Branch, or Layer? Zeroth-Order Optimization for Continual Learning of Vision-Language Models

Ziwei Liu[1,3*], Borui Kang[2*], Wei Li[1], Hangjie Yuan[4], Yanbing Yang[1], Wenbin Li[2], Yifan Zhu[5], Tao Feng[6†], Jun Luo[3]

[1]College of Computer Science, Sichuan University, China
[2]School of Computer Science, Nanjing University, China
[3]College of Computing and Data Science, Nanyang Technological University, Singapore
[4]DAMO Academy, Alibaba Group, China
[5]College of Computer Science, Beijing University of Posts and Telecommunications, China
[6]Department of Computer Science and Technology, Tsinghua University, China

Abstract

Vision-Language Continual Learning (VLCL) has attracted significant research attention for its robust capabilities, and the adoption of Parameter-Efficient Fine-Tuning (PEFT) strategies is enabling these models to achieve competitive performance with substantially reduced resource consumption. However, dominated First-Order (FO) optimization is prone to trap models in suboptimal local minima, especially in limited exploration subspace within PEFT. To overcome this challenge, this paper pioneers a systematic exploration of adopting Zeroth-Order (ZO) optimization for PEFT-based VLCL. We first identify the incompatibility of naive full-ZO adoption in VLCL due to optimization process instability. We then investigate the application of ZO optimization from a modality branch-wise to a fine-grained layer-wise across various training units to identify an optimal strategy. Besides, a key theoretical insight reveals that vision modality exhibit higher variance than language counterparts in VLCL during the ZO optimization process, and we propose a modality-aware ZO strategy, which adopts gradient sign normalization in ZO and constrains vision modality perturbation to further improve performance. Benefiting from the adoption of ZO optimization, PEFT-based VLCL fulfills better ability to escape local minima during the optimization process, extensive experiments on four benchmarks demonstrate that our method achieves state-of-the-art results.

Introduction

Continual Learning (CL) has witnessed significant advancements in convolutional architectures (e.g., ResNet (Feng, Wang, and Yuan 2022; Rebuffi et al. 2017; Feng et al. 2022) and ViT(Wang et al. 2022b,a; Gao, Cen, and Chang 2024)). Recently, Vision-Language Models-based Continual Learning (VLCL) approaches have attracted growing research attention. Particularly, CLIP-based methods (Thengane et al. 2022; Ding et al. 2022; Zhao et al. 2023; Ni et al. 2023) have demonstrated robust continual learning capabilities. However, these methods require full-parameter fine-tuning of CLIP models, which incurs substantial computational overhead. To overcome this critical bottleneck, Parameter-Efficient Fine-Tuning (PEFT) strategies (Zhang et al. 2025a; Houlsby et al. 2019; Hung et al. 2019; Kang et al. 2025) have recently emerged as a compelling alternative. These techniques make it possible to achieve competitive CL performance with significantly reduced resource consumption. For instance, the VLCL method MoE4Adapter (Yu et al. 2024) leverages a PEFT approach to address this limitation.

Existing VLCL methods predominantly employ First-Order (FO) optimization strategy (Ruder 2016), which update parameters using precise gradients derived from backpropagation. While valued for their stable directional guidance, this approach becomes a drawback in the context of PEFT. Its deterministic update paths limit exploration during training , and the low-dimensional subspace to which PEFT confines optimization makes these methods susceptible to converging to sharp local optima that overfit the current task (Keskar et al. 2017; Möllenhoff and Khan 2023), leading to a performance drop when faced with new tasks and potentially exacerbating catastrophic forgetting. To explore solutions with stronger generalization capabilities, Zeroth-Order (ZO) optimization offers a promising alternative (Feng et al. 2025). Unlike traditional FO optimization, this method forgoes precise gradients from backpropagation, instead estimating performance with random perturbations, making it less prone to getting trapped in local minima when exploring a constrained space (Malladi et al. 2023; Zhang et al. 2025b), hence potentially applied in PEFT-based VLCL.

There is already a vast amount of ZO finetuning researches (Malladi et al. 2023; Zhang et al. 2025b), while they lacks of considering its common strategy of fully replacing FO optimization is applicable to VLCL, given the documented optimization disparities between modality branches (Liang et al. 2022; Jha, Gong, and Yao 2024; Peng et al. 2022; Sun et al. 2020; Cheng et al. 2024) and the inherent sensitivity of ZO's perturbation-based approach. In this paper, we initiate the study of applying ZO optimization into VLCL, and aim to answer the following key question:

*These authors contributed equally.
†Corresponding author.

How can ZO be integrated in VLCL settings, and can it boost overall performance?

To answer the question, this paper adopts a fine-grained perspective, investigating the performance of ZO when applied to different modality branches and trainable layers within PEFT-based VLCL. Specifically, we respectively explore ZO in the vision or language modality branch while retaining FO in the other, to identify their benefits and limitations. Based on the insights gained, we then extend our investigation to a more granular layer-wise and adopt ZO into different training units, including continuous *prefix/suffix* layers and interleaved *odd/even* layers across a modality branch, to obtain a optimal result. Meanwhile, we identify a convergence discrepancy between modalities of VLCL under ZO optimization, thus proposing a **Mo**dality-aware **ZO** (**MoZO**) strategy, adopting gradient sign normalization in ZO and constraining vision modality perturbation to further boost VLCL performance.

In summary, the main contributions of this work are:

- We present the optimization challenge in PEFT-based VLCL, where conventional FO method is prone to converging to suboptimal local optima, and explore to leverage the ZO optimization to address the problem.
- We adopt the ZO optimization at both branch-wise and fine-grained layer-wise within PEFT-based VLCL, achieving optimal performance through detailed analysis and refined application strategies.
- We identify the issue of optimization discrepancy between modalities of PEFT-based VLCL under ZO optimization, and propose a **MoZO** strategy further improving overall performance.

A Preliminary for ZO Optimization in PEFT-based VLCL

Traditional optimization methods in continual learning primarily rely on FO gradient descent (Cha et al. 2020; Hadsell et al. 2020), updating model parameters θ based on precise gradients computed via backpropagation:

$$\theta_{t+1} = \theta_t - \eta_t \cdot \nabla_\theta \mathscr{L}(\theta_t), \tag{1}$$

where $\nabla_\theta \mathscr{L}(\theta_t)$ is the first-order gradient. While FO methods offer accurate gradient directions and have been widely, their deterministic update paths tend to limit exploration during training, potentially increasing the risk of convergence to suboptimal local minima and reducing adaptability in dynamic continual learning scenarios.

In contrast, ZO optimization estimates gradients through forward passes through purposeful perturbations. For a parameter subset θ_m^k, ZO approximates gradients via directional perturbations:

$$\nabla_{\mathrm{ZO}} \mathscr{L}(\theta_m^k) \approx \frac{\mathscr{L}(\theta_m^k + \varepsilon\Delta) - \mathscr{L}(\theta_m^k)}{\varepsilon} \cdot \Delta, \tag{2}$$

where Δ is a random directional vector, and ε is a small perturbation scale. ZO methods introduce gradient stochasticity (Berahas et al. 2022), which may improve the ability to explore non-convex loss landscapes and escape poor local minima. However, the reliance on randomized perturbations can lead to variance in gradient estimates, whose effect may vary across different model architectures or modalities. To validate this point, we first establish a baseline and investigate the most straightforward strategy: a complete replacement of the FO optimizer with ZO in PEFT-based VLCL, which means adopt ZO in both vision-language branches and overall trainable units. However, our subsequent experiments demonstrate this strategy leads to severe training instability, evidenced by loss oscillations during the convergence process, and results in a significant performance degradation.

Recent research suggests that partially incorporating exploratory while high-variance ZO estimators into the model architecture improves its global optimization performance (Talaei et al. 2025; Chen, Huang, and Wen 2025). We translate this advance to the context of PEFT-based VLCL. As shown in Figure 1, we adopt a empirical exploration to progressively investigate how to best apply the ZO optimization in VLCL. To explore suitable application paradigms, we designed a comprehensive set of experiments applying ZO at varying levels of granularity, from entire modality branches to fine-grained partially trainable units. However, given the vast diversity of configurations arising from combinations of modality branches and PEFT trainable units, constructing a single, unified theoretical framework is intractable. Consequently, we conduct a comprehensive empirical analysis to investigate both the differential impacts of applying ZO across modality branches and the distinct characteristics of employing it in various trainable layer configurations (such as continuous and interleaved layers), which allows us to achieve optimal results.

Study of ZO Optimization in VLCL

Implementation

Datasets and task construction. We evaluate our method on three datasets: CIFAR-100 (CIFAR), Tiny-ImageNet (TinyImg), and ImageNet-R (ImgR). For task construction under the CIL paradigm, we adopt the IncX configuration (e.g., Inc20 denotes 5 tasks with 20 classes each on CIFAR). All tasks enforce disjoint class distributions and exclude task-specific identifiers during inference to ensure a rigorous evaluation protocol.

Baseline. We choose MoE4Adapter (Yu et al. 2024) as the SOTA baseline, which incorporates Mixture of Experts (MoE) into CLIP for VLCL. To explore PEFT alternatives, we also replaces the MoE modules with Low-Rank Adaptation (LoRA (Hu et al. 2022)) modules. For LoRA results, we present a part of them and put the remaining in supplementary material.

Implementation details. All experiments employ the CLIP-ViT-B/16 backbone architecture. The CLIP backbone remains frozen, with only task-specific adapters (MoE or LoRA modules) being trainable. For the ZO-based method, we adopt a more conservative ZO strategy which evaluates multiple candidate updates and selectively applies the one that yields the lowest loss(Feng et al. 2025). Hyperparameters including perturbation scale $\varepsilon = 0.001$ for ZO gradients and FO/ZO mixing ratio $\lambda = 1$, are validated on the first task and retained for subsequent tasks.

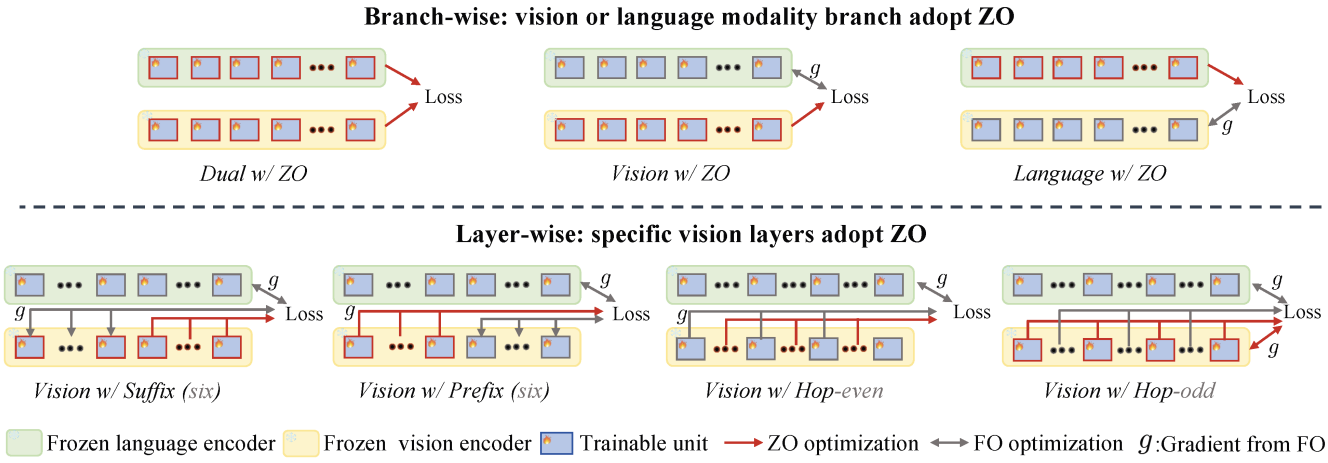
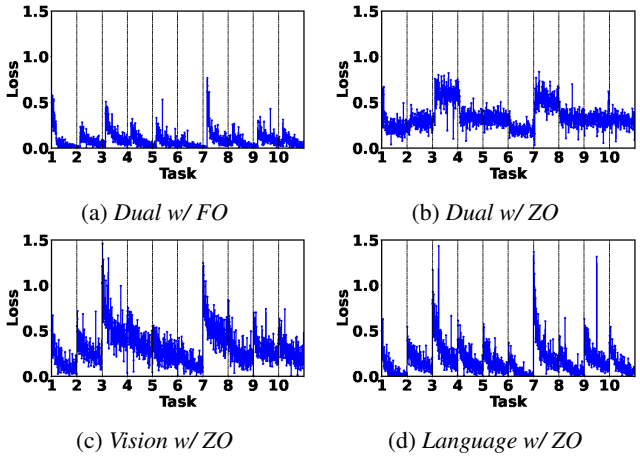


Figure 1: Illustration of our study. The language and vision encoders of **CLIP** are frozen, only the trainable units attached to each layer is performed to parameters update. To sum up, we systematically explores how ZO optimization operates in VLCL, including branches (*Dual*, *Vision*, or *Language*) and layers (*w/ Hop-odd*, *w/ Hop-even*, *w/ Prefix (six)* and *w/ Suffix (six)*).

Method	CIFAR Inc20		CIFAR Inc10		TinyImg Inc20		ImgR Inc20	
	Last.	Avg.	Last.	Avg.	Last.	Avg.	Last.	Avg.
Baseline	80.47	86.97	77.52	85.21	52.13	60.55	65.36	71.53
Du. w/ ZO	69.29	77.36	67.64	75.88	42.40	47.64	58.56	65.92
Vis. w/ ZO	76.05	83.93	72.98	82.08	49.65	57.90	62.54	69.84
Lan. w/ ZO	80.94	87.00	76.74	85.03	49.14	58.69	64.38	70.38
Baseline†	80.44	87.10	79.66	86.34	51.93	59.80	64.34	71.79
Du. w/ ZO	71.07	78.37	69.70	76.82	44.53	52.71	58.71	65.97
Vis. w/ ZO	75.86	83.86	73.73	82.25	49.98	58.11	62.70	69.88
Lan. w/ ZO	77.47	85.40	79.63	87.01	49.90	58.92	64.33	70.59

Table 1: How ZO optimization affects VLCL in different branches (CLIP). *w/ ZO* denotes the branch (*Du. (Dual)*, *Vis. (Vision)*, or *Lan. (Language)*) where ZO optimization is applied. The † indicates MoE modules in baseline are replaced with LoRA.



(a) *Dual w/ FO* (b) *Dual w/ ZO*

(c) *Vision w/ ZO* (d) *Language w/ ZO*

Figure 2: How ZO optimization affects loss convergence of VLCL in different branches (CLIP). *w/ ZO* denotes the branch (*Dual*, *Vision*, or *Language*) where ZO applied.

Rethinking ZO Optimization in VLCL

The potential of ZO optimization in VLCL. PEFT-based VLCL harnesses trainable units in vision-language model to achieve parameter-efficient adaptation. Conventional FO optimization method relies on precise gradient descent, hence leading to the attraction by local optima and suboptimal convergence. To mitigate these challenges, we first explore integrate full ZO optimization into VLCL. By leveraging its perturbation-based search mechanism, ZO enhances exploration in the parameter space, enabling escape from local optima. *However, does naively replacing FO with ZO necessarily lead to better performance?*

Analysis of naive ZO optimization failure in VLCL. We attempt to apply ZO into the VLCL, a straightforward intuition is to replace FO optimizers with ZO methods across both the vision and language branches including all trainable units, and the results are shown in Table 1. However, it can be observed when ZO is adopted in both vision and language branch (dual *w/ ZO*), the performances are significantly degraded regardless of MoE or LoRA settings, with Last. and Avg. averagely decreasing by 8.5% and 9.5% respectively. It can be attributed to the reason that the variance of ZO destabilizes VLCL training, leading to the optimization process is difficult to converge. To verify this point, we plot the loss function trajectories across these experimental settings shown in Figure 2. The Figure 2a and 2b reveal that dual-branch ZO approach suffers from severe loss oscillations and failed convergence compared to the original FO optimization. From this observation, we conclude that full ZO optimization is fundamentally ill-suited for VLCL, as its inherent gradient estimation fluctuations induce training instability. On the contrary, FO optimization provides stable gradient directions throughout the training process. This raises the question: *Can synergistic integration of ZO and FO optimization achieve better performance?*

How can ZO optimization be effective in VLCL? Branches, or Layers? We further investigate a hetero-

geneous optimization strategy: applying ZO to only one modality branch while retaining FO for the other. From a qualitative perspective, Figure 2c and 2d reveal when ZO is adopted into a single branch (vision or language), the overall loss function trajectory is promising to converge compared with dual branch ZO. We consider that the single-branch FO provides optimization stability, guiding the overall training process maintains consistency. Meanwhile, we can find that the performance obtains significant improvement shown in Table 1, most of the results are close to the baseline, and even some results outperform baseline. It can be explained that ZO's perturbation-based gradient estimation introduces controlled stochasticity into the optimization process, which probabilistically assists the optimizer in evading suboptimal local minima. Additionally, we find that the performance of language *w/ ZO* generally outperform vision *w/ ZO*, we argue that the optimization stability of language branch is stronger than vision branch, which is verified by the loss trajectories of Figure 2c and 2d. These findings suggest a ZO-FO synergy strategy which balances performance exploration and training convergence in VLCL, validating the feasibility of the integration optimization.

Building on these observations, the effectiveness of coarse-grained integration motivates us to explore the potential of synergistic optimization between ZO and FO methods. We further consider a fine-grained perspective: *Can layer-wise allocation of ZO and FO optimization within modality-specific enhance CL performance?*

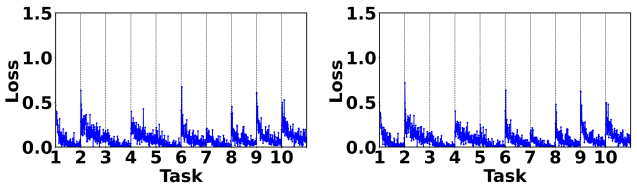
Why Layers Matter: Triggering Effective Continual Adaptation

To explore the performance regarding layer-wise allocation of ZO and FO, four layer-wise ZO patterns were tested: *Hop-odd* (adopt ZO in odd layers), *Hop-even* (adopt ZO in even layers), *Prefix (six)* (adopt ZO in first six layers) and *Suffix (six)* (adopt ZO in last six layers) in dual or single modality branch, with FO used in other layers. The baseline and selection of modality branch is still refer to Table 1.

Layer-wise ZO unlocks VLCL performance potential. We then apply layer-wise ZO optimization to both dual and single modality branches, with the remaining layers optimized by FO. As shown in Table 2, we observe that layer-wise strategy can provide significant performance improvements compared to applying ZO across all trainable units in branches. In dual *w/ ZO*, the layer-wise ZO optimization averagely improves 9.4% accuracy on four patterns across all dataset. More interestingly, we observe that certain fine-grained layer-wise ZO patterns can outperform full FO approaches. To investigate the layer-wise effectiveness, we first analyze the performance of applying the collaborative ZO-FO optimization to a uniform layering strategy (with *Hop-odd* selected as a representative case). The loss trajectories under different modalities are recorded and presented in Figure 3, it demonstrates a more stable convergence process during training compared to that shown in Figure 2. We argue that more fine-grained layer-wise strategy further amplifies the respective advantages of ZO and FO in VLCL, FO provides stable gradient directions and ZO's stochastic perturbations help the optimization escape from local minima.

Method	CIFAR Inc20		CIFAR Inc10		TinyImg Inc20		ImgR Inc20	
	Last.	Avg.	Last.	Avg.	Last.	Avg.	Last.	Avg.
Dual w/ ZO	69.29	77.36	67.64	75.88	42.40	47.64	58.56	65.92
w/ Hop-odd	81.64	88.11	79.22	86.98	51.68	59.75	65.24	71.59
w/ Hop-even	81.59	88.07	79.12	86.81	51.95	60.53	64.99	71.29
w/ Prefix (six)	79.12	85.89	76.53	84.77	50.47	59.34	65.01	71.36
w/ Suffix (six)	80.98	87.52	78.03	86.63	51.48	59.33	63.10	69.99
Vision w/ ZO	76.05	83.93	72.98	82.08	49.65	57.90	62.54	69.84
w/ Hop-odd	81.83	88.36	**79.39**	86.96	51.56	60.11	**65.68**	72.05
w/ Hop-even	**82.41**	88.36	78.95	86.74	**52.27**	**60.98**	64.99	72.10
w/ Prefix (six)	79.34	86.17	76.72	84.85	50.14	59.73	65.25	71.96
w/ Suffix (six)	82.21	88.33	79.23	87.03	51.65	60.20	64.25	70.76
Language w/ ZO	80.94	87.00	76.74	85.03	49.14	58.69	64.38	70.38
w/ Hop-odd	82.28	**88.51**	79.28	**87.05**	51.73	60.59	65.20	71.90
w/ Hop-even	82.17	88.45	78.73	86.27	52.07	60.71	65.06	**72.16**
w/ Prefix (six)	82.19	88.51	79.07	86.83	52.02	60.49	64.87	71.64
w/ Suffix (six)	82.19	88.30	78.82	86.71	51.57	60.38	65.17	72.10

Table 2: How ZO optimization affects VLCL through different layers (CLIP). We design four configurations across layers from different branches: *w/ Hop-odd* (ZO in odd layers), *w/ Hop-even* (ZO in even layers), *w/ Prefix (six)* (ZO in first six layers) and *w/ Suffix (six)* (ZO in last six layers).



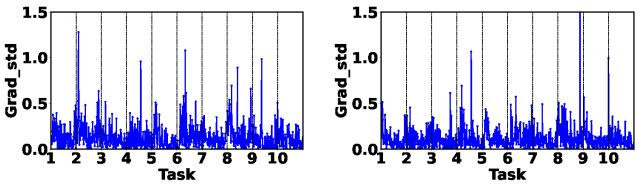
(a) Du. *w/Hop-odd*. (b) Lan. *w/Hop-odd*.

Figure 3: Analyzing convergence behavior of VLCL in *Hop-odd* across Dual (Du.) and Language (Lan.) branches.

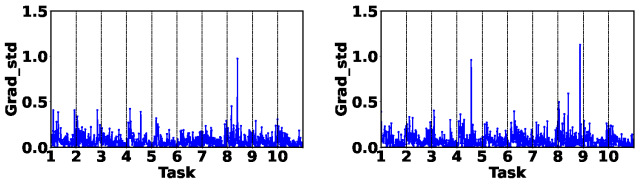
Observation on Layer-wise Heterogeneity in ZO for VLCL. To gain deeper insights, we start to analyze the impact of different layer-wise settings on performance. Interestingly, we observe that all SOTA results emerge when ZO and FO optimization are applied in an interleaved manner across layers (e.g., on *Hop-odd* or *Hop-even*). To further investigate this phenomenon, we analyze the gradient behavior under four different layer-wise configurations within the language branch. The corresponding gradient distributions are visualized in Figure 4. We clearly observe that the gradient variance is significantly lower when ZO and FO are interleaved across layers, compared to configurations where either ZO or FO is applied continuously throughout. We hypothesize that this benefit stems from the functional heterogeneity across layers: shallow layers focus on local features, while deeper layers capture abstract semantics. A uniform optimization method may overlook such diversity, whereas interleaving ZO and FO better aligns with each layer's exploration and stability needs, leading to a more robust optimization process that stabilizes gradient flow while facilitating escape from local minima.

A New Enhancement from Vision Discrepancy

Understanding the discrepant behavior of ZO in vision branch. From Figure 2, we observe that when ZO is adopted



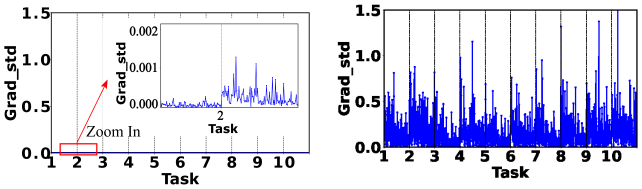
(a) *Prefix (six)*. (b) *Suffix (six)*.



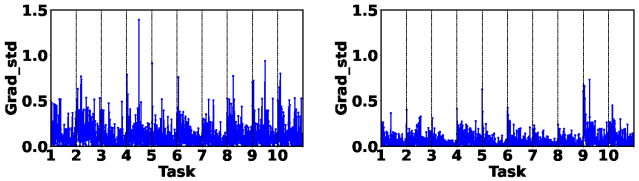
(c) *Hop-odd*. (d) *Hop-even*.

Figure 4: How ZO optimization affects gradient variance across layers in VLCL.

in a single branch, the loss convergence trajectory of visual branch demonstrates significantly inferior performance compared with language branch. To further investigate this phenomenon, we record the gradient variance distribution of different optimization strategies from a layer-wise perspective in VLCL, since layer-wise ZO obtain better performance, and the visualization results are shown in Figure 5 and 6. It can be observed that dual branch FO optimization exhibit a minimal numerical fluctuations in gradient variance, reflecting robust convergence stability regardless of *Hop-odd* or *Hop-even* layers setting, while the lack of sufficient fluctuation might lead the optimization process to converge to local optima. In contrast, dual branch ZO induce severe oscillations, causing the optimization trajectory to diverge. When the ZO-FO collaborative mechanism is employed, we interestedly observe that under the same layer wise setting, the gradient variance of the visual branch ZO is also more violent than that of language. This observation motivates a critical inquiry: *Could targeted suppression of*



(a) *Dual w/ FO* (b) *Dual w/ ZO*



(c) *Vision w/ ZO* (d) *Language ZO*

Figure 5: Analyzing gradient variance of VLCL in *Hop-odd* across *Dual*, *Vision*, *Language*.

Method	CIFAR Inc10		TinyImg Inc20		ImgR Inc20	
	Last.	Avg.	Last.	Avg.	Last.	Avg.
Dual *w/ Hop-odd*	79.22	86.98	51.68	59.75	65.24	71.59
MoZO	**79.36**	**87.02**	**52.35**	59.75	**65.32**	**71.93**
Dual *w/ Hop-even*	79.12	86.81	51.95	60.53	64.99	71.29
MoZO	**79.87**	**87.25**	**52.46**	**61.23**	**65.80**	**71.82**

Table 3: Effect of MoZO optimization on performance. Dual *w/ Hop-odd/even* indicates the results of adopting *Hop-odd* and *Hop-even* layer-wise ZO in dual branch.

ZO-induced perturbations in the visual modality yield performance gains?

Gradient Regularization and Vision Branch Perturbation Control. To validate this hypothesis, we propose a MoZO optimization strategy that incorporates gradient regularization during the estimation process, with explicit constraints on perturbations applied to the vision branch to mitigate instability. Specifically, we first introduce a signed gradient transformation to regularize the ZO-estimated gradient $\hat{g}_{ZO}$. This can be expressed as:

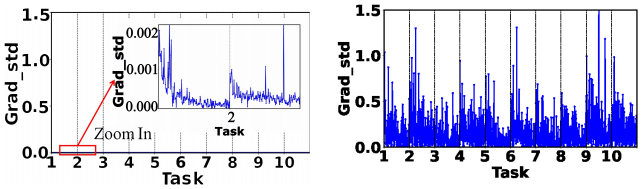
$$\tilde{g} = \text{sign}(\hat{g}_{ZO}), \tag{3}$$

where $\hat{g}_{ZO}$ denotes the original ZO-estimated gradient, and $\tilde{g}$ is the transformed signed gradient. The $\text{sign}(\cdot)$ function is applied element-wise, defined as:

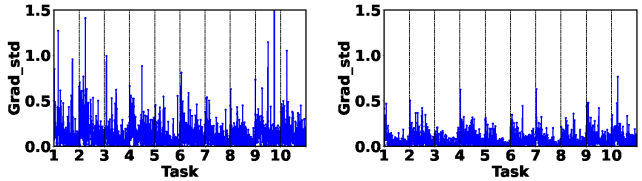
$$\text{sign}(x_i) = \begin{cases} +1. & \text{if } x_i > 0 \\ 0. & \text{if } x_i = 0 \\ -1. & \text{if } x_i < 0 \end{cases} \tag{4}$$

This transformation retains only the direction information of the gradient, discarding the amplitude information. Furthermore, we implement modality-specific perturbation factors for ZO optimization, assigning a deliberately lower value to the vision branch (ε_v) compared to the language branch (ε_l). Hence, the final MoZO update rule can be expressed as:

$$\theta_{m,t+1} = \begin{cases} \theta_{m,t} - \eta_t \cdot \tilde{g}(\theta_{m,t}, \varepsilon_v \xi_t), & \text{if } m = \text{vision} \\ \theta_{m,t} - \eta_t \cdot \tilde{g}(\theta_{m,t}, \varepsilon_l \xi_t), & \text{if } m = \text{language} \end{cases} \tag{5}$$

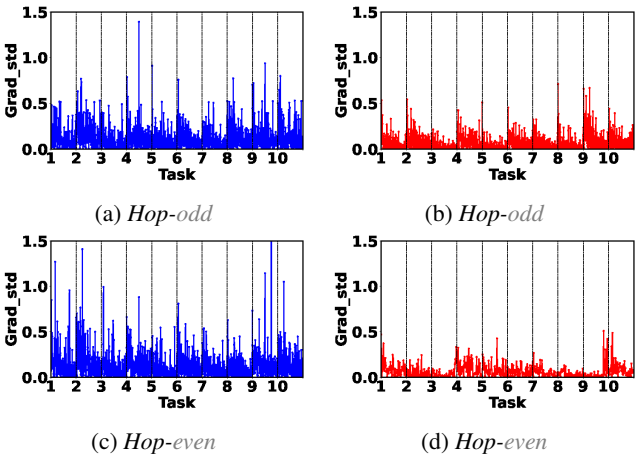


(a) *Dual w/ FO* (b) *Dual w/ ZO*



(c) *Vision w/ ZO* (d) *Language w/ ZO*

Figure 6: Analyzing gradient variance of VLCL in *Hop-even* across *Dual*, *Vision*, *Language*.



(a) *Hop-odd* (b) *Hop-odd*

(c) *Hop-even* (d) *Hop-even*

Figure 7: Analyzing the effect of MoZO optimization in *Hop-odd* vs. *Hop-even*. Blue shows original results, while red shows the positive impact of vision discrepancy.

where θ is the gradient parameter, η_t is the learning rate, and ξ_t represents a perturbation vector. To verify this strategy, we first record the gradient variance on CIFAR, and plot the comparison shown in Figure 7. It can be observed that our strategy significant reduce the vision branch gradient fluctuations expressed in Figure 7b and 7d (shown in red line), regardless of *Hop-odd* or *Hop-even*. We further validate this conjecture through quantitative analysis shown in Table 3, it can be find that stabilizing ZO gradient estimation and reducing vision branch perturbations yields consistent performance gains across CIFAR, TinyImg and ImgR datasets. This method focus on the instability limits of ZO in VLCL and maintains the balance of the optimization process across different modality branches, paving the way to further refines the application strategy of ZO optimization in VLCL.

Ablation, Demonstration and Beyond

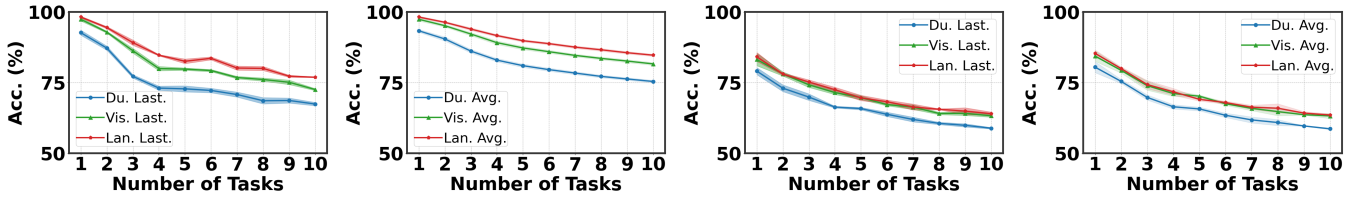
Exploring diverse ZO strategies. Our ablation start to explore other ZO strategies of hybrid ZO-FO collaboration in VLCL. As shown in Table 4, applying ZO to dual branches significantly degrades performance compared to the baseline, confirming the instability caused by excessive gradient oscillations. In contrast, single-branch ZO optimization mitigates this issue, with the significant improvement in Last. and Avg. metrics. We then explore the impact of different ZO optimization on VLCL performance. As we default to the conservative ZO strategy in the main analysis, we further conduct more aggressive ZO variants. Specifically, we examine a naive ZO approach—referred to as ZO* which performs a single gradient estimation and directly updates the parameters without any loss-based validation. On top of this, we introduce a variant named Sign, which incorporates a signed gradient transformation to regulate the magnitude of the estimated gradients. This design aims to mitigate the instability caused by ZO estimates while preserving directional information. It can be observed that the aggressive ZO* optimization degrades performance, as it relies on a single gradient estimation without validating the update direction, making it more sensitive to the fluctuations in the loss landscape compared with conservative ZO. Notably, adopting Sign-based gradient estimation enhances performance across all configurations, suggesting that enforcing gradient amplitude consistency in ZO perturbations helps stabilize the optimization trajectory and improves convergence. This observation aligns with our proposed ZO optimization strategy in Section , which advocates for gradient magnitude control as a means to suppress fluctuation updates and maintain optimization balance across modalities.

Method	**Strategy**	**CIFAR Inc10** Last.	Avg.
MoE4Adapter			
Dual w/ ZO	ZO*	66.67	75.08
Vision w/ ZO	ZO*	72.32	81.64
Language w/ ZO	ZO*	**76.19**	**84.58**
Dual w/ ZO	Sign	66.68	77.20
Vision w/ ZO	Sign	73.34	83.47
Language w/ ZO	Sign	**78.52**	**85.91**

Table 4: Behavioral consistency across diverse ZO strategies. * indicates a naive ZO strategy, Sign indicates a ZO strategy using the gradient transformation method.

Significance analysis. We then explore the average incremental performance among five runs on two dataset (CIFAR and ImgR), with three different ZO configurations (dual branch, vision branch and language branch), recording the accuracy of Last. and Avg. shown in Figure 8. We observe that ZO optimization yields smaller performance variance on the CIFAR dataset than on ImgR across different configurations, regardless of Last. and Avg. metrics. One possible explanation is that ImgR derived from the larger-scale ImageNet, contains more fine-grained and semantically diverse categories than CIFAR. This increased label complexity leads to a more rugged and high-dimensional loss landscape, which makes ZO gradient estimation on stochastic perturbations harder to stabilize. Additionally, from the perspective of different ZO configurations, applying ZO optimization to the language branch consistently yields the better performance. We hypothesize that this is because the language branch typically operates on lower-dimensional tensors, making it less susceptible to the instability introduced by random perturbations in ZO gradient estimation. In contrast, the higher-dimensional vision branch benefits more from FO gradients, which provide precise gradient optimization direction. The combination of FO-dominated updates in the vision branch and ZO exploration in the language branch may facilitate better escape from local minima and lead to more robust convergence.

Quantifying memory efficiency. We record the memory usage across different ZO settings, as shown in Table 5. Compared to applying FO optimization in the dual branch baseline, using ZO optimization leads to a 89.1% reduction in memory consumption, due to the elimination of gradient backpropagation and storage overhead. When applying ZO optimization to a single branch, memory consumption is reduced by 65.3% on the visual branch and 37.9% on the lan-



(a) Last. of CIFAR. (b) Avg. of CIFAR. (c) Last. of ImgR. (d) Avg. of ImgR.

Figure 8: Significance analysis of performance across *Dual (Du.)*, *Vision (Vis.)*, *Language (Lan.)*.

Method	MoE4Adapter	MoE4Adapter†
Baseline	∼19.96GB	∼15.11GB
Dual w/ ZO	∼2.17GB ↓	∼1.73GB ↓
Vision w/ ZO	∼6.93GB ↓	∼5.71GB ↓
Language w/ ZO	∼12.39GB ↓	∼11.09GB ↓

Table 5: Comparison of GPU memory usage between different ZO settings across branches (**`CLIP`**).

guage branch, respectively. The greater memory reduction on the visual branch is attributed to its higher input dimensionality, as it processes image data. Applying ZO in this context helps to alleviate memory pressure by avoiding storage of high-dimensional gradients. For the results in LoRA architecture, it inherently requires fewer trainable units to perform parameters update, resulting in overall lower memory consumption. Overall, these findings highlight ZO's inherent advantage in significantly reducing memory consumption, especially in high-dimensional settings, making it a highly practical and efficient optimization alternative for resource-constrained VLCL scenarios where memory efficiency is critical.

Related Work

Continual learning of VLM. Recent advances in CL have witnessed the emergence of VLMs as promising solutions to mitigate catastrophic forgetting through their generalized multimodal representations. Pioneering studies (Ding et al. 2022; Thengane et al. 2022) demonstrate that pretrained VLMs like CLIP (Radford et al. 2021) inherently possess remarkable continual learning capabilities even without fine-tuning. PROOF (Ding et al. 2022) further enhances CL robustness by integrating multimodal cues with adaptive mapping strategies. Nevertheless, conventional VLM-based approaches predominantly emphasize task-specific feature acquisition for new domains (Huang et al. 2024), inadvertently compromising the integrity of previously learned representations and leading to progressive performance degradation. To address this limitation, recent efforts adopt PEFT techniques (Houlsby et al. 2019; Hung et al. 2019) that selectively update lightweight modules while maintaining frozen backbone parameters. MoE4Adapters (Yu et al. 2024) introduces a mixture-of-experts (MoE) architecture, enabling task-specific feature specialization without cross-task interference and achieving state-of-the-art performance. However, these methods universally rely on FO optimization, inherently restricting their capacity to explore optimal parameter trajectories during optimization.

Optimization for continual learning. From an optimization view, existing CL methods predominantly focus on reconciling the stability-plasticity dilemma (Lu et al. 2025) through gradient technology (Li et al. 2024). Orthogonal parameter updates (Saha, Garg, and Roy 2021; Lin et al. 2022; Lopez-Paz and Ranzato 2017; Farajtabar et al. 2020) and sharpness-aware minimization (Deng et al. 2021; Shi et al. 2021; Bian et al. 2024) represent two mainstream directions, which respectively aim to decouple task-specific gradients and converge to flat loss minima for improved generalization. While these strategies enhance optimization stability, they often inadequately address the exploration-exploitation trade-off, as deterministic gradient descent trajectories tend to converge to suboptimal local minima with limited perturbation resilience. Emerging ZO optimization(Nesterov and Spokoiny 2017; Berahas et al. 2022) presents a paradigm shift by employing gradient-free stochastic perturbations to estimate descent directions. This approach offers dual advantages: 1) inherent stochasticity facilitates escape from local optima through controlled parameter space exploration, and 2) elimination of backward propagation reduces memory overhead by avoiding gradient matrix storage.

Our work. This work investigates how ZO optimization can be effectively adapted to VLCL, addressing catastrophic forgetting by harmonizing the inter-modal asymmetry of VLMs with the stochastic perturbations of ZO optimization.

Conclusion

In this work, we systematically investigate ZO optimization into VLCL and propose a novel hybrid ZO-FO optimization paradigm. Through extensive empirical analysis, we identify two critical challenges of applying ZO in VLCL: destabilized training caused by excessive gradient variance and modality-specific optimization discrepancies. To address these, we first demonstrate that selectively applying ZO to specific branches (vision or language) while retaining FO optimization in others significantly outperforms naive full-ZO approaches. Building on this, we further propose a layer-wise collaborative strategy that interleaves ZO and FO across network layers, achieving state-of-the-art performance by harmonizing stochastic exploration with deterministic refinement.

Limitation. This work currently focuses on CLIP-based vision-language modalities. The generalization of ZO-FO collaboration to other VLMs (e.g., multimodal transformers with audio or video inputs) remains unexplored, particularly in scenarios where modalities exhibit heterogeneous feature distributions or temporal dependencies.

Acknowledgements

This work is supported in part by the National Natural Science Foundation of China (62192783), Young Elite Scientists Sponsorship Program by CAST (2023QNRC001), and China Scholarship Council with award number 202406240114.

References

Berahas, A. S.; Cao, L.; Choromanski, K.; and Scheinberg, K. 2022. A theoretical and empirical comparison of gradient approximations in derivative-free optimization. *Foundations of Computational Mathematics*, 22(2): 507–560.

Bian, A.; Li, W.; Yuan, H.; Wang, M.; Zhao, Z.; Lu, A.; Ji, P.; Feng, T.; et al. 2024. Make Continual Learning Stronger via C-Flat. *Advances in Neural Information Processing Systems*, 37: 7608–7630.

Cha, S.; Hsu, H.; Hwang, T.; Calmon, F. P.; and Moon, T. 2020. CPR: classifier-projection regularization for continual learning. *arXiv preprint arXiv:2006.07326*.

Chen, M.; Huang, Y.-L.; and Wen, Z. 2025. Towards Efficient Low-Order Hybrid Optimizer for Language Model Fine-Tuning. In *AAAI*.

Cheng, S.; He, C.; Chen, K.; Xu, L.; Li, H.; Meng, F.; and Wu, Q. 2024. Vision-sensor attention based continual multimodal egocentric activity recognition. In *ICASSP*, 6300–6304.

Deng, D.; Chen, G.; Hao, J.; Wang, Q.; and Heng, P.-A. 2021. Flattening Sharpness for Dynamic Gradient Projection Memory Benefits Continual Learning. *Advances in Neural Information Processing Systems*, 34: 18710–18721.

Ding, Y.; Liu, L.; Tian, C.; Yang, J.; and Ding, H. 2022. Don't Stop Learning: Towards Continual Learning for the CLIP Model. *arXiv preprint arXiv:2207.09248*.

Farajtabar, M.; Azizan, N.; Mott, A.; and Li, A. 2020. Orthogonal Gradient Descent for Continual Learning. In *International conference on artificial intelligence and statistics*, 3762–3773. PMLR.

Feng, T.; Ji, K.; Bian, A.; Liu, C.; and Zhang, J. 2022. Identifying players in broadcast videos using graph convolutional network. *Pattern Recognition*.

Feng, T.; Li, W.; Zhu, D.; Yuan, H.; Zheng, W.; Zhang, D.; and Tang, J. 2025. ZeroFlow: Overcoming Catastrophic Forgetting is Easier than You Think. *arXiv preprint arXiv:2501.01045*.

Feng, T.; Wang, M.; and Yuan, H. 2022. Overcoming Catastrophic Forgetting in Incremental Object Detection via Elastic Response Distillation. In *CVPR*.

Gao, Z.; Cen, J.; and Chang, X. 2024. Consistent prompting for rehearsal-free continual learning. In *CVPR*.

Hadsell, R.; Rao, D.; Rusu, A. A.; and Pascanu, R. 2020. Embracing change: Continual learning in deep neural networks. *Trends in cognitive sciences*, 24(12): 1028–1040.

Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-Efficient Transfer Learning for NLP. In *International Conference on Machine Learning (ICLR)*, 2790–2799. PMLR.

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

Huang, L.; Cao, X.; Lu, H.; and Liu, X. 2024. Class-incremental learning with clip: Adaptive representation adjustment and parameter fusion. In *ECCV*, 214–231. Springer.

Hung, C.-Y.; Tu, C.-H.; Wu, C.-E.; Chen, C.-H.; Chan, Y.-M.; and Chen, C.-S. 2019. Compacting, Picking and Growing for Unforgetting Continual Learning. *Advances in Neural Information Processing Systems*, 32.

Jha, S.; Gong, D.; and Yao, L. 2024. Clap4clip: Continual learning with probabilistic finetuning for vision-language models. *NeurIPS*.

Kang, B.; Wang, L.; Wu, Z.; Feng, T.; Li, Y.; Gao, Y.; and Li, W. 2025. Dynamic Multi-Layer Null Space Projection for Vision-Language Continual Learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2077–2086.

Keskar, N. S.; Mudigere, D.; Nocedal, J.; Smelyanskiy, M.; and Tang, P. T. P. 2017. On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima. In *ICLR*.

Li, W.; Feng, T.; Yuan, H.; Bian, A.; Du, G.; Liang, S.; Gan, J.; and Liu, Z. 2024. UniGrad-FS: Unified Gradient Projection With Flatter Sharpness for Continual Learning. *IEEE Transactions on Industrial Informatics*.

Liang, V. W.; Zhang, Y.; Kwon, Y.; Yeung, S.; and Zou, J. Y. 2022. Mind the Gap: Understanding the modality gap in multi-modal contrastive representation learning. *NeurIPS*.

Lin, S.; Yang, L.; Fan, D.; and Zhang, J. 2022. TRGP: Trust Region Gradient Projection for Continual Learning. In *International Conference on Learning Representations (ICLR)*.

Lopez-Paz, D.; and Ranzato, M. 2017. Gradient Episodic Memory for Continual Learning. *NeurIPS*, 30.

Lu, A.; Yuan, H.; Feng, T.; and Sun, Y. 2025. Rethinking the Stability-Plasticity Trade-off in Continual Learning from an Architectural Perspective.

Malladi, S.; Gao, T.; Nichani, E.; Damian, A.; Lee, J. D.; Chen, D.; and Arora, S. 2023. Fine-tuning language models with just forward passes. *NeurIPS*.

Möllenhoff, T.; and Khan, M. E. 2023. SAM as an Optimal Relaxation of Bayes. In *ICLR*.

Nesterov, Y.; and Spokoiny, V. 2017. Random Gradient-Free Minimization of Convex Functions. *Foundations of Computational Mathematics*, 17(2): 527–566.

Ni, Z.; Wei, L.; Tang, S.; Zhuang, Y.; and Tian, Q. 2023. Continual vision-language representation learning with off-diagonal information. In *International Conference on Machine Learning*, 26129–26149. PMLR.

Peng, X.; Wei, Y.; Deng, A.; Wang, D.; and Hu, D. 2022. Balanced multimodal learning via on-the-fly gradient modulation. In *CVPR*, 8238–8247.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*.

Rebuffi, S.-A.; Kolesnikov, A.; Sperl, G.; and Lampert, C. H. 2017. ICARL: Incremental classifier and representation learning. In *CVPR*.

Ruder, S. 2016. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

Saha, G.; Garg, I.; and Roy, K. 2021. Gradient Projection Memory for Continual Learning. In *International Conference on Learning Representations (ICLR)*.

Shi, G.; Chen, J.; Zhang, W.; Zhan, L.-M.; and Wu, X.-M. 2021. Overcoming Catastrophic Forgetting in Incremental Few-Shot Learning by Finding Flat Minima. *Advances in neural information processing systems*, 34: 6747–6761.

Sun, F.; Liu, H.; Yang, C.; and Fang, B. 2020. Multimodal continual learning using online dictionary updating. *IEEE Transactions on Cognitive and Developmental Systems*, 13(1): 171–178.

Talaei, S.; Ansaripour, M.; Nadiradze, G.; and Alistarh, D. 2025. Hybrid Decentralized Optimization: Leveraging Both First-and Zeroth-Order Optimizers for Faster Convergence. In *AAAI*.

Thengane, V.; Khan, S.; Hayat, M.; and Khan, F. 2022. Clip Model is an Efficient Continual Learner. *arXiv preprint arXiv:2210.03114*.

Wang, Z.; Zhang, Z.; Ebrahimi, S.; Sun, R.; Zhang, H.; Lee, C.-Y.; Ren, X.; Su, G.; Perot, V.; Dy, J.; et al. 2022a. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *ECCV*.

Wang, Z.; Zhang, Z.; Lee, C.-Y.; Zhang, H.; Sun, R.; Ren, X.; Su, G.; Perot, V.; Dy, J.; and Pfister, T. 2022b. Learning to prompt for continual learning. In *CVPR*.

Yu, J.; Zhuge, Y.; Zhang, L.; Hu, P.; Wang, D.; Lu, H.; and He, Y. 2024. Boosting Continual Learning of Vision-Language Models via Mixture-of-Experts Adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 23219–23230.

Zhang, D.; Feng, T.; Xue, L.; Wang, Y.; Dong, Y.; and Tang, J. 2025a. Parameter-Efficient Fine-Tuning for Foundation Models. arXiv:2501.13787.

Zhang, Z.; Yang, Y.; Zhen, K.; Susanj, N.; Mouchtaris, A.; Kunzmann, S.; and Zhang, Z. 2025b. MaZO: Masked Zeroth-Order Optimization for Multi-Task Fine-Tuning of Large Language Models. *arXiv preprint arXiv:2502.11513*.

Zhao, Z.; Bai, H.; Zhang, J.; Zhang, Y.; Xu, S.; Lin, Z.; Timofte, R.; and Van Gool, L. 2023. CDDFuse: Correlation-Driven Dual-Branch Feature Decomposition for Multi-Modality Image Fusion. In *CVPR*, 5906–5916.