Cloud-Device Collaborative Learning for Multimodal Large Language Models

Guanqun Wang[1*] Jiaming Liu[1*] Chenxuan Li[1*] Yuan Zhang[1] Junpeng Ma[1] Xinyu Wei[1] Kevin Zhang[1]
Maurice Chong[1] Renrui Zhang[2] Yijiang Liu[3] Shanghang Zhang[1†]
[1]National Key Laboratory for Multimedia Information Processing, School of Computer Science,
Peking University [2]Shanghai AI Lab [3]Nanjing University

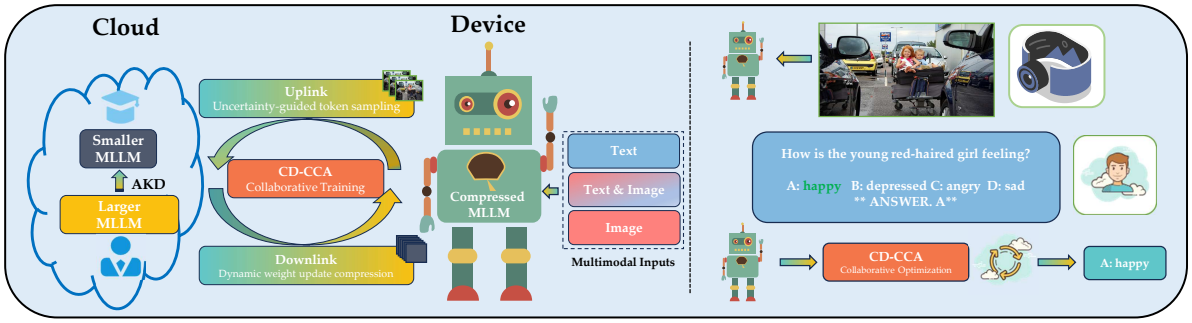

Figure 1. **Cloud-Device Collaborative Continual Adaptation framework (CD-CCA).** Our CD-CCA, specifically designed for MLLMs, embodies a cloud-device collaborative paradigm. It is adept at receiving various modalities and executing multimodal comprehension tasks. As illustrated on the left side of the figure, our approach facilitates collaborative learning between device and cloud, enabling the update on the device-side deployed MLLM to adapt to dynamically changing scenarios. On the right side, an application instance of CD-CCA is depicted, demonstrating its capability to achieve accurate multimodal comprehension in the face of evolving scenarios at the device.

Abstract

The burgeoning field of Multimodal Large Language Models (MLLMs) has exhibited remarkable performance in diverse tasks such as captioning, commonsense reasoning, and visual scene understanding. However, the deployment of these large-scale MLLMs on client devices is hindered by their extensive model parameters, leading to a notable decline in generalization capabilities when these models are compressed for device deployment. Addressing this challenge, we introduce a Cloud-Device Collaborative Continual Adaptation framework, designed to enhance the performance of compressed, device-deployed MLLMs by leveraging the robust capabilities of cloud-based, larger-scale MLLMs. Our framework is structured into three key components: a device-to-cloud uplink for efficient data transmission, cloud-based knowledge adaptation, and an optimized cloud-to-device downlink for model deployment. In the uplink phase, we employ an Uncertainty-guided Token Sampling (UTS) strategy to effectively filter out-of-distribution tokens, thereby reducing transmission costs and improving training efficiency. On the cloud side, we propose Adapter-based Knowledge Distillation (AKD) method to transfer refined knowledge from large-scale to compressed, pocket-size MLLMs. Furthermore, we propose a Dynamic Weight update Compression (DWC) strategy for the downlink, which adaptively selects and quantizes updated weight parameters, enhancing transmission efficiency and reducing the representational disparity between cloud and device models. Extensive experiments on several multimodal benchmarks demonstrate the superiority of our proposed framework over prior Knowledge Distillation and device-cloud collaboration methods. Notably, we also validate the feasibility of our approach to real-world experiments.

1. Introduction

In recent years, we have witnessed a proliferation of Large Language Models (LLMs) and Multimodal Large Language Models (MLLMs), with models like GPT4 [1] demonstrating exceptional performance across various tasks, including visual question answering (VQA) and commonsense reasoning. These MLLMs, such as Flamingo [2] and BLIP-2 [3], empower LLMs with the capability to comprehend and reason about visual scenes. Due to their large amount of parameters, MLLMs are commonly deployed on

*These authors contributed equally to this work.
†Corresponding Author E-mail: shanghang@pku.edu.cn

cloud servers, demonstrating strong generalization capability. However, their large-scale parameters make it challenging to directly deploy MLLMs on the device, which also limits their practicality.

Since the client device is resource-constrained, MLLMs need to be compressed for the deployment on the device. The compressed MLLMs indeed demonstrate remarkable performance when the test data distribution closely matches the training data distribution. However, this assumption encounters significant challenges in real-world scenarios, where non-static environments and distribution shifts are prevalent [4, 5]. The small-size MLLMs are susceptible to severe performance degradation when confronted with dynamic distribution shifts [5–7]. There are two principal challenges: (1) The limited computational capacity of edge devices hinders the ability to perform timely model updates, leading to performance decay when encountering distribution shifts. (2) Compressed models, which have a relatively small capacity, struggle to adapt to continuously changing environments, leading to insufficient generalization ability.

To empower device models in dynamic environments, we propose a Cloud-Device Collaborative Continual Adaptation (CD-CCA) framework for MLLMs (Figure 1). Our key insight is harnessing cloud-larger MLLMs to boost the generalization capability of smaller, compressed MLLMs deployed on the device. In pursuit of augmenting the generalization capabilities of device models without compromising their efficiency, as well as facilitating their dynamic adaptability to ever-changing distributions, we propose a new learning paradigm: Cloud-Device Collaborative Continual Adaptation. The paradigm has three key components: the device-to-cloud uplink, the cloud-side knowledge update, and the cloud-to-device downlink.

First, in order to enable the MLLM deployed on devices to have the capability of dynamic parameter updating, we design a device-to-cloud uplink for transmitting uncertainty tokens generated on the device side. Specifically, we propose a coarse-to-fine token filtering approach known as the Uncertainty-guided Token Sampling (UTS) strategy to minimize upstream transmission costs. We begin by utilizing sample-level uncertainty to identify and filter out corner case samples from the target distribution data. Subsequently, we adopt token-level uncertainty to perform a secondary filtering process, isolating out-of-distribution tokens. This approach helps alleviate network transmission bandwidth constraints and enhances training efficiency on the cloud server.

Second, on the cloud side, we develop a novel Adapter-based Knowledge Distillation (AKD) method, specially designed for MLLMs. The purpose of AKD is to transfer dark knowledge from the original huge MLLMs to the compressed pocket-size MLLMs. MLLMs typically consist of three main components: a vision encoder, a large language model (LLM) [8], and a cross-modal transformer, which fuses the high-level vision and language context [2, 3, 9]. Therefore, our approach initially focuses on conducting KD for the learnable query adapter from the cross-modal transformer, enhancing the small MLLMs' vision-to-text alignment capabilities. Simultaneously, since the LLM occupies the majority of parameters in MLLM, the primary objective for the compressed model is to reduce the LLM's parameter. Consequently, we further conduct KD for learnable language adapters, which are plugged into the LLM, to enhance the student MLLMs' language communication and reasoning abilities.

Furthermore, to account for the varying computational capabilities of edge devices, we employ an adaptive quantization and compression technique for the dynamically updated weight parameters for the device-side MLLMs. These compressed weight parameters are then transmitted to the device through the downlink, narrowing the gap in representation between the device and cloud MLLMs. We conducted extensive experiments on two cross-domain visual reasoning benchmarks, one from VQA-v2 [10] to A-OKVQA [11] and the other from COCO Captions 2017 [12] to nocaps [13]. Our proposed framework achieved superior performance compared to previous methods. Additionally, for the uplink, we maintain the performance while reducing transmission costs to 4.71% and 20.6% compared to transferring the entire dataset. As for the downlink, we can deliver the compressed dynamically updated weight parameters with almost negligible transmission cost to the device, resulting in 3.93% and 2.20% improvements in domain-shifted VQA tasks and captioning tasks. Our contributions can be summarized as follows:

- We introduce the CD-CCA framework that involves the continuous utilization of cloud-based large MLLMs to enhance the generalization capabilities of smaller, compressed MLLMs on the device.
- For the device-to-cloud uplink, we propose UTS strategy, which serves to filter out-of-distribution tokens during data transmission from the device to the cloud.
- On the cloud side, we introduce the AKD manner to facilitate the transfer of dark knowledge from the original huge MLLMs to the compressed pocket-size MLLMs.
- For the cloud-to-device downlink, we propose a dynamic weight updating compression method that significantly enhances the transmission efficiency of updated weights from cloud to device, which establishes a practical foundation for the application of the Cloud-Device collaborative learning paradigm.
- Extensive experiments demonstrate CD-CCA outperforms previous methods, effectively enhancing the continuous domain adaptation capability of device-compressed MLLMs. Moreover, we validate the feasibility of our approach through real-world experiments.

2. Related Work

MLLMs. Recent advancements in LLMs[8, 14] have marked a shift from single to multi-modal capabilities, with MLLMs [2, 3, 15] emerging as a significant development. However, this expansion has led to increased model sizes, escalating training costs to prohibitive levels. Despite the efforts to minimize the trainable parameters [16], model deployment on device continues to pose significant challenges, constrained by limited computational power and network bandwidth. In this work, we conceive a new training strategies to replicate the magic of large models in resource-constrained environments.

Cloud-Device Collaborative Learning. Previous approaches have attempted to offload the computational workload to the cloud [17–21], effectively reducing the hardware requirements on devices. However, these methods usually represent a superficial level of cloud-device collaboration. Our method introduces the UTS strategy, designed to filter out-of-distribution image tokens from devices to cloud, which significantly reduces the required upstream bandwidth while ensuring that the selected image tokens are rich in semantic information. Knowledge Distillation (KD) methods have been proposed that perform distillation over intermediate features [22, 23], relation representation [24–26], attention [27, 28]. However, for MLLMs, there is currently no specific knowledge distillation method available to compress them effectively.

Continual Domain Adaptation. Devices are commonly deployed in real-world scenarios where data is continuously evolving. In recent years, several works have been proposed to continually adapt the model to the changing target domain [29–32]. Our work proposes a Cloud-Device Collaborative Continual Adaptation framework, enabling the model to adapt to dynamically changing distributions. This approach allows for the simultaneous improvement of the teacher model in cloud and student model on devices.

3. Approach

In this section, we propose CD-CCA to enhance device-deployed MLLMs through efficient cloud-device collaboration. We describe the overall pipeline in Sec. 3.1, and then introduce the key components in the following subsections.

3.1. Overview of CD-CCA Framework

In the landscape of pervasive computing, edge devices are increasingly tasked with complex multimodal interactions, necessitating models that are not only robust but also adaptive to continual environmental shifts. The CD-CCA framework, shown in Figure 2, emerges as a paradigm designed to synergize the computational prowess of cloud resources with the operational nimbleness of edge devices. This dynamic adaptability of the CD-CCA framework can be succinctly encapsulated in the following optimization process:

$$\mathcal{M}' = C\left(K\left(U\left(\mathcal{D}, \mathcal{M}_{\text{edge}}\right), \mathcal{M}_{\text{cloud}}^{\text{teacher}}\right), \mathcal{M}_{\text{cloud}}^{\text{student}}\right) \quad (1)$$

where $\mathcal{M}'$ signifies the refined model deployed back on the edge device, $\mathcal{D}$ represents the dataset of multimodal instances, U delineates the UTS for uplink efficiency, K depicts the AKD on the cloud, and C denotes the Dynamic Weight update Compression (DWC) for the downlink transmission.

Initially, the framework employs UTS, a novel approach that discernibly filters the influx of multimodal data, earmarking only the most pivotal tokens for cloud-assisted refinement. The selective process is pivotal in distilling the essence of data that demands the cloud's attention, thereby conserving bandwidth and reducing uplink latency. Subsequently, the framework leverages an AKD technique in the cloud, which distills and transfers the rich knowledge from an expansive teacher model to a compact student counterpart. The AKD process is fine-tuned to cater to the specific learning nuances of multimodal data, ensuring that the student model is endowed with enhanced generalization capabilities. Culminating the framework's process is DWC, an innovative strategy that dynamically quantizes and compresses the updated model parameters before transmission via the downlink, significantly alleviating the latency typically associated with updating device-resident models. The DWC ensures that the updated intelligence is delivered promptly, maintaining the real-time responsiveness crucial for device applications. Collectively, these components of the CD-CCA framework constitute a powerful conduit for continual learning, enabling MLLMs to evolve in situ, with a level of acuity and efficiency previously unattainable in device computing paradigms.

3.2. Uncertainty-guided Token Sampling (UTS)

As devices operate within the intrinsic variability of real-world scenarios, there is a crucial need for the continual adaptation of MLLMs that can process data selectively, concentrating computational efforts where they are most needed. To this end, the UTS component of the CD-CCA framework serves as an intelligent filtration mechanism, enabling the discernment and prioritization of multimodal instances for transmission. This is rooted in the understanding that not every instance contributes equally to the model's learning and that some may be more pivotal for adaptation.

In the first stage of UTS, an MLLM with parameters Θ deployed on an edge device processes a multimodal instance $(v_i, t_i) \in \mathcal{D}$, and its predictive uncertainty $\mathcal{U}$ is evaluated as follows:

$$\mathcal{U}(v_i, t_i; \Theta) = -\sum_j p(y_{ij}|v_i, t_i; \Theta) \log p(y_{ij}|v_i, t_i; \Theta) \quad (2)$$

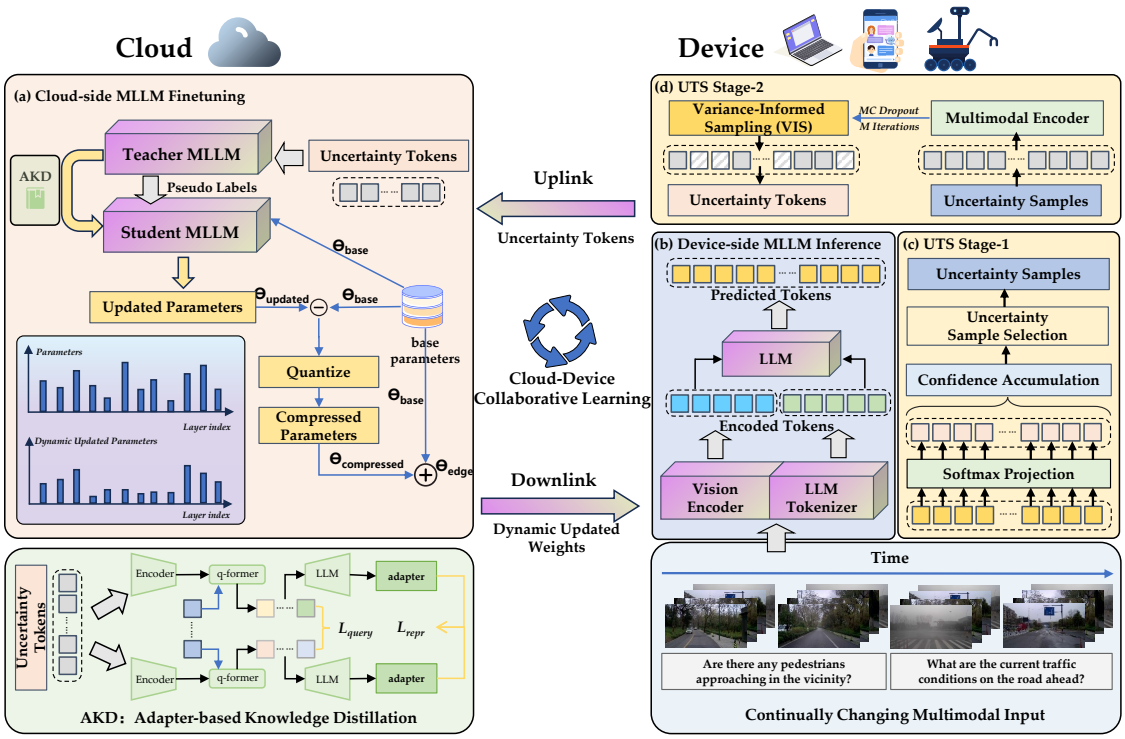

Figure 2. **The overall pipeline of CD-CCA.** (a) Cloud: Upon receiving a token from the device, the Teacher MLLM generates pseudo labels and distills knowledge for the smaller model. (b) Device: Upon receiving an image and a human prompt, it generates the corresponding answer. (c) First stage of Uncertainty-guided Token Sampling (UTS). (d) Second stage of UTS.

Eq. 2 calculates the entropy of the predicted token probabilities, which serves as a measure of uncertainty for the given instance. Instances with high uncertainty are flagged as candidates for further analysis.

In the subsequent phase, we propose Variance-Informed Sampling (VIS) technique as a refinement step to further sift through the pre-selected instances. VIS applies Monte Carlo dropout to the encoded multimodal input tensors, deriving a variance measure across multiple forward passes to identify which tokens within these instances exhibit significant variability in their representations:

$$\sigma^2(v_i, t_i; \Theta) = \frac{1}{M} \sum_{m=1}^{M} \left(\mathcal{F}_m(v_i, t_i; \Theta) - \bar{\mathcal{F}}(v_i, t_i; \Theta) \right)^2 \tag{3}$$

Here, tokens with a variance σ^2 exceeding a predefined threshold β are retained, ensuring that only the most informative tokens are considered for cloud processing, as shown in Eq. 4:

$$\tau(\sigma^2(v_i, t_i; \Theta), \beta) = \begin{cases} 1, & \text{if } \sigma^2(v_i, t_i; \Theta) > \beta \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

By implementing this two-stage approach, UTS significantly reduces the volume of data required for uplink transmission, thereby optimizing bandwidth usage and minimizing latency. The VIS, in particular, plays a critical role by ensuring that the model's enhancement is driven by data points that are likely to contribute the most to its learning progress, embodying the essence of targeted and efficient learning within the CD-CCA framework.

3.3. Adapter-Based Knowledge Distillation (AKD)

The AKD strategy hones the capabilities of device-deployed MLLMs by leveraging the computational abundance of cloud resources. In this process, a high-capacity teacher MLLM and a structurally identical student MLLM coexist on the cloud, engaging in a targeted knowledge transfer. This exchange is facilitated by adapters—auxiliary linear layers that introduce minimal parameters to the model while providing pathways for significant updates.

During the AKD phase, we focus on fine-tuning the student model $\mathcal{M}_{\text{student}}$ to encapsulate the high-level multimodal comprehension exhibited by the teacher model $\mathcal{M}_{\text{teacher}}$. Specifically, the adapters are employed to fine-tune the query representations and the cross-attention outputs, which are critical for processing and integrating multimodal information. These adapters act as targeted modification modules, aligning the student's latent space with the teacher's refined feature space, effectively compressing

the teacher's extensive knowledge into the student's more concise structure.

This fine-grained distillation process is facilitated through adapters that are strategically placed to intercept and transform the query vectors and the attention-mediated multimodal representations. By so doing, the adapters enable a direct knowledge flow from the teacher's rich feature space to the student's corresponding layers, ensuring the retention of critical multimodal insights.

The effectiveness of this adapter-based fine-tuning is measured by a composite loss function, comprising:

Query Alignment Loss ($\mathcal{L}_{query}$): Minimizes the difference between the query representations of the student and teacher models, thereby ensuring that the student can generate queries that encapsulate the complexity of the multimodal data as effectively as the teacher. Regularly, $\boldsymbol{Q}^{(t)} \in \mathbb{R}^{B\times L\times C}$ and $\boldsymbol{Q}^{(s)} \in \mathbb{R}^{B\times L\times C_s}$ denote the feature maps of teacher and student queries respectively, and the Query Alignment imitation can be fulfilled via:

$$\mathcal{L}_{\text{query}} = \frac{1}{BLC}\left\|\boldsymbol{Q}^{(t)} - \phi(\boldsymbol{Q}^{(s)})\right\|_2^2, \tag{5}$$

where ϕ is a linear projection layer to adapt $\boldsymbol{Q}^{(s)}$ to the same channels as $\boldsymbol{Q}^{(t)}$.

Representation Alignment Loss ($\mathcal{L}_{repr}$): Aims to synchronize the attention-driven multimodal representations between the student and teacher models, enhancing the student's ability to process and integrate multimodal cues.

Cross-Entropy Loss ($\mathcal{L}_{CE}$): Utilizes the teacher model's output on challenging multimodal instances, which have been identified and transmitted via the uplink after UTS, as pseudo-labels. These labels serve to calibrate the student model's parameter updates, enhancing its capacity to address the complexities inherent in multimodal data. The inclusion of UTS-selected instances ensures that the student model focuses its learning on the data points that are most indicative of its current limitations, thereby promoting a more efficient and targeted learning process.

The distillation procedure optimizes a weighted sum of these loss components, carefully calibrated to achieve a harmonious balance between mimicking the teacher's output and maintaining the student's intrinsic characteristics:

$$\mathcal{L}_{total} = \lambda_{query}\mathcal{L}_{query} + \lambda_{repr}\mathcal{L}_{repr} + \lambda_{CE}\mathcal{L}_{CE} \tag{6}$$

By minimizing $\mathcal{L}_{total}$, AKD ensures that the student MLLM not only accurately reflects the teacher's adeptness in handling multimodal data but also remains agile and efficient, key for deployment within the resource-constrained environments typical of device computing.

3.4. Dynamic Weight update Compression (DWC)

DWC forms an integral pillar of the CD-CCA framework, addressing the transmission efficiency of model updates from cloud to device. DWC specifically targets the challenge of bandwidth constraints and latency in updating device-deployed MLLM by introducing a quantization-based compression mechanism for model parameters.

DWC operates on the premise that efficient model updates are not solely contingent on the volume of data transmitted but also on the significance of the parameters updated. This leads to the development of a quantization scheme that selectively targets the parameters refined during the AKD phase, optimizing the update payload for transmission efficiency without compromising the model's performance integrity.

The DWC process can be formalized through the following quantization operation:

$$\Theta_{\text{compressed}} = \text{Quantize}(\Theta_{\text{updated}} - \Theta_{\text{base}}, \mathcal{Q}) \tag{7}$$

Here, Θ_{updated} represents the parameters post-AKD, Θ_{base} denotes the pre-update baseline parameters, and $\mathcal{Q}$ is the quantization function that adaptively maps parameters to a compact, lower-bit representation. This function is meticulously calibrated to ensure that the most critical updates are preserved, while the overall update size is reduced.

The quantization process strategically applies a higher compression ratio to less impactful parameters, while preserving the fidelity of more significant updates:

$$\Theta_{\text{edge}} = \Theta_{\text{base}} + \Theta_{\text{compressed}} \tag{8}$$

The edge device, upon receiving $\Theta_{\text{compressed}}$, integrates these updates directly into the MLLM. This direct integration circumvents the need for dequantization, as the device MLLM operates effectively within the quantized parameter space, reflecting the nuanced enhancements learned through cloud-based distillation.

DWC thus enables a practical and scalable approach to model updating in device computing environments, where transmission overhead is a critical concern. By facilitating smaller, yet impactful updates, DWC ensures that the device-deployed MLLMs can continually evolve and adapt to new data without the latency typically associated with large-scale model retraining or full-model updates.

3.5. Collaborative Learning Strategy

The essence of CD-CCA resides in its Collaborative Learning Strategy, a synergistic approach that harmonizes the model refinement process across cloud and device platforms, shown in Algorithm 1. This strategy encapsulates the concerted efforts of edge devices and cloud services to perpetually enhance the MLLMs seamlessly and efficiently. The optimization pivots on two key fronts: the edge devices perform UTS to identify and forward challenging multimodal instances to the cloud, while the cloud engages in AKD and DWC to refine and compress the parameter updates, respectively. The culmination of this process is

Algorithm 1 Collaborative Learning in CD-CCA

Initialize edge model $\mathcal{M}_{\text{edge}}$ with parameters Θ_{edge}
Deploy teacher model $\mathcal{M}_{\text{teacher}}$ and student model $\mathcal{M}_{\text{student}}$ on cloud
Define UTS, AKD, and DWC procedures
repeat
    Edge performs inference and UTS to identify high-uncertainty instances
    Transmit selected instances to cloud
    Cloud performs AKD, utilizing $\mathcal{M}_{\text{teacher}}$ to refine $\mathcal{M}_{\text{student}}$
    Compress updated parameters Θ_{updated} using DWC to obtain $\Theta_{\text{compressed}}$
    Transmit $\Theta_{\text{compressed}}$ back to device
    Update $\mathcal{M}_{\text{edge}}$ with $\Theta_{\text{compressed}}$
until convergence or a predefined number of cycles are completed

the application of compressed updates to the device-side MLLM, ensuring it remains adept and up-to-date with minimal transmission overhead. The Collaborative Learning Strategy is a testament to the potential of CD-CCA in fostering a dynamic learning environment where edge-deployed MLLMs can thrive. By leveraging the strengths of both cloud and device computing, it stands as a paradigmatic shift towards more intelligent and adaptable multimodal interactions in real-world applications.

4. Experiments

4.1. Experimental Setups

Datasets. To validate the persistent generalization ability of our proposed CD-CCA for multimodal large language model (MLLM) in the scenario of language domain-shifted distribution, we conducted experiments based on two pairs of datasets, VQA-v2 [10], A-OKVQA [11]. and COCO Caption 2017 [12], Nocaps [13].

Evaluation Metrics. To demonstrate the MLLM's persistent generalization capability under the proposed CD-CCA and other SOTA domain adaptation methods, VQA Accuracy, BLeU-4, and CIDEr scores are uniformly used as the evaluation metrics. In addition, in real-world validations, we further calculate the quantity of transmitted parameters and data size in the uplink and downlink of CD-CCA, as well as the Cloud-Device transfer delay (TD), respectively.

Implementation Details. In our experiments, we use LLaMA-Adapter [33] with LLaMA2-13B [8] as the large teacher MLLM on the cloud, and we employ LLaMA-Adapter [33] with LLaMA2-7B [8] as the small student MLLM (same as the device model). In addition, to further reduce the quantity of device-side model parameters, we reduce the student MLLM's Q-former [34] hidden layers, from 12 to 6. The above MLLMs are first pre-trained on large-scale image-text pairs: COYO [35], LAION [36], CC3M [37], CC12M [38], SBU [39]. Then, they are further tuned with 52K single-turn instruction data from GPT4-LLM [40] and 567K captioning data from COCO Caption [12]. For both cloud and device models, all the parameters in LLaMA normalization layers, linear layer bias, LoRA [41], and query tokens in Q-Former [34] are set to be updated during finetuning with the remaining parameters kept frozen. In the specific experiments, we further finetuned the MLLMs on the corresponding datasets elaborated before.

4.2. Comparison Analysis

In this subsection, we conduct comparison experiments between our CD-CCA and the existing SOTA domain adaptation methods [4, 5, 42, 43].Tent[4] updates the trainable parameters in the Batchnorm layer to adapt to the test data by minimizing entropy. Cotta[5] employs weight-averaged and augmentation-averaged predictions to reduce the accumulation of errors in pseudo-labeling and utilizes stochastically restore to prevent the issue of catastrophic forgetting. PKD[42] utilizes feature imitation based on the Pearson Correlation Coefficient, relaxing constraints on the magnitude of the features while focusing on the relationship information from the teacher. ChannelWiseDivergence[43] normalizes the activation maps of each channel, yielding soft probability maps for the two networks, and minimizes the Kullback-Leibler divergence between the channel probability maps. All the experiments are carried out using LLaMA-Adapter [33] as the underlying MLLM. First, to verify the persistent generalization ability of our proposed CD-CCA under the condition of language domain-shifted distribution, we use the VQAv2-to-AOKVQA datasets for evaluation. Specifically, we adopt VQA-v2 [10] to finetune the pre-trained MLLM, LLaMA-Adapter (7B & 13B). Then, the VQA accuracy results on A-OKVQA [11] under different conditions (multiple choices (MC) & direct answers (DA)) are evaluated and recorded in Table 1 and

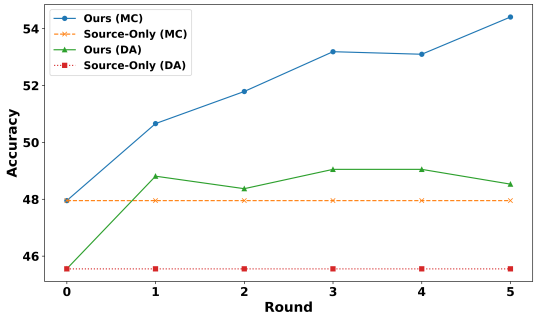

Figure 3. Comparative analysis of CD-CCA and source-only method. The MC and DA accuracy are evaluated over five rounds.

Table 1. **Persistent generalization capability on VQAv2-to-AOKVQA.** Visual question-answering results are evaluated on the VQAv2-to-AOKVQA online continual adaptation task. MC and DA are VQA accuracy (%) calculated following [11] under different conditions (multiple choices and direct answers). Gain (%) refers to the accuracy improvement compared with the source-only method.

Time	$t \longrightarrow$									
Round	1_{st}		2_{nd}		3_{rd}		All			
Condition	MC	DA	MC	DA	MC	DA	Mean_{MC}	Mean_{DA}	Gain_{MC}	Gain_{DA}
Source-only [33]	47.95	45.55	47.95	45.55	47.95	45.55	47.95	45.55	/	/
TENT-continual [4]	47.42	45.17	48.12	45.52	47.34	44.86	47.63	45.18	-0.32	-0.37
CoTTA [5]	47.77	45.02	47.77	45.30	48.30	45.02	47.95	45.11	+0.00	-0.44
PKD [42]	48.21	45.05	48.73	45.24	47.77	45.24	48.23	45.18	+0.28	-0.37
ChannelWiseDivergence [43]	48.03	44.78	48.21	44.65	48.47	44.93	48.24	44.79	+0.29	-0.76
Ours (CD-CCA)	**50.65**	**48.80**	**51.79**	**48.37**	**53.19**	**49.05**	**51.88**	**48.74**	**+3.93**	**+3.19**

Table 2. **Persistent generalization capability on COCO-to-nocaps.** Image captioning results are evaluated on the COCO-to-nocaps online continual adaptation task. BLeU@4, CIDEr scores (%) are calculated following [12] under different conditions (in-domain, near-domain, out-domain, etc.). Gain (%) refers to the improvement compared with the source-only method.

Condition	In-domain		Near-domain		Out-domain		All		Gain	
Score	BLeU	CIDEr	BLeU	CIDEr	BLeU	CIDEr	BLeU	CIDEr	BLeU	CIDEr
Source-only [33]	39.55	72.33	39.72	77.32	31.20	76.95	36.82	75.53	/	/
TENT-continual [4]	39.92	71.81	39.60	74.49	30.28	72.69	36.60	73.00	-0.22	-2.53
CoTTA [5]	40.12	73.87	40.08	76.52	30.04	74.19	36.74	74.86	-0.08	-1.34
PKD [42]	39.43	73.67	39.46	76.33	31.12	76.88	36.67	75.63	-0.15	+0.10
ChannelWiseDivergence [43]	39.03	73.82	39.10	75.87	30.15	75.77	36.18	75.15	-0.64	-0.38
Ours (CD-CCA)	**41.34**	**74.47**	**40.67**	**77.78**	**33.04**	**80.93**	**38.35**	**77.73**	**+1.53**	**+2.20**

Figure 3. In the VQA task, our CD-CCA framework of 1-round scenario has already surpassed the highest accuracy of the comparative models both in MC and DA questions. Notably, we observe that previous methods sometimes lead to performance deterioration. We attribute this to the fact that most of the previous methods were not specifically designed for MLLM, as the model parameter size increases, methods like CoTTA and Tent tend to exhibit a decrease in performance. In contrast, our approach is specifically designed for MLLM, as shown in Table 2, our accuracy is higher by 3.64% (MC) and 3.19% (DA) compared to the best-performing comparative model on average. This strongly demonstrates that our framework can maintain a high level of accuracy when faced with constantly changing data distributions. Figure 4 visually illustrates the experimental results on multimodal comprehension of our proposed framework.

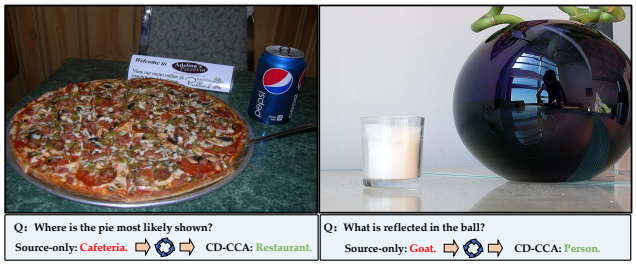

Figure 4. **Visual results of CD-CCA.** The figure demonstrates the improvement in visual reasoning of device-deployed MLLM facilitated by CD-CCA. 'Source-only' refers to MLLM deployed on the device side without undergoing Cloud-Device Learning.

Second, we use the COCO-to-nocaps datasets for further evaluation. We finetune the pre-trained LLaMA-Adapter (7B & 13B) on the COCO Captions 2017 dataset [12]. Then, the visual caption results (BLeU@4, CIDEr) on nocaps [13] are evaluated and recorded in Table 2. Based on the overlap of the training-test image categories, following reference [13], the test images are categorized into in-domain, near-domain, and out-domain cases.

In the image caption task, our framework significantly outperforms the best comparative methods in all cases. In the In-domain and Near-domain tasks, our framework surpasses the best comparative method by 1.22% and 0.59% (BLeU), 0.6%, and 0.46% (CIDEr) respectively. In the out-domain task, our CD-CCA's superiority is even more pronounced, with 1.84% (BLeU) and 3.98 %(CIDEr). This reflects the strong generalization ability of our CD-CCA framework, which can effectively help the model extract intrinsic knowledge from images and understand them when transferring to new tasks. Moreover, the experimental results in Table 2 reaffirm that previous methods sometimes

do not apply to MLLM, while our CD-CCA consistently improves performance. This further reflects the effectiveness of our method specifically designed for MLLM.

Table 3. **Ablation studies.** We conduct experiments on VQAv2-to-AOKVQA. PL refers to Pseudo Labels. UTS-1 and UTS-2 represent the first and second stage in UTS, respectively.

PL	AKD	UTS-1	UTS-2	MC	DA	Gain_{MC}	Gain_{DA}
				47.95	45.55	/	/
✓				50.48	48.89	2.53	3.34
✓	✓			50.39	**49.05**	2.44	**3.50**
✓	✓	✓		50.82	48.93	2.87	3.38
✓	✓	✓	✓	**53.19**	**49.05**	**5.24**	**3.50**

Table 4. **Performance (MC/DA) comparison in UTS (Token-Level)**. We report VQA score (MC/DA) using different token sampling strategies. Optimal performance is obtained using the uncertainty token guided sampling (UTS) strategy in CD-CCA.

	25%		50%		75%	
	MC	DA	MC	DA	MC	DA
Random	50.74	49.46	50.13	48.40	50.91	49.09
UTS	52.49	48.92	53.19	49.05	53.19	48.96
Gain	+1.75	-0.54	+3.06	+0.65	+2.28	-0.13

4.3. Ablation Studies

In this section, we meticulously dissected the proposed CD-CCA framework and its performance across various test scenarios. To gain granular insights into the individual contributions of various components to the framework's efficacy, we systematically dismantle key components.

Effectiveness of UTS strategy. Our UTS strategy effectively reduces transmission costs while maintaining performance, as shown in Table 5, we achieve the same performance with only 0.21% in transmission data volume and 0.20% in transfer latency compared to transmitting the entire dataset. To further validate the effectiveness of UTS, we explore VQA results at different mask ratios, as shown in Table 4. The model performs best when the mask ratio is set at 50%. Specifically, we achieved a notable increase of 3.06% in the accuracy of MC and a 0.65% increase in the accuracy of DA. Furthermore, we also investigate the effectiveness of each stage in UTS, as shown in Table 3, and the results indicate that each stage of UTS contributes to improving the performance of the model. When both stages are used in conjunction, there is a significant improvement of 5.24% and 3.50% in the MC and DA problems.

Effectiveness of Cloud-device joint optimization with AKD. our proposed AKD strategy utilizes adapters for targeted knowledge transfer between teacher and student models, enhancing the generalization ability of the student model. As shown in Table 3, compared to the pure pseudo-labeling method, AKD improves performance by 2.53% (MC) and 3.34% (DA) in VQA tasks, while combining AKD with other modules further enhances performance steadily. The model parameters obtained after AKD are further quantitatively compressed through the DWC method.

Effectiveness of DWC. The DWC strategy in the cloud aims to quantitatively compress model parameters, ensuring that only the most effective parameters are updated on the device. This alleviates the performance burden on the device and effectively reduces the amount of data transmitted to the device during downlink. Here, we utilize the widely adopted 4-bit NormalFloat quantization, QLora [44], as the basic quantization function. As shown in Table 5, compared to no processing, our approach significantly reduces the weight parameter quantity, data quantity, and transmission latency of the model transmitted to the device, by 99.98%, 99.99%, and 99.98% respectively. This effectively guarantees real-time updates of device parameters.

Table 5. **Validation of transmission parameters in real machine**. We report a quantitative analysis of bidirectional transmission parameters size (P), transmission data volume (D), and transfer latency (TL) in a real-world robot system. Uplink parameters are calculated with a five-frame input.

	P	*D*	*TL*
Uplink	/	31.10 MB	0.498s
Uplink-UTS	/	65.54 KB	0.001s
Downlink	7.78B	14.48 GB	65.490s
Downlink-DWC	1.65M	0.791 MB	0.013s

4.4. Real-world Validations

We utilized Gigabit Ethernet as the actual network environment with a theoretical peak of 1000Mbps, adhering to the 802.11ac (Wi-Fi 5) standard. We employed the Realsense D435i as the image capture device on the device, collecting images at a resolution of 1920×1080. The effectiveness of our CD-CCA was further validated through experiments on a real machine, as shown in Table 5, which includes the bidirectional transmission parameters size (P), transmission data volume (D), and transfer latency (TL).

5. Conclusion

We propose CD-CCA to empower device models in dynamic environments. Experimental results in the open-world scenario demonstrate performance improvements of 2.20% (CIDEr) and 3.93% (MC), 3.19% (DA) in the domain-shifted captioning and VQA tasks. Furthermore, real-world experiments have shown that the system delay of CD-CCA is able to support practical applications.

Acknowledgement. Shanghang Zhang is supported by the National Science and Technology Major Project of China (No. 2022ZD0117801).

References

[1] OpenAI. GPT-4 technical report, 2023. 1

[2] Jean-Baptiste Alayrac et al. Flamingo: a visual language model for few-shot learning. 2022. 1, 2, 3

[3] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023. 1, 2, 3

[4] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020. 2, 6, 7

[5] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, 2022. 2, 6, 7

[6] Riccardo Volpi, Diane Larlus, and Grégory Rogez. Continual adaptation of visual representations via domain randomization and meta-learning. *computer vision and pattern recognition*, 2020.

[7] Jiaming Liu, Senqiao Yang, Peidong Jia, Ming Lu, Yandong Guo, Wei Xue, and Shanghang Zhang. Vida: Homeostatic visual domain adapter for continual test time adaptation. *arXiv preprint arXiv:2306.04344*, 2023. 2

[8] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2, 3, 6

[9] Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023. 2

[10] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015. 2, 6

[11] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *European Conference on Computer Vision*, pages 146–162. Springer, 2022. 2, 6, 7

[12] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 2, 6, 7

[13] Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. Nocaps: Novel object captioning at scale. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8948–8957, 2019. 2, 6, 7

[14] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3

[15] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022. 3

[16] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp, 2019. 3

[17] Sandeep Chinchali, Apoorva Sharma, James Harrison, Amine Elhafsi, Daniel Kang, Evgenya Pergament, Eyal Cidon, Sachin Katti, and Marco Pavone. Network offloading policies for cloud robotics: a learning-based approach, 2019. 3

[18] Daniel Crankshaw, Xin Wang, Giulio Zhou, Michael J. Franklin, Joseph E. Gonzalez, and Ion Stoica. Clipper: A low-latency online prediction serving system, 2017.

[19] Sadjad Fouladi, John Emmons, Emre Orbay, Catherine Wu, Riad S. Wahby, and Keith Winstein. Salsify: Low-Latency network video through tighter integration between a video codec and a transport protocol. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*, pages 267–282, Renton, WA, April 2018. USENIX Association. ISBN 978-1-939133-01-4. URL https://www.usenix.org/conference/nsdi18/presentation/fouladi.

[20] Yiping Kang, Johann Hauswald, Cao Gao, Austin Rovinski, Trevor Mudge, Jason Mars, and Lingjia Tang. Neurosurgeon: Collaborative intelligence between the cloud and mobile edge. *ACM SIGARCH Computer Architecture News*, 45 (1):615–629, 2017.

[21] Yulu Gan, Mingjie Pan, Rongyu Zhang, Zijian Ling, Lingran Zhao, Jiaming Liu, and Shanghang Zhang. Cloud-device collaborative adaptation to continual changing environments in the real-world, 2022. 3

[22] Tao Huang, Yuan Zhang, Shan You, Fei Wang, Chen Qian, Jian Cao, and Chang Xu. Masked distillation with receptive tokens. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=mWRngkvIki3. 3

[23] Tao Huang, Yuan Zhang, Mingkai Zheng, Shan You, Fei Wang, Chen Qian, and Chang Xu. Knowledge diffusion for distillation. *NeurIPS*, 30, 2023. 3

[24] Linfeng Zhang and Kaisheng Ma. Improve object detection with feature-based knowledge distillation: Towards accurate and efficient detectors. In *ICLR*, 2020. 3

[25] Chuanguang Yang, Helong Zhou, Zhulin An, Xue Jiang, Yongjun Xu, and Qian Zhang. Cross-image relational knowledge distillation for semantic segmentation. In *CVPR*, pages 12319–12328, 2022.

[26] Zheqi Lv, Wenqiao Zhang, Shengyu Zhang, Kun Kuang, Feng Wang, Yongwei Wang, Zhengyu Chen, Tao Shen, Hongxia Yang, Beng Chin Ooi, et al. Duet: A tuning-free device-cloud collaborative parameters generation framework

for efficient device model generalization. In *Proceedings of the ACM Web Conference 2023*, pages 3077–3085, 2023. 3

[27] Zhendong Yang, Zhe Li, Xiaohu Jiang, Yuan Gong, Zehuan Yuan, Danpei Zhao, and Chun Yuan. Focal and global knowledge distillation for detectors. In *CVPR*, pages 4643–4652, 2022. 3

[28] Yuan Zhang, Weihua Chen, Yichen Lu, Tao Huang, Xiuyu Sun, and Jian Cao. Avatar knowledge distillation: Self-ensemble teacher paradigm with uncertainty. *arXiv preprint arXiv:2305.02722*, 2023. 3

[29] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization, 2021. 3

[30] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation, 2021.

[31] Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without source data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9641–9650, 2020.

[32] Yuang Liu, Wei Zhang, and Jun Wang. Source-free domain adaptation for semantic segmentation, 2021. 3

[33] Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, et al. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*, 2023. 6, 7

[34] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 6

[35] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022. 6

[36] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 6

[37] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 6

[38] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568, 2021. 6

[39] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24, 2011. 6

[40] Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*, 2023. 6

[41] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 6

[42] Weihan Cao, Yifan Zhang, Jianfei Gao, Anda Cheng, Ke Cheng, and Jian Cheng. Pkd: General distillation framework for object detectors via pearson correlation coefficient. *Advances in Neural Information Processing Systems*, 35: 15394–15406, 2022. 6, 7

[43] Changyong Shu, Yifan Liu, Jianfei Gao, Zheng Yan, and Chunhua Shen. Channel-wise knowledge distillation for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5311–5320, 2021. 6, 7

[44] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36, 2024. 8