HALoRA: Low-Rank Adaptation with Hierarchical Budget Allocation for Efficient Vision-Language Alignment

Letian Zhang[1*], **GuangHao Meng**[1*], **Xudong Ren**[1], **Jinpeng Wang**[2†]
[1]Tsinghua Shenzhen International Graduate School, Tsinghua University
[2]Harbin Institute of Technology, Shenzhen
{zlt23, menggh22, rxd21, wjp20}@mails.tsinghua.edu.cn

Abstract

With the emergence of large multimodal models, dual-encoder alignment via contrastive learning has seen a resurgence. However, the escalating model size demands effective Parameter-Efficient Fine-Tuning (PEFT). While LoRA is a promising inference-free alternative to adapters, we find that its naive application to multimodal tasks causes a severe rank imbalance, favoring the text modality and FFN layers. To address this, we propose HALoRA (**H**ierarchical **A**llocation LoRA), which introduces a component-wise budget allocator to ensure balanced fine-tuning across both modalities and their internal components. This is complemented by a gradient-approximated initialization to accelerate convergence. With only half the parameters of adapters, HALoRA achieves superior or competitive performance in retrieval and zero-shot classification. Our work presents a more principled approach to multimodal LoRA, uncovering an intriguing asymmetry in vision-language alignment.

Introduction

Modal alignment through contrastive learning (Radford et al. 2021; Wang et al. 2022a; Zhao et al. 2023; Wang et al. 2024a,c; Li et al. 2025a; Zhang et al. 2025) has been a prominent area of focus in multimodal research. This dual-encoder paradigm proved foundational, inspiring a wave of subsequent models (Jia et al. 2021; Yu et al. 2022) that push the state-of-the-art by scaling up architectures and web-scale data. With expanding parameters and data, Parameter-Efficient Fine-Tuning (PEFT) has gained attention for its cost-effectiveness and lower risk of catastrophic forgetting, especially in data-limited scenarios, making it a practical alternative to fully fine-tuning. PEFT approaches include adapter (Houlsby et al. 2019; Rebuffi, Bilen, and Vedaldi 2017), prompt tuning (Li and Liang 2021; Liu et al. 2021), and Low-Rank Adaptation (LoRA) (Hu et al. 2022; Liang et al. 2025). While adapters are widely used in multimodal alignment, they introduce additional inference latency. LoRA, with its ability to be merged into the original weights, offers an inference-free advantage, making it a highly attractive option.

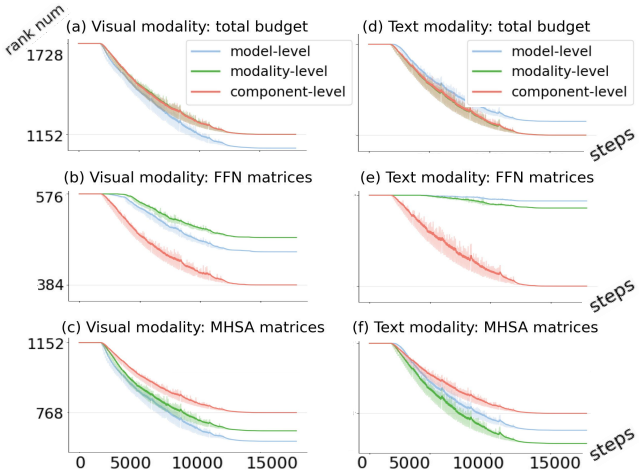

Figure 1: Illustration of hierarchical rank imbalance in adaptive LoRA. The baseline model-level approach (blue line)—using a single, shared pool of ranks—creates a severe skew, favoring the text modality (d vs. a) and FFN layers (e vs. f). In contrast, our component-level approach (red line)—which allocates ranks to each sub-layer (MHSA/FFN) independently—rectifies this imbalance. The modality-level scheme (green line), assigning separate rank pools per modality, represents an intermediate solution.

However, LoRA's success in unimodal NLP does not guarantee similar performance in vision-language alignment. Our investigation, visualized in Figure 1 (blue line), reveals a critical, previously overlooked problem: a severe rank imbalance emerges when naively applying adaptive LoRA to dual-encoder architectures. We identify two levels of this phenomenon: a cross-modal imbalance, where the text encoder monopolizes the budget at the expense of the vision encoder, and an intra-modal imbalance, where FFN layers are favored over MHSA modules within each encoder. This unchecked, hierarchical imbalance hobbles the fine-tuning process, leading to unstable training and suboptimal performance as crucial components are "starved" of adaptive capacity. We hypothesize that this occurs because naive optimizers disproportionately assign credit to the final, more task-specific layers (like FFNs) and the more flexible modal-

[*]These authors contributed equally.
[†]Corresponding author.
Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

ity (text), fundamentally misunderstanding the distributed nature of knowledge within dual-encoder architectures.

To rectify this fundamental imbalance, we introduce HALoRA (Hierarchical Allocation LoRA), a principled adaptive framework designed specifically for dual-encoder alignment. Our approach is twofold. First, to directly counteract the resource monopolization, we introduce a more granular, structure-aware control mechanism. Instead of a single, shared rank pool, it enforces balanced adaptation by managing resources independently for each modality (vision and text) and for each internal component type (MHSA and FFN). Second, recognizing that such fine-grained adaptation can initially slow down convergence, we complement our allocator with a gradient-aware initialization. This provides a much stronger starting point for optimization by aligning the initial update direction with that of full fine-tuning, significantly accelerating convergence and improving stability.

Extensive experiments demonstrate that HALoRA, despite using only half the trainable parameters of adapter-based methods, achieves superior or competitive performance on various retrieval and classification benchmarks Intriguingly, our analysis of the final rank distribution uncovers a consistent asymmetry between modalities, offering novelty insights into their interplay.

Our contributions can be summarized as follows:

- We identify and analyze a critical, previously overlooked hierarchical rank imbalance when applying adaptive LoRA to dual-encoders, where ranks are skewed across modalities and components.
- To resolve this, we propose HALoRA, a principled framework that implements a hierarchical allocation strategy for adaptive LoRA. It enforces balanced fine-tuning by managing resources independently per modality and component, ensuring the model adapt effectively.
- We further introduce a gradient-approximated initialization to accelerate convergence and enhance stability by providing a more effective initial optimization direction, addressing a common pitfall of adaptive methods.
- Extensive experiments show HALoRA achieves superior or competitive performance with fewer parameters. Our analysis also uncovers a consistent rank asymmetry between modalities, opening new avenues for architecturally-aware PEFT design.

Related Work

Aligning Image to Text. The dual-encoder paradigm, pioneered by CLIP (Radford et al. 2021) and ALIGN (Jia et al. 2021), established a foundational approach for aligning visual and language representations. Numerous follow-up works (Wang et al. 2022b, 2023; Meng et al. 2025, 2026a,b; Tang et al. 2025, 2026) have since refined this paradigm with improved training objectives, datasets, and architectural tweaks (Li et al. 2021; Yao et al. 2022). While alternative architectures like fusion encoders (Li, Li et al. 2022; Kim, Son, and Kim 2021) have demonstrated strong performance on certain benchmarks, their inherent need to process modalities jointly makes them unsuitable for latency-sensitive applications like large-scale retrieval (Wang et al. 2024b; Li et al. 2025b). Consequently, the efficient dual-encoder structure remains the cornerstone for many state-of-the-art Large Multimodal Models (LMMs) (Liu et al. 2023, 2024a, 2025; Gao et al. 2025), where it serves as a critical feature alignment stage. This underscores the continued importance of optimizing the foundational contrastive alignment process itself.

Parameter-Efficient Finetuning. PEFT methods are critical for adapting large-scale models and are broadly categorized into adapter tuning (Chen et al. 2022; Sung, Cho et al. 2022), prompt tuning (Zhou et al. 2022; Jia et al. 2022), and parameter tuning (Hu et al. 2022). Adapter-based methods are dominant in multimodal alignment. These approaches range from inserting small network layers directly within encoders, such as in LilT (Khan and Fu 2023), to employing dedicated modules like Q-Former (Li, Li et al. 2022, 2023) or projection layers (Liu et al. 2023, 2024a) to bridge vision encoders with Large Language Models. The ubiquity of adapters is further highlighted by their widespread use in unified modal tasks, where various modalities are projected into a common space (Shukor et al. 2023). In contrast, parameter tuning methods like LoRA are appealing for their zero inference overhead but are less explored for foundational dual-encoder alignment. The few existing works either apply LoRA to downstream LMM instruction tuning, like MixLoRA (Shen et al. 2024), or for few-shot adaptation, as in CLIP-LoRA (Zanella and Ayed 2024). Our work differs fundamentally by investigating a more basic question: how to properly apply adaptive LoRA to the core dual-encoder alignment task itself. We are the first to identify and address the inherent structural imbalance that these prior methods overlook, making the process more principled and effective.

Enhancements to LoRA. LoRA (Hu et al. 2022) approximates weight changes in full fine-tuning using low-rank matrices. Several variants have been proposed to improve it. LoRA+ (Hayou, Ghosh, and Yu 2024) uses different learning rates for LoRA's matrices to enhance convergence. DoRA (Liu et al. 2024b) boosts model expressiveness with learnable magnitudes. AdaLoRA (Zhang et al. 2023) selectively prunes weights during fine-tuning. While these methods refine LoRA, they overlook the structural imbalance that becomes evident in multimodal domains, a problem our work directly addresses by ensuring fair attention to every component. Separately, methods like LoRA-GA (Wang, Yu, and Li 2024) and LoRA-Pro (Wang et al. 2025) align LoRA with full fine-tuning gradients to accelerate convergence. We build upon and adapt these ideas for our novel architecture.

Method

HALoRA achieves balanced and stable fine-tuning of dual-encoders via three core components: Contribution Index (CI) to quantify parameter importance, Hierarchical Rank Allocators that use CI to intelligently distribute ranks, and Gradient-Approximation Initialization for stable convergence. The complete workflow is outlined in Algorithm 1.

Quantifying Parameter Contribution

To enable dynamic rank allocation, we first need a reliable metric to assess the real-time contribution of each trainable

parameter. We achieve this through a two-step process: designing a re-parameterization of LoRA that allows for flexible rank adjustment, and then defining a contribution index to guide this adjustment.

SVD-like LoRA. In traditional LoRA, the increment of the parameter matrix $W_0 \in \mathbb{R}^{n\times m}$ is represented by a low-rank decomposition:

$$W^{'} = W_0 + \Delta W = W_0 + \gamma BA, \tag{1}$$

where $W^{'}, \Delta W \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{m\times r}$, $A \in \mathbb{R}^{r\times n}$. $r \ll \min(m, , n)$, and $\gamma = \frac{\alpha}{r}$ represents the scaling hyperparameter. A limitation of this representation is that, without modifying the matrix structure, adjusting the number of trainable parameters in each weight matrix becomes challenging. Building upon Zhang et al. (2023), we also employ the SVD form to construct our LoRA module. The low-rank decomposition is expressed as follows:

$$W = W_0 + \Delta W = W_0 + \gamma P\Lambda Q. \tag{2}$$

Here $P \in \mathbb{R}^{m\times r}$, $\Lambda \in \mathbb{R}^{r\times r}$, $Q \in \mathbb{R}^{r\times n}$ represent the SVD decomposition of the update ΔW, where P and Q are the left and right singular matrices, and $\Lambda = \{\lambda_i\}_{1\leqslant i\leqslant r}$. In this form, during training, the unnecessary singular values λ_k are masked to 0, and consequently, the corresponding feature vectors $P_{\cdot k}$ and $Q_{k\cdot}$ will not produce gradients in the next gradient propagation. This reduces the rank of the trainable parameter matrix W by one, and pruning the matrix Λ corresponds to a reduction in the rank of the LoRA module.

Contribution Index. We draw on the design of PLATON (Zhang et al. 2022) and utilize sensitivity of parameters to assess the contribution of each parameter to the model. The first-order Taylor expansion of the model's loss quantifies the impact of setting this parameter to zero on the model output, while the momentum update method mitigates fluctuations between mini-batches:

$$\tilde{S}(\theta) \leftarrow m\tilde{S}(\theta) + (1-m)|\theta\,\nabla_\theta\mathcal{L}|. \tag{3}$$

Here $m \in [0, 1)$ is a momentum coefficient.

In addition, we corrected the importance $\tilde{S}(\theta)$ by calculating the matrix's overall fluctuations where the parameters reside, further enhancing numerical stability during training:

$$\tilde{\lambda}(\boldsymbol{\theta}) \leftarrow m\tilde{\lambda}(\boldsymbol{\theta}) + (1-m)\left(1 + \frac{\|S(\boldsymbol{\theta}) - \tilde{S}(\boldsymbol{\theta})\|_F}{\|\tilde{S}(\boldsymbol{\theta})\|_F}\right). \tag{4}$$

The initial value of λ is set to 1, fluctuating around this value during training, thereby adjusting $\tilde{S}(\theta)$.

We compute the corrected sensitivity of parameter θ using the sensitivity of parameters $\tilde{S}(\theta)$ and the correction term $\tilde{\theta}$:

$$\hat{S}(\theta) = \tilde{\lambda}(\boldsymbol{\theta})\tilde{S}(\theta). \tag{5}$$

We focus on the contribution index of each singular value $\lambda_k^{(i)}$ in the SVD-like LoRA matrix, where the superscript $^{(i)}$ represents the parameter matrix indexed by i, and $k \in \{1, 2, \cdots, r\}$. In the low-rank decomposition $P^{(i)}\Lambda^{(i)}Q^{(i)}$, the k-th singular value corresponds to the feature vectors $P^{(i)}_{[:,k]}$ and $Q^{(i)}_{[k,:]}$, which together represent the **C**ontribution **I**ndex (CI) of the singular value $\lambda_k^{(i)}$:

$$CI_k^{(i)} \triangleq \hat{S}(\lambda_k^{(i)}) + \langle\hat{S}(P^{(i)}_{[:,k]})\rangle + \langle\hat{S}(Q^{(i)}_{[k,:]})\rangle, \tag{6}$$

$\langle\hat{S}(\cdot)\rangle$ denotes the average sensitivity of each parameter in the vector.

We adopt the pruning method proposed in Zhang et al. (2023) during training, maintaining a monotonically decreasing singular value budget scheduler $b^{(t)}$. Singular values ranked ranked lower than the $b^{(t)}$ in the contribution index are directly set to 0, while others are updated via standard gradient descent:

$$\lambda_{kk}^{(i)} \leftarrow \begin{cases} \lambda_{kk}^{(i)} - \eta\nabla\mathcal{L}, & CI_k^{(i)} \text{ in top-}b^{(t)} \text{ of } \{CI_k^{(i)}\} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

η is the learning rate, and $\mathcal{L}$ is the loss function. Through this approach, the model enables autonomously select parameters that are more valuable.

Hierarchical Rank Allocation

With the Contribution Index established, the core challenge is how to allocate ranks effectively. A naive, model-level allocation—where a single rank pool is shared globally—proves problematic. Our initial experiments confirmed that in dual-encoder models, this leads to the text encoder monopolizing resources and starving other crucial components. This is not a sign of importance, but an artifact of flawed, direct comparison.

To resolve this, our first logical step was to introduce a modality-level allocator. This design creates two independent rank pools, one for the vision encoder and one for the text encoder, preventing cross-modal interference. By isolating the modalities, we ensure that the learning process within the vision encoder, for example, is no longer suppressed by the typically faster-evolving text encoder. This established a much healthier inter-modal balance.

However, further analysis revealed a persistent intra-modal imbalance, as FFN layers consumed more ranks than their MHSA counterparts. To achieve a truly fine-grained balance, we therefore developed our final component-level allocator. This approach further subdivides each modality's pool, creating four independent allocators for Vision-MHSA, Vision-FFN, Text-MHSA, and Text-FFN. This hierarchical design, which forms the core of HALoRA, ensures that every functionally distinct part of the model receives a fair and dynamically adjusted share of resources.

Gradient-Approximation Initialization

The fine-grained, dynamic nature of our hierarchical allocator, while ensuring balance, introduces a new challenge: potential instability and slower initial convergence compared to static LoRA. To mitigate this, we developed a specialized initialization scheme that provides a stronger, more stable starting point for optimization.

Adapting Gradient-Approximation for an SVD-like Structure. In the original structure of LoRA Equation (1), A is usually initialized with Kaiming initialization, while B is initialized to zero (Hu et al. 2022). LoRA-GA (Wang, Yu, and Li 2024) and LoRA-Pro (Wang et al. 2025) provide solutions to align gradients with that in full fine-tuning.

Building upon their work, we explore analogous initializations under an SVD-like structure. In cases where the initial LoRA matrix is non-zero, the LoRA structure in Equation (2) is reformulated in the form of Equation (8):

$$W^{'} = (W_0 - \gamma P_0 \Lambda_0 Q_0) + \gamma P \Lambda Q. \tag{8}$$

Approximating the gradient of LoRA to that of full fine-tuning means:

$$P_0^\star,\, \Lambda_0^\star,\, Q_0^\star = \underset{P_0,\,\Lambda_0,\,Q_0}{\arg\min} \|\Delta(\gamma P \Lambda Q)_1 - \zeta \Delta W\|_F, \tag{9}$$

ζ is a positive hyperparameter.

According to the Eckart-Young-Mirsky theorem (Eckart and Young 1936; Mirsky 1960), if the SVD decomposition of $\nabla_{W_0}\mathcal{L}$ is given by $\nabla_{W_0}\mathcal{L} = U\Sigma V^\top$ and the rank of LoRA is r, then the following solution is a feasible solution to Equation (9) under the $2r$ low-rank approximation:

$$\begin{aligned} &P_0^\star = \frac{\sqrt{\zeta}}{\alpha\sqrt{\gamma}} U_{\mathbb{a}},\, Q_0^\star = \frac{\sqrt{\zeta}}{\alpha\sqrt{\gamma}} V_{\mathbb{b}}^\top,\, \Lambda_0^\star = \alpha \mathbb{I}_r, \\ &\text{s.t.} |\mathbb{a}| = |\mathbb{b}| = r,\, \mathbb{a} \cup \mathbb{b} = \{i \,|\, 1 \leqslant i \leqslant 2r,\, i \in \mathbb{N}\}, \end{aligned} \tag{10}$$

where $\mathbb{I}_r$ represents the identity matrix of rank r, and α is a hyperparameter used to scale the singular value matrix Λ_0, ensuring numerical stability, and is typically smaller than 1.

Taking into account the intrinsic constraint of the SVD-like LoRA, $P_0^\top P_0 = Q_0 Q_0^\top = \mathbb{I}_r$, the initialization scheme actually implemented in our experiments is as follows:

$$P_0^\star = U_{\mathbb{a}},\, Q_0^\star = V_{\mathbb{b}}^\top, \Lambda_0^\star = \sqrt{\zeta/\gamma}\,\mathbb{I}_r = \alpha^{'}\mathbb{I}_r. \tag{11}$$

By adjusting the merged hyperparameter $\alpha^{'}$, we derive an SVD-like initialization scheme which approximates the full fine-tuning gradient.

Algorithm 1: HALoRA Workflow

Require: Dataset $\mathcal{D}$, model $f(\cdot, W)$, loss function $\mathcal{L}$, training iterations T, scaling factor α, component-wise budget schedulers $\{b_{\text{attn}}^{\text{image}}, b_{\text{ffn}}^{\text{image}}, b_{\text{attn}}^{\text{text}}, b_{\text{ffn}}^{\text{text}}\}_{t=0}^{T}$.
Ensure: Fine-tuned LoRA params $\{P^{(T)}, \Lambda^{(T)}, Q^{(T)}\}$

Sample a mini-batch (x, y) from $\mathcal{D}$
$\hat{y} \leftarrow f(x, W)$ ▷ Init from gradient-approximation
$\ell \leftarrow \mathcal{L}(y, \hat{y})$
for $w^{(i)}$ in W **do**
  $U, \Sigma, V \leftarrow \texttt{svd\_lowrank}(\nabla_{w^{(i)}}\ell)$
  $P^{(i)}, \Lambda^{(i)}, Q^{(i)} \leftarrow U_{[:r,:]}, \alpha\mathbb{I}, V_{[r+1:2r,:]}$
  $W^{(i)} \leftarrow W^{(i)} - P^{(i)}\Lambda^{(i)}Q^{(i)}$
end for
for $t = 1, \cdots, T$ **do** ▷ Allocate rank during training
  Sample a mini-batch (x, y) from $\mathcal{D}$
  $\hat{y} \leftarrow f(x, P, Q, \Lambda)$
  $\ell \leftarrow \mathcal{L}(y, \hat{y})$
  for $w^{(i)}$ in W **do** ▷ Prune with hierarchical budgets
    Compute $CI_k^{(i)}$ by Equation (3) $\sim$ (6)
    Update $P^{(i)}$, $Q^{(i)}$ via gradient descend
    Update $\Lambda^{(i)}$ by Equation (7) based on $b^{(t)}$
  end for
end for
return $P^{(T)}$, $\Lambda^{(T)}$, $Q^{(T)}$

Experiments

Settings

Dataset. We investigate multimodal alignment on the COCO2014 dataset (Lin et al. 2014), which has $\sim$ 118K unique images, following the split defined in Karpathy and Fei-Fei (2017). Each image is associated with an average of five captions, resulting in $\sim$591K text-image pairs. This follows the experimental settings of LilT (Khan and Fu 2023), where we conduct full fine-tuning and parameter-efficient fine-tuning on a relatively small yet high-quality training set to compare alignment performance.

Training. The vision encoders' weights are initialized from DeiT (Touvron et al. 2021; Touvron, Cord, and Jégou 2022), and the text encoders' weights are initialized from SimCSE (Gao, Yao, and Chen 2021). We use a mini-batch size of 512 in our training configuration. The optimizer used is AdamW (Loshchilov and Hutter 2019), with a weight decay of 0.02. The learning rate is warmed up to 1e-4 over the first 1,000 steps, then decayed to 1e-5 using a cosine scheduler. These settings are generally consistent with ALBEF (Li et al. 2021) and LilT (Khan and Fu 2023). The training process takes approximately 5 hours on four NVIDIA A100 40GB GPUs.

Tasks. The effectiveness of model alignment is evaluated using two types of tasks: image-text retrieval and zero-shot natural-language-guided image classification (Radford et al. 2021). For image-text retrieval, since the training data includes the COCO validation set, we evaluate on the Flickr30k (Young et al. 2014; Plummer et al. 2017) dataset, which contains 1000 images and 5000 captions. The reported metrics include recall rates for both image-to-text retrieval and text-to-image retrieval. For zero-shot classification, we selected a set of representative image classification datasets, including ImageNet-V2 (Recht et al. 2019), ImageNet-A (Hendrycks et al. 2019), CIFAR100 (Krizhevsky, Hinton et al. 2009), UCF101 (Soomro, Zamir, and Shah 2012), SUN397 (Xiao et al. 2010), Aircraft (Maji et al. 2013), EuroSAT (Helber et al. 2019), StanfordCars (Krause et al. 2013), Food101 (Bossard, Guillaumin, and Gool 2014), OxfordPets (Parkhi et al. 2012), Flower102 (Nilsback and Zisserman 2008), Caltech101 (Fei-Fei, Fergus, and Perona 2007) and DTD (Cimpoi et al. 2014), following the setup in Zhou et al. (2022).

Baselines. We selected various parameter-tuning strategies as baselines, including fully fine-tuned method (Radford et al. 2021), single-model fine-tuned LiT (Zhai et al. 2022), and LilT (Khan and Fu 2023), which uses adapters for parameter-efficient fine-tuning. We reproduced these approaches under our COCO training setting, reporting their results. Additionally, our approach drew inspiration from AdaLoRA (Zhang et al. 2023) and LoRA-GA (Wang, Yu, and Li 2024), whose methods we also replicated and incorporated into our baselines.

Method	% Trained	Flickr30k				ImageNet V2		ImageNet-A		CIFAR100		UCF101	
		TR@1	IR@1	TR@5	IR@5	Acc-1	Acc-5	Acc-1	Acc-5	Acc-1	Acc-5	Acc-1	Acc-5
Full FT	100.0%	56.1	44.3	81.7	72.0	13.5	30.6	5.48	18.1	27.3	52.9	23.9	46.0
LiT	56.01%	44.1	29.6	72.1	59.9	**15.1**	31.3	6.22	19.7	29.3	54.1	25.3	48.1
LilT_{DA}	7.51%	47.6	34.5	74.1	64.9	13.8	30.9	6.39	20.1	32.7	59.8	25.9	49.1
LilT_{LwA}	7.01%	**56.8**	41.7	81.1	70.7	13.4	30.5	6.25	19.6	32.3	**60.4**	24.0	48.4
LoRA	2.56%	54.7	40.9	81.5	70.2	12.8	29.7	5.99	19.6	30.0	55.2	26.1	49.0
AdaLoRA	4.22%	52.8	37.9	79.1	68.5	13.3	31.2	6.44	20.7	31.6	57.5	25.7	47.8
LoRA-GA	2.56%	56.4	**42.0**	**82.2**	70.6	12.8	29.4	5.85	18.9	30.1	56.3	25.0	48.6
HALoRA	4.22%	**56.8**	**42.0**	81.9	**71.0**	14.0	**31.5**	**6.93**	**20.9**	**32.8**	60.2	**27.2**	**49.4**

Table 1: Performance of multimodal alignment in retrieval and classification. All methods were trained for 15 epochs using identical hyperparameter configurations. *Flickr30k*: The retrieval accuracy of the model was tested on the Flickr30k test set, with TR and IR representing image-to-text and text-to-image tasks, respectively. *ImageNet V2, ImageNet-A, CIFAR100, UCF101*: The model's zero-shot natural-language guided classification performance was tested with Acc-1/5 accuracy. Highest value is highlighted in **bold**, and the second highest is underlined (over 3 trials).

Main Results

In Table 1, we compare HALoRA with other methods trained using the CLIP-based contrastive learning on Flickr30k for retrieval, ImageNet V2, ImageNet-A and CIFAR100 for general classification, and UCF for action recognition. The textual descriptions for zero-shot classification are consistent with CLIP (Radford et al. 2021). The same training dataset and hyperparameters were employed across all methods.

Table 1 shows that the standard LoRA method, lacking adaptive training budget allocation, exhibits significantly worse than the adapter-based method (Table 1, "LilT"). While the improved LoRA-GA performs well on retrieval tasks, it yields limited improvements for classification. On the other hand, the straightforward adaptive rank allocation strategy employed by AdaLoRA is not well-suited for retrieval tasks. Our improvements outperform all these LoRA-based baselines and remain highly competitive when compared to all fine-tuning baselines in the rankings.

Ablation Study

Following the evaluation of basic accuracy, we further performed ablation studies on the critical components of HALoRA to elucidate their respective contributions to the model's capabilities.

Effectiveness of Hierarchical Allocator. As visually demonstrated in the Introduction's Figure 1, our different allocator designs lead to distinct rank budget distributions. Table 2 quantifies how these distributional differences directly impact final model performance.

We first revisit the allocation dynamics shown in Figure 1 (b, c, e, f), which detail the budget split between MHSA and FFN layers. Although the modality-level allocator (green line) resolves the cross-modal imbalance (Figure 1a vs. 1d), it fails to address the intra-modal skew, where FFN layers still seize the majority of the budget. Ideally, the MHSA budget should be double the FFN budget, as attention sublayers contain four matrices (W_q, W_k, W_v, and W_o) compared to two in the linear sublayers (W_{f_1} for upsampling and W_{f_2} for downsampling). Our component-level allocator (red line) successfully enforces this more balanced and theoretically-grounded allocation.

Method	TR@1	IR@1	TR@5	IR@5
model-level	55.6	40.9	80.7	70.3
modality-level	54.9	41.0	81.2	70.6
component-level	**56.8**	**42.0**	**81.9**	**71.0**

Method	Acc^{V2}	Acc^{A}	$\text{Acc}^{\text{cifar}}$	Acc^{ucf}
model-level	13.4	6.65	32.2	26.4
modality-level	13.6	6.67	32.0	26.5
component-level	**14.0**	**6.93**	**32.8**	**27.2**

Table 2: Ablation study on different parameter allocator levels. Performance comparison of the model-level, modality-level, and component-level allocators. The top sub-table reports retrieval results on Flickr30k, while the bottom reports zero-shot classification accuracy (Acc-1) on four datasets. Bold indicates the best performance.

The performance results in Table 2 confirm our hypothesis. The component-level approach, by ensuring a fair distribution of trainable parameters to all functional components, consistently and significantly outperforms the other two methods across all evaluated retrieval and classification tasks. This aligns with findings from studies like Geva et al. (2021), which show that MHSA and FFN serve distinct functions and thus benefit from separate treatment.

Effectiveness of Gradient-Approximating Initialization. Figure 2 and Table 3 illustrates the improvements achieved by approximating full fine-tuning gradients with a specialized initialization compared to general initialization (following the design of Zhang et al. (2023), where general initialization refers to normal initialization for P and Q, and zero initialization for Λ). Gradient-approximating initialization facilitates faster convergence, with particularly pronounced effects in retrieval tasks. The only additional cost is computing a single step of full fine-tuning, which is entirely acceptable even with CPU resources. This initialization is crucial for our SVD-like structure—where standard zero-initialization is inapplicable—and provides a principled starting point to bridge the gap between our adaptive method and the convergence speed of full fine-tuning.

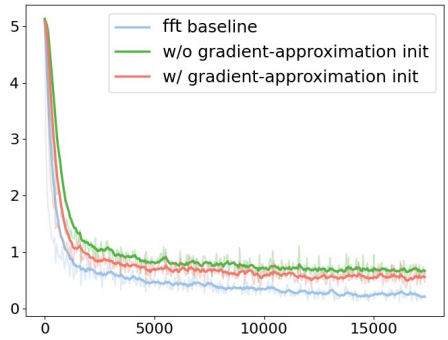

Figure 2: HALoRA initialization with *vs.* without gradient-approximation initialization. Training loss curves under gradient approximation and standard initialization. Under gradient approximation, the model loss decreases more rapidly, with the curve closely approaching full fine-tuning. The loss curves shown in the figure are smoothed using Exponential Moving Average (EMA).

Method	TR	IR	Acc^{V2}	Acc^{A}	$\text{Acc}^{\text{cifar}}$	Acc^{ucf}
w/o init	74.2	62.8	13.6	**6.69**	32.3	24.8
w/ init	**76.4**	**65.1**	**13.9**	6.54	**32.8**	**27.3**

Table 3: Effect of gradient-approximation initialization on downstream performance. From left to right are the Flickr image-to-text and text-to-image recall (mean over Rank-1, 5, and 10), followed by zero-shot classification Acc-1 on ImageNet V2, ImageNet-A, CIFAR100, and UCF101.

Rank Distribution Analysis

We conducted an analysis of the fine-tuned model by counting the number of non-zero singular values in Λ for each SVD-like LoRA module. Figure 3 shows the distribution results obtained under the standard experimental setup.

The allocation across layers reveals that the value and projection matrices in the attention sublayers receive more focus than query and key. Aligning discrete text modalities with continuous image modalities places higher demands on the model's ability to parse and adapt to new input forms. At the same time, the attention allocation mechanism exhibits commonalities across modalities, partially explaining the variance among these matrices. In the linear sublayers, the upsampling matrices are allocated more attention than the downsampling ones. The former expands feature dimensions into higher-dimensional space, capturing richer semantic information that plays a crucial role in handling unfamiliar semantic structures in multimodal tasks. In contrast, downsampling matrices function as feature compression modules, leveraging single-modality compression knowledge, which is likely to be generalizable.

From the inter-layer allocation perspective, deeper transformer blocks are allocated more resources than shallower ones, a topic that has been extensively studied in previous research (Raghu et al. 2021; Chefer, Gur et al. 2021).

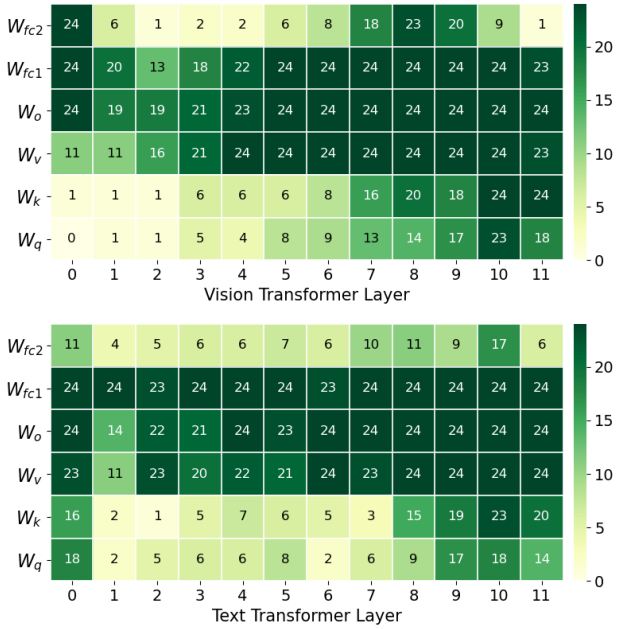

Figure 3: The resulting rank of each LoRA module. This result was obtained through training under the standard experimental setting. In the figure, the x-axis represents the indices of transformer blocks, and the y-axis represents the weight matrices of each block, where W_{f_1} and W_{f_2} belong to the FFN, W_q, W_k, W_v, and W_o belong to the MHSA. **Upper plot**: Vision Transformer. **Bottom plot**: Text Transformer.

This indicates that complex contexts and high-level semantic relationships are key to contrastive learning, explaining why models like CLIP perform well in natural-language-guided zero-shot tasks. These functionalities impose greater demands on deeper transformer blocks.

It is noteworthy that the shallowest layer (Layer 0) of the text modality exhibits counterintuitive behavior, with a significantly different allocation of resources compared to subsequent shallow transformer blocks, highlighting its critical role in multimodal alignment. Since PEFT typically does not fine-tune the embedding matrix, the anomalous behavior of Layer 0 may suggest that it assumes the function of modifying embeddings for new tasks, somewhat akin to soft prompt (Qin and Eisner 2021).

From Dynamic Discovery to Static Design: Can HALoRA's Insights Guide Manual LoRA? A key outcome of HALoRA's dynamic allocation is the consistent, asymmetric rank distribution it converges to Figure 3. While this distribution is a byproduct of our adaptive process, it raises a compelling question: does it merely reflect the quirks of our optimizer, or does it encapsulate a fundamentally superior static configuration? To investigate this, we tested whether this discovered "blueprint" could be used to improve standard, non-adaptive LoRA.

As shown in Table 4, we created two fixed-rank LoRA variants: one with an intuitive symmetric distribution (LoRA/LoRA-GA‡), where deeper layers in both modalities

Method	Flickr30k				ImageNet V2		ImageNet-A		CIFAR100		UCF101	
	TR@1	IR@1	TR@5	IR@5	Acc-1	Acc-5	Acc-1	Acc-5	Acc-1	Acc-5	Acc-1	Acc-5
LoRA	54.7	40.9	81.5	70.2	12.8	29.7	5.01	15.4	30.0	55.2	26.1	49.0
LoRA†	54.3	40.1	81.5	69.8	13.0	30.0	5.90	18.2	31.3	57.0	**26.7**	48.2
LoRA‡	**55.3**	**41.1**	**81.6**	**70.8**	**13.1**	**30.3**	**6.27**	**20.0**	**33.2**	**59.1**	26.6	**49.2**
LoRA-GA	56.4	42.0	82.2	70.6	12.8	29.4	5.35	17.0	30.1	56.3	25.4	48.6
LoRA-GA†	53.7	41.4	81.7	71.2	13.3	29.8	5.41	18.5	31.4	57.4	26.0	48.8
LoRA-GA‡	**56.6**	**42.4**	**82.7**	**72.3**	**13.5**	**30.7**	**6.06**	**20.2**	**33.3**	**59.9**	**27.1**	**49.3**

Table 4: LoRA under empirical rank allocation. The table compares different LoRA variants across datasets under manually designed rank distributions. † denotes a *symmetric* rank allocation strategy, where both vision and text encoders follow the same layer-wise trend, assigning lower ranks to shallow layers and higher ranks to deeper layers. ‡ denotes an *asymmetric* strategy, where vision and text encoders adopt different rank allocation patterns, allowing modality-specific emphasis at different depths. This asymmetric design reflects the intrinsic imbalance between visual and textual representations in vision-language alignment. The best performance in each column is highlighted in **bold**.

Shots	Method	ImageNet	SUN	Aircraft	EuroSAT	Cars	Food	Pets	Flowers	Caltech	DTD	UCF	Average
0	CLIP	66.7	62.6	24.7	47.5	65.3	86.1	89.1	71.4	92.9	43.6	66.7	65.1
	CLIP-LoRA	**70.4**	**70.4**	30.2	72.3	70.1	84.3	**92.3**	**83.2**	93.7	54.3	76.3	72.5
1	CLIP-LoRA†	70.2	70.3	29.6	71.9	69.7	85.0	91.7	82.6	93.8	54.1	76.0	72.3
	CLIP-LoRA‡	70.3	**70.4**	**30.3**	**72.5**	**70.4**	**85.5**	91.8	82.9	**94.0**	**54.7**	**76.7**	**72.7**
	CLIP-LoRA	71.4	**72.8**	37.9	84.9	77.4	82.7	91.0	93.7	95.2	63.8	81.1	77.4
4	CLIP-LoRA†	**71.7**	72.4	38.1	86.1	77.3	82.9	90.4	93.9	95.4	64.4	81.2	77.6
	CLIP-LoRA‡	71.5	72.6	**38.2**	**86.2**	**77.7**	**83.0**	**90.8**	**94.3**	**95.5**	**65.3**	**81.5**	**77.9**
	CLIP-LoRA	73.6	76.1	54.7	92.1	86.3	84.2	**92.4**	98.0	96.4	72.0	**86.7**	83.0
16	CLIP-LoRA†	73.6	76.4	56.2	91.8	86.3	**84.8**	91.8	**98.3**	96.5	73.3	86.4	83.2
	CLIP-LoRA‡	**73.9**	**76.5**	**56.7**	**92.5**	**86.4**	84.7	92.3	**98.3**	**96.7**	**73.5**	86.5	**83.5**

Table 5: Detailed results for 11 datasets with the ViT-B/16 as visual backbone. In each group, the third row (CLIP-LoRA‡) represents results using the asymmetric distribution. For comparison, a symmetric version was also designed and is displayed in the second row of each group (CLIP-LoRA†). Highest value is highlighted in **bold**.

receive higher ranks, and another with the asymmetric distribution discovered by HALoRA (LoRA/LoRA-GA‡). The results clearly show that the HALoRA-inspired asymmetric setup significantly outperforms both uniform allocation and the symmetric scheme. This confirms that the distribution learned by HALoRA is not just an artifact, but a valuable and generalizable principle for PEFT design.

Exploiting the Asymmetry in New Scenarios: A Case Study in Few-Shot Learning. To further test the utility of this discovered asymmetry, we applied it to a different task: the few-shot adaptation of CLIP, following the CLIP-LoRA setup (Zanella and Ayed 2024). We replaced their default uniform rank assignment with two configurations: a symmetric one and our discovered asymmetric one.

	Symmetry rank of LoRA for each layer											
# vision	1	1	1	1	1	1	3	3	3	3	3	3
# text	1	1	1	1	1	1	3	3	3	3	3	3

	Asymmetry rank of LoRA for each layer											
# vision	1	1	1	1	1	1	3	3	3	3	3	3
# text	3	1	1	1	1	1	1	3	3	3	3	3

The results in Table 5 consistently demonstrate that across multiple datasets and shot counts, the asymmetric rank assignment (CLIP-LoRA‡) yields superior performance compared to both the original CLIP-LoRA and the symmetric baseline (CLIP-LoRA†). This successful application in a new context provides strong evidence that the vision-language asymmetry is a robust phenomenon. It suggests that future PEFT methods, even static ones, can benefit from incorporating this architectural prior.

Conclusion

In this work, we address the challenge of applying LoRA to vision-language alignment. Our core contribution is the identification and mitigation of the hierarchical rank imbalance inherent in dual-encoder architectures. Through a novel Component-Wise Budget Allocator, our method, HALoRA, enables balanced and efficient fine-tuning, achieving performance comparable to adapter-based approaches with fewer parameters. Furthermore, our analysis of the learned rank distribution reveals a consistent asymmetry between modalities. This phenomenon hints at deeper architectural principles and warrants further investigation. Future work could explore the theoretical underpinnings of this asymmetry or extend our fine-grained allocation strategy to scenarios with additional modalities or diverse architectures. We hope our work fosters a shift towards more architecturally-aware PEFT designs in the multimodal domain.

Acknowledgments

We sincerely thank the anonymous reviewers and chairs for their efforts and constructive suggestions, which have greatly helped us improve the manuscript. This work is supported in part by the National Natural Science Foundation of China under grant 624B2088.

References

Bossard, L.; Guillaumin, M.; and Gool, L. V. 2014. Mining discriminative components with random forests. In *ECCV*.

Chefer, H.; Gur, S.; et al. 2021. Transformer interpretability beyond attention visualization. In *CVPR*.

Chen, S.; GE, C.; Tong, Z.; Wang, J.; Song, Y.; Wang, J.; and Luo, P. 2022. Adaptformer: Adapting vision transformers for scalable visual recognition. In *NeurIPS*.

Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; et al. 2014. Describing textures in the wild. In *CVPR*.

Eckart, C.; and Young, G. 1936. The approximation of one matrix by another of lower rank. *Psychometrika*.

Fei-Fei, L.; Fergus, R.; and Perona, P. 2007. An incremental bayesian approach tested on 101 object categories. *CVIU*.

Gao, K.; Li, Y.; Du, C.; Wang, X.; Ma, X.; Xia, S.-T.; and Pang, T. 2025. Imperceptible Jailbreaking against Large Language Models. *arXiv preprint arXiv:2510.05025*.

Gao, T.; Yao, X.; and Chen, D. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *EMNLP*.

Geva, M.; Schuster, R.; Berant, J.; and Levy, O. 2021. Transformer feed-forward layers are key-value memories. In *ACL*.

Hayou, S.; Ghosh, N.; and Yu, B. 2024. Lora+: Efficient low rank adaptation of large models. In *ICML*.

Helber, P.; Bischke, B.; Dengel, A.; and Borth, D. 2019. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE JSTARS*.

Hendrycks, D.; Zhao, K.; Basart, S.; et al. 2019. Natural adversarial examples. *arXiv preprint arXiv:1907.07174*.

Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; de Laroussilhe, Q.; Gesmundo, A.; et al. 2019. Parameter-efficient transfer learning for nlp. In *ICML*.

Hu, E. J.; yelong shen; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. Lora: Low-rank adaptation of large language models. In *ICLR*.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.; Parekh, Z.; Pham, H.; Le, Q. V.; Sung, Y.; Li, Z.; and Duerig, T. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*.

Jia, M.; Tang, L.; Chen, B.-C.; Cardie, C.; and Belongie, o. 2022. Visual prompt tuning. In *ECCV*.

Karpathy, A.; and Fei-Fei, L. 2017. Deep visual-semantic alignments for generating image descriptions. *IEEE TPAMI*.

Khan, Z.; and Fu, Y. 2023. Contrastive alignment of vision to language through parameter-efficient transfer learning. In *ICLR*.

Kim, W.; Son, B.; and Kim, I. 2021. Vision-and-language transformer without convolution or region supervision. In *ICML*.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. Object representations for fine-grained categorization. In *ICCVW*.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Li, J.; Li, D.; et al. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*.

Li, J.; Li, D.; et al. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*.

Li, J.; Selvaraju, R. R.; Gotmare, A.; Joty, S. R.; Xiong, C.; and Hoi, S. C. 2021. Vision and language representation learning with momentum distillation. In *NeurIPS*.

Li, J.; Wang, J.; Tan, C.; Lian, N.; Chen, L.; Wang, Y.; Zhang, M.; Xia, S.-T.; and Chen, B. 2025a. Enhancing partially relevant video retrieval with hyperbolic learning. In *ICCV*.

Li, X. L.; and Liang, P. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *ACL*.

Li, Y.; Zhen, L.; Sun, Y.; Peng, D.; Peng, X.; and Hu, P. 2025b. Deep Evidential Hashing for Trustworthy Cross-Modal Retrieval. In *AAAI*.

Liang, J.; Huang, W.; Wan, G.; Yang, Q.; and Ye, M. 2025. Sculpting lora for harmonizing general and specialized knowledge in multimodal large language models. In *CVPR*.

Lin, T.; Maire, M.; Belongie, S. J.; Hays, J.; Perona, P.; et al. 2014. Common objects in context. In *ECCV*.

Liu, H.; Gao, K.; Bai, Y.; Li, J.; Shan, J.; Dai, T.; and Xia, S.-T. 2025. Protecting your video content: Disrupting automated video-based llm annotations. In *CVPR*.

Liu, H.; Li, C.; Li, Y.; and Lee, Y. J. 2024a. Improved baselines with visual instruction tuning. In *CVPR*.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual instruction tuning. In *NeurIPS*.

Liu, S.; Wang, C.; Yin, H.; Molchanov, P.; Wang, Y. F.; Cheng, K.; et al. 2024b. Dora: Weight-decomposed low-rank adaptation. In *ICML*.

Liu, X.; Zheng, Y.; Du, Z.; Ding, M.; Qian, Y.; Yang, Z.; et al. 2021. Gpt understands, too. *arXiv preprint arXiv:2103.10385*.

Loshchilov, I.; and Hutter, F. 2019. Decoupled weight decay regularization. In *ICLR*.

Maji, S.; Rahtu, E.; Kannala, J.; et al. 2013. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*.

Meng, G.; He, S.; Wang, J.; Dai, T.; Zhang, L.; Zhu, J.; Li, Q.; Wang, G.; Zhang, R.; and Jiang, Y. 2025. EvdCLIP: Improving Vision-Language Retrieval with Entity Visual Descriptions from Large Language Models. In *AAAI*.

Meng, G.; Wang, J.; Wang, Q.-W.; Ren, X.; and Zhao, D. 2026a. Imagine with Layout and Sketch: Enhancing Vision-Language Retrieval with Dual-Stream Multi-Modal Query Refinement. In *AAAI*.

Meng, G.; Wang, J.; Zhu, J.; Zhang, L.; Jiang, Y.; Zhao, D.; and Li, Q. 2026b. Suit the Remedy to the Retriever: Interpretable Query Optimization with Retriever Preference Alignment for Vision-Language Retrieval. In *AAAI*.

Mirsky, L. 1960. Symmetric gauge functions and unitarily invariant norms. *The Quarterly Journal of Mathematics*.

Nilsback, M.-E.; and Zisserman, A. 2008. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics & Image Processing*.

Parkhi, O. M.; Vedaldi, A.; Zisserman, A.; and Jawahar, C. V. 2012. Cats and dogs. In *CVPR*.

Plummer, B. A.; Wang, L.; Cervantes, C. M.; Caicedo, J. C.; Hockenmaier, J.; et al. 2017. Collecting region-to-phrase correspondences for richer image-to-sentence models. *IJCV*.

Qin, G.; and Eisner, J. 2021. Learning how to ask: Querying lms with mixtures of soft prompts. In *NAACL*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*.

Raghu, M.; Unterthiner, T.; Kornblith, S.; Zhang, C.; and Dosovitskiy, A. 2021. Do vision transformers see like convolutional neural networks? In *NeurIPS*.

Rebuffi, S.-A.; Bilen, H.; and Vedaldi, A. 2017. Learning multiple visual domains with residual adapters. In *NeurIPS*.

Recht, B.; Roelofs, R.; Schmidt, L.; and Shankar, V. 2019. Do imagenet classifiers generalize to imagenet? In *ICML*.

Shen, Y.; Xu, Z.; Wang, Q.; Cheng, Y.; Yin, W.; and Huang, L. 2024. Multimodal instruction tuning with conditional mixture of lora. In *NAACL*.

Shukor, M.; Dancette, C.; Ramé, A.; and Cord, M. 2023. Unival: Unified model for image, video, audio and language tasks. *TMLR*.

Soomro, K.; Zamir, A. R.; and Shah, M. 2012. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Sung, Y.-L.; Cho, J.; et al. 2022. Parameter-efficient transfer learning for vision-and-language tasks. In *CVPR*.

Tang, H.; Wang, J.; Peng, Y.; Meng, G.; Luo, R.; Chen, B.; Chen, L.; Wang, Y.; and Xia, S.-T. 2025. Modeling uncertainty in composed image retrieval via probabilistic embeddings. In *ACL*.

Tang, H.; Wang, J.; Zhao, M.; Meng, G.; Luo, R.; and Long Chen, S.-T. X. 2026. Heterogeneous Uncertainty-Guided Composed Image Retrieval with Fine-Grained Probabilistic Learning. In *AAAI*.

Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; et al. 2021. Training data-efficient image transformers & distillation through attention. In *ICML*.

Touvron, H.; Cord, M.; and Jégou, H. 2022. Deit iii: Revenge of the vit. In *ECCV*.

Wang, J.; Chen, B.; Liao, D.; Zeng, Z.; Li, G.; Xia, S.-T.; and Xu, J. 2022a. Hybrid contrastive quantization for efficient cross-view video retrieval. In *WWW*.

Wang, J.; Zeng, Z.; Chen, B.; Wang, Y.; Liao, D.; Li, G.; Wang, Y.; and Xia, S.-T. 2022b. Hugs Are Better Than Handshakes: Unsupervised Cross-Modal Transformer Hashing with Multi-granularity Alignment. In *BMVC*.

Wang, J.; Zeng, Z.; Chen, B.; Wang, Y.; Liao, D.; Li, G.; Wang, Y.; and Xia, S.-T. 2024a. Hugs bring double benefits: Unsupervised cross-modal hashing with multi-granularity aligned transformers. *IJCV*.

Wang, J.; Zeng, Z.; Wang, Y.; Wang, Y.; Lu, X.; Li, T.; Yuan, J.; Zhang, R.; Zheng, H.-T.; and Xia, S.-T. 2023. Missrec: Pre-training and transferring multi-modal interest-aware sequence representation for recommendation. In *MM*.

Wang, L.; Qin, Y.; Sun, Y.; Peng, D.; Peng, X.; and Hu, P. 2024b. Robust contrastive cross-modal hashing with noisy labels. In *MM*.

Wang, S.; Yu, L.; and Li, J. 2024. Lora-ga: Low-rank adaptation with gradient approximation. In *NeurIPS*.

Wang, Y.; Wang, J.; Chen, B.; Zeng, Z.; and Xia, S.-T. 2024c. Gaussian-mixture-model based transformer for efficient partially relevant video retrieval. In *AAAI*.

Wang, Z.; Liang, J.; He, R.; Wang, Z.; and Tan, T. 2025. Are low-rank adapters properly optimized? In *ICLR*.

Xiao, J.; Hays, J.; Ehinger, K. A.; Oliva, A.; and Torralba, A. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*.

Yao, L.; Huang, R.; Hou, L.; Lu, G.; Niu, M.; Xu, H.; Liang, X.; Li, Z.; Jiang, X.; and Xu, C. 2022. Filip: Fine-grained interactive language-image pre-training. In *ICLR*.

Young, P.; Lai, A.; Hodosh, M.; and Hockenmaier, J. 2014. New similarity metrics for semantic inference over event descriptions. *TACL*.

Yu, J.; Wang, Z.; Vasudevan, V.; Yeung, L.; Seyedhosseini, M.; et al. 2022. Coca: Contrastive captioners are image-text foundation models. *TMLR*.

Zanella, M.; and Ayed, I. B. 2024. Low-rank few-shot adaptation of vision-language models. In *CVPR*.

Zhai, X.; Wang, X.; Mustafa, B.; Steiner, A.; Keysers, D.; Kolesnikov, A.; and Beyer, L. 2022. Lit: Zero-shot transfer with locked-image text tuning. In *CVPR*.

Zhang, Q.; Chen, M.; Bukharin, A.; He, P.; Cheng, Y.; Chen, W.; and Zhao, T. 2023. Adaptive budget allocation for parameter-efficient fine-tuning. In *ICLR*.

Zhang, Q.; Zuo, S.; Liang, C.; Bukharin, A.; et al. 2022. Pruning large transformer models with upper confidence bound of weight importance. In *ICML*.

Zhang, T.; Gao, K.; Bai, J.; Zhang, L. Y.; Yin, X.; Wang, Z.; Ji, S.; and Chen, W. 2025. Pre-training CLIP against Data Poisoning with Optimal Transport-based Matching and Alignment. In *EMNLP*.

Zhao, M.; Wang, J.; Liao, D.; Wang, Y.; Duan, H.; and Zhou, S. 2023. Keyword-Based Diverse Image Retrieval by Semantics-aware Contrastive Learning and Transformer. In *SIGIR*.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022. Learning to prompt for vision-language models. *IJCV*.