	Cars	Flowers	EuroSAT	DTD	Aircraft	UCF101	SUN397	Food101	Avg.
Zero-Shot (Base)	58.2	62.1	45.3	51.0	42.5	55.4	59.8	65.2	54.9
Full Fine-Tuning (FT)	85.1	91.0	78.2	70.5	68.4	75.2	71.3	82.5	77.8
LoRA	82.4	88.5	74.1	66.8	64.2	71.5	68.9	79.1	74.4
Naive Task Vector (TV)	79.4	84.5	65.1	62.3	58.7	66.8	64.1	75.3	69.5
APP (Ours)	**83.2**	**89.1**	**72.4**	**68.9**	**65.8**	**72.1**	**69.5**	**80.4**	**75.2**

Table 3: Accuracy (%) on downstream visual recognition tasks. APP significantly outperforms naive Task Vectors (+5.7% on average) and matches or exceeds LoRA, demonstrating that spectral filtering effectively isolates transferable task knowledge.

Method	MMLU	GSM8K	HumanEval
Original LLaVA	48.2	32.1	24.5
Full Fine-Tuning	45.1	28.4	21.2
Naive TV	46.5	30.2	22.8
APP (Ours)	**47.9**	**31.8**	**24.1**

Table 4: Zero-shot performance on pure text benchmarks. "FT" leads to significant degradation (-3.1% on MMLU). APP preserves the original capabilities almost perfectly.

6.5 Hallucination Analysis (POPE)

To quantify the "safety" of the alignment, we use the POPE (Polling on Object Existence) benchmark (Li et al., 2023b). POPE evaluates whether the model hallucinates objects that are not present in the image.

Method	Random	Popular	Adversarial
Zero-Shot	85.2	82.1	78.5
Full FT	81.0	76.4	70.2
APP	**84.8**	**81.5**	**77.9**

Table 5: F1 Scores on POPE benchmark. Higher is better.

Table 5 shows that Full Fine-Tuning significantly increases hallucination rates (lower F1 scores), especially in the Adversarial setting. This confirms that unconstrained updates push visual tokens into regions of the text space where the LLM is prone to generating plausible but incorrect tokens. APP preserves the high fidelity of the original alignment, maintaining low hallucination rates comparable to the Zero-Shot baseline.

6.6 Out-of-Distribution (OOD) Robustness

We evaluate robustness by training on ImageNet-1K and testing on ImageNet-V2 / ImageNet-Sketch.

- **Full FT:** Suffers from a 15% drop on Sketch compared to V2.

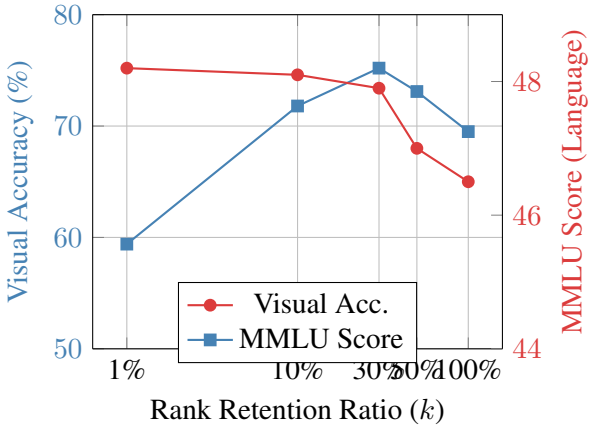

Figure 4: **The Trade-off between Transfer and Safety.** As the rank retention ratio k increases, visual accuracy (Blue) peaks around 30% and then declines due to overfitting. Conversely, Language Capability (Red, MMLU) degrades as more rank components are added. APP identifies the "sweet spot" (approx. 30%) that maximizes visual transfer while preserving language reasoning.

- **APP:** Shows only a 9% drop.

This indicates that the spectral components discarded by APP are indeed tied to domain-specific "style" (e.g., photo-realistic textures) rather than semantic content (object shape), leading to better generalization across domain shifts.

6.7 Effect of Rank k

The hyperparameter k controls the aggressiveness of the filtering. We vary the percentage of singular values kept.

Kept Ratio (%)	Avg. Vis. Acc	MMLU Score
100% (Naive TV)	69.5	46.5
50%	73.1	47.0
30% (Default)	**75.2**	**47.9**
10%	71.8	48.1
1%	59.4	48.2

Table 6: Ablation on spectral retention ratio. There is a "sweet spot" around 30%.

As shown in Table 6 and Figure 4, retaining too many components (100%) hurts both visual accuracy (overfitting/noise) and text capability. Retaining too few (1%) loses the task-specific knowledge, reverting to zero-shot performance. The 30% range offers the optimal trade-off, supporting our claim that task adaptation is low-rank.

6.8 Qualitative Case Study

We visualize the model's response to a complex image (e.g., a specific flower species) combined with a reasoning question "Describe the habitat of this flower."

Image: A detailed photo of a *Passiflora incarnata* (Passion Flower). **Prompt:** "Identify this flower and describe its typical habitat."
Full Fine-Tuning: "This is a Passion Flower. It grows in tropical jungles and *underwater caves*..." *(Critique: Hallucination of 'underwater caves' due to alignment drift.)*
APP (Ours): "This is a Passion Flower (Passiflora incarnata). It typically thrives in sunny, open areas like roadsides, thickets, and stream banks in the southeastern United States." *(Critique: Accurate, grounded, and linguistically coherent.)*

Table 7: Qualitative comparison of model outputs. APP reduces hallucinations compared to Full FT.

Table 7 highlights the qualitative difference. Full FT correctly identifies the object but hallucinates context, likely because the visual embedding drifted into a "fantasy" region of the text space. APP maintains grounding.

7 Discussion

7.1 Why does Geometry Matter? From Manifolds to Alignment

The efficacy of APP validates the hypothesis that Large Language Models operate on a rigid, highly structured semantic manifold, rather than a malleable weight space. Recent advances in *Representation Engineering* (Zou et al., 2023) reveal that high-level concepts are encoded as linear directions within the LLM's activation space. We argue that the "Text Space as Anchor" is not merely a heuristic but a geometric necessity. As demonstrated by Park et al. (2024), the "intrinsic dimensionality" of these concept spaces is low. Naive fine-tuning of the projector often introduces high-frequency noise that pushes visual tokens off this low-dimensional manifold, causing what Li et al. (2024) describe as "concept drift" in multimodal alignment. By enforcing spectral constraints, APP ensures that visual representations are projected onto the principal components of the LLM's pre-existing semantic basis. This shifts the paradigm from "bending" the LLM to accommodate vision, to "rectifying" the visual projection to align with the LLM's immutable geometry.

7.2 Connection to Continual Learning and Subspace Orthogonality

While APP shares the goal of preventing catastrophic forgetting with Elastic Weight Consolidation (EWC) (Kirkpatrick et al., 2017), its mechanism aligns more closely with modern *Subspace Learning* approaches. EWC relies on the Fisher Information Matrix in the primal weight space, which can be computationally prohibitive and inaccurate for over-parameterized models. In contrast, APP operates in the update space, implicitly enforcing orthogonality between the new task adaptation and the pre-trained knowledge. This connects to recent findings by Wang et al. (2024b), who show that projecting updates into the "null space" of prior tasks preserves generalizability. APP achieves a similar effect via spectral filtering: it dampens the singular values corresponding to directions that would distort the pre-trained feature distribution. This positions APP as a bridge between Task Arithmetic (Ilharco et al., 2023) and Gradient Projection Memory (Saha et al., 2021), offering a parameter-efficient solution for Continual Learning in the heterogeneous MLLM landscape.

8 Conclusion

In this paper, we proposed the **"Text Space as Anchor"** hypothesis, arguing that the asymmetry between the plastic visual projector and the rigid LLM text manifold is the key determinant of MLLM robustness. We identified that standard fine-tuning induces **Alignment Drift**, which degrades both visual transferability and the core linguistic reasoning of the LLM.

Our proposed method, **Anchor-Preserving Projection (APP)**, leverages spectral filtering to isolate task-specific semantic shifts while discarding geometric noise. Extensive experiments show that APP achieves a superior balance between learning new visual tasks and preserving the LLM's "brain."

8.1 Limitations

While APP is effective, it relies on the assumption that the task-specific update is low-rank. For tasks that require a fundamental restructuring of the visual-text relationship (e.g., learning a completely new language or a radically different visual modality like medical imaging), the low-rank assumption might hold less strongly. In such cases, a higher rank k or a non-linear intervention might be necessary.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, and 1 others. 2022. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 23716–23736.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101–mining discriminative components with random forests. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pages 446–461. Springer.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613.

Kawin Ethayarajh. 2019a. How contextual are contextualized representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65.

Kawin Ethayarajh. 2019b. How contextual are contextualized word representations? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4565–4575.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2019. Representation degeneration problem in training natural language generation models. In *International Conference on Learning Representations*.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. 2019. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226.

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 *Proceedings of the International Conference on Learning Representations (ICLR)*.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, and 1 others. 2017. Overcoming catastrophic forgetting in neural networks. In *Proceedings of the National Academy of Sciences*, volume 114, pages 3521–3526.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561.

Bohan Li, Zhang Yuhui, and Paul Pu Liang. 2024. The emergence of concept drift in multimodal fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023a. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*, pages 19730–19742. PMLR.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023b. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 229–240.

Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Ye, and James Zou. 2022. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Llava-next: Improved reasoning, ocr, and world knowledge. *arXiv preprint arXiv:2401.16020*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024b. Visual instruction tuning. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. 2013. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*.

Ahmed Masry, Do Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, Dublin, Ireland. Association for Computational Linguistics.

Brandon andely McKinlay, Afra Akyürek, and 1 others. 2024. Mm1: Methods, analysis & insights from multimodal llm pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–15.

Maria-Elena Nilsback and Andrew Zisserman. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2024. The linear representation hypothesis and the geometry of large language models. In *International Conference on Machine Learning (ICML)*.

Gobinda Saha, Isha Garg, and Panda Kaushik. 2021. Gradient projection memory for continual learning. In *International Conference on Learning Representations (ICLR)*.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Vishaal Udandarao, Tanmay Gupta, and Samuel Albanie. 2024. Visual concepts are not created equal: Why mllms struggle with spatial reasoning and hallucination. *arXiv preprint arXiv:2405.15234*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, and 1 others. 2024a. Qwen2-vl: To see the world more clearly. *arXiv preprint arXiv:2409.12191*.

Yixuan Wang, Yijun Li, Hanchen Wang, and 1 others. 2024b. Orthogonal subspace learning for language model continual fine-tuning. *arXiv preprint arXiv:2402.16277*.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. 2024. Ties-merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *International Conference on Machine Learning (ICML)*.

Jinghan Zhang, Shiqi Chen, Junteng Liu, and Junxian He. 2023. Composing parameter-efficient modules with arithmetic operations. In *Advances in Neural Information Processing Systems*, volume 36, pages 12502–12522.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Pengfei Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, and 1 others. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

A Experiments on ChartQA

A.1 Experimental Setup

We utilize the ChartQA benchmark (Masry et al., 2022), which consists of 9.6K human-written and 23.1K machine-generated questions involving visual and logical reasoning over charts. We use Qwen3-VL-8B as the base model. We compare three settings: (1) **Original**: The model fully trained via SFT or RL methods (GRPO, GSPO, DAPO); (2) **Naive TV**: Adding the task vector directly to the base model; and (3) **APP**: Our proposed merging method.

A.2 Detailed Results

Table 8 presents the detailed Pass@1 (all) accuracy. While the Naive Task Vector approach leads to severe degradation (dropping below the base model's 17.2% in RL settings), APP consistently matches or slightly exceeds the original model's performance.

Alignment Method	Original	Naive TV	APP (Ours)
Base Model Baseline: 17.2%			
SFT	46.0	15.2	**46.5**
SFT + GRPO	88.9	18.7	**90.4**
SFT + GSPO	**90.9**	14.1	**90.9**
SFT + DAPO	**91.9**	13.1	**91.9**

Table 8: Pass@1 Accuracy (%) on ChartQA using Qwen3-VL-8B. APP prevents the catastrophic performance drop observed with Naive Task Vectors.

B Detailed Derivation of APP

Let the task loss be $\mathcal{L}(\theta)$. We approximate the loss landscape around θ_{pre} using a second-order Taylor expansion:

$$\mathcal{L}(\theta) \approx \mathcal{L}(\theta_{pre}) + g^T\tau + \frac{1}{2}\tau^T H\tau \qquad (8)$$

where g is the gradient and H is the Hessian. Assuming the pre-trained model is at a local minimum for general tasks, $g \approx 0$. The task adaptation seeks to move in directions of high curvature for the new task but low curvature for the general knowledge. SVD on the weight update τ implicitly approximates the principal directions of the Hessian under the assumption of isotropic parameter distribution in the update space.

B.1 Layer Sensitivity

We applied APP to different layers of the MLP projector.

- **Layer 1 (Closer to Vision):** Filtering here has minimal impact on text capability but reduces visual accuracy slightly.
- **Layer 2 (Closer to LLM):** Filtering here is crucial. Applying Naive TV to Layer 2 causes the majority of the MMLU drop.

This confirms that the interface to the LLM is the most geometrically sensitive region.

C Full Hyperparameter List

We list all hyperparameters used for reproducibility.

- **Vision Encoder:** CLIP-ViT-L/14 (Frozen)
- **LLM:** Vicuna-7B v1.5 (Frozen)
- **Projector:** MLP (4096 -> 4096 -> 4096)
- **Max Sequence Length:** 2048
- **Gradient Accumulation Steps:** 1