BEYOND NEXT-TOKEN ALIGNMENT: DISTILLING MULTIMODAL LARGE LANGUAGE MODELS VIA TOKEN INTERACTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal Large Language Models (MLLMs) have demonstrated impressive cross-modal understanding capabilities, yet their substantial model size poses significant challenges for widespread deployment. Knowledge distillation (KD) presents a promising solution for compressing these large-scale MLLMs. However, existing KD methods primarily rely on static next-token alignment, neglecting to model the dynamic token interactions, which embed essential capabilities for multimodal understanding and generation. To this end, we introduce **Align-TI**, a novel KD framework designed from the perspective of **T**oken **I**nteractions. Our approach is motivated by the insight that MLLMs rely on two primary interaction types: vision-instruction token interactions to extract instruction-relevant visual information, and intra-response token interactions for dynamic reasoning and coherent generation. Accordingly, Align-TI introduces two components: Instruction-aware Vision Alignment (IVA) and Transition Probability Alignment (TPA). IVA enables the student model to imitate the teacher's ability to extract instruction-relevant visual information by aligning on salient visual regions. TPA captures the teacher's dynamic generative logic by aligning the sequential token-to-token transition probabilities. Extensive experiments on standard multimodal benchmarks demonstrate the superiority of Align-TI. Notably, our approach achieves 3.7% relative improvement over direct supervised fine-tuning across multiple benchmarks. Moreover, our distilled Align-TI-2B even outperforms LLaVA-1.5-7B (a much larger MLLM) by 7.0%, establishing a new state-of-the-art distillation framework for training parameter-efficient MLLMs.

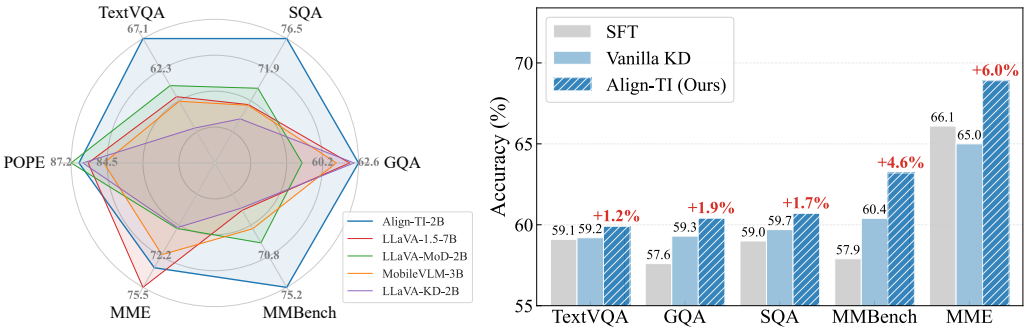


Figure 1: Experimental results overview. **Left:** Performance comparison between MLLMs distilled using our proposed Align-TI and other state-of-the-art MLLMs. **Right:** Performance gains achieved by Align-TI relative to the SFT and Vanilla KD baselines. (Details provided in Appendix E.1.)

1 INTRODUCTION

Multimodal Large Language Models (MLLMs) (Liu et al., 2023c; Hurst et al., 2024; Guo et al., 2025; Comanici et al., 2025) have emerged as a cornerstone in the pursuit of Artificial General Intelligence (AGI), showcasing remarkable capabilities in cross-modal understanding and generation. The success of contemporary MLLMs is predominantly built upon the foundation of autoregressive large language models (Floridi & Chiriatti, 2020; Touvron et al., 2023; Yang et al., 2025),

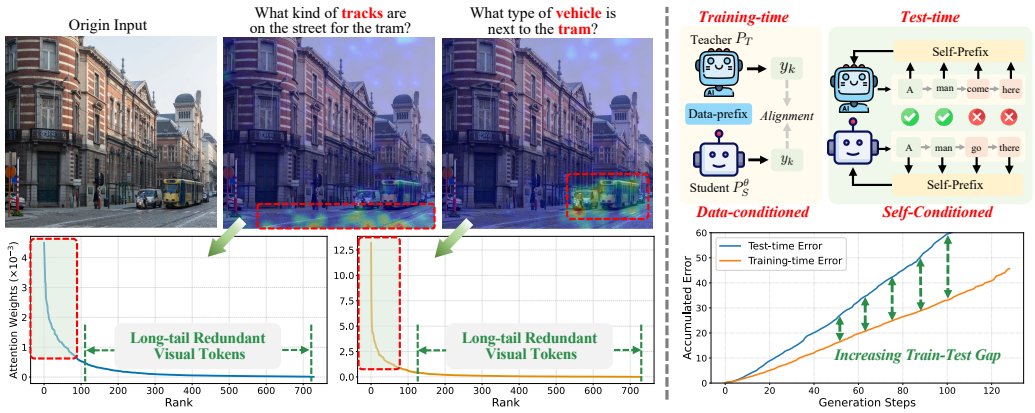


Figure 2: Motivation of MLLM distillation in view of token interactions. **Left: Vision-instruction token interaction analysis.** Visualizations of instruction-to-vision attention weights demonstrate that different instructions activate distinct visual focus areas, while exhibiting significant token redundancy. **Right: Intra-response token interaction analysis.** The discrepancy between data-conditioned prefix during training-time and self-conditioned prefix during test-time amplifies autoregressive accumulated error. (More details are provided in Appendix B.4.1.)

which adopt self-supervised pretraining with the next-token prediction paradigm. Previous studies (Radford et al., 2018; Mann et al., 2020; Achiam et al., 2023) have demonstrated that scaling up these models' parameters and training data continuously pushes the performance boundaries, while it also results in large-scale models with significant computational demands. Knowledge distillation (KD) (Hinton et al., 2015) provides a promising pathway to reduce computational overhead requirements by replicating large-scale teacher model capabilities in more parameter-efficient students through systematic knowledge transfer.

Prior work (Xu et al., 2024a) demonstrates that knowledge transfer in MLLM distillation via intermediate features or attention maps is often ineffective, primarily due to functional misalignment between student and teacher layers. In contrast, aligning output token distributions has proven to be a more effective way. Building upon the foundation of token-level alignment, subsequent research (Cai et al., 2024; Feng et al., 2025; Shu et al., 2024) introduces additional components, such as MoE architectures (Dai et al., 2024). **However, the token-level alignment in these methods remains limited to static next-token alignment, neglecting to model the dynamic token interactions that encode critical capabilities for MLLM understanding and generation.** Specifically, these interactions encode crucial capabilities in two stages: (1) Prefilling: Vision-instruction token interactions encode instruction-aware visual information extraction capability. (2) Decoding: Intra-response token interactions encode dynamic reasoning and generation capability. The absence of such interaction modeling restricts the student to acquiring shallow statistical patterns from the teacher's outputs, rather than its deeper mechanisms for understanding and generation, thus resulting in insufficient knowledge transfer.

To provide better knowledge transfer, we further analyze the underlying characteristics of these two types of interactions. **(1) Vision-instruction token interactions.** In Fig. 2 (Left), we visualize the instruction-to-vision attention weights, it reveals two key observations: (a) Identical images elicit distinct region activations under different instructions. (b) Instruction tokens attend primarily to a few salient visual tokens, while the majority follow a long-tailed distribution. This imbalance indicates that prior distillation methods compel the capacity-constrained student to misallocate precious resources toward mimicking the teacher's processing of low-utility tokens, thereby hindering its ability to master instruction-critical representations. **(2) Intra-response token interactions.** As depicted in Fig. 2 (Right), prior distillation methods primarily align the teacher and student models' data-conditioned next token prediction probabilities, where the prefix originates from the training corpus. This approach neglects the transition dynamics inherent in test-time generation, where predictions are conditioned on the self-generated outputs. As shown in Fig. 2 (Right), this issue leads to a widening accumulated error gap between training-time and test-time, a challenge referred to as exposure bias in imitation learning (Ross et al., 2011; Arora et al., 2022; Kim et al., 2024).

Based on the aforementioned discussion, we propose **Align-TI**, a framework that explicitly modeling KD for MLLMs from the perspective of token interactions. Align-TI consists of two core com-

ponents: **I**nstruction-aware **V**ision **A**lignment (**IVA**) and **T**ransition **P**robability **A**lignment (**TPA**), corresponding to the two interaction types. Specifically, IVA enables the student model to learn on the teacher model's instruction-aware visual focus, facilitating the transfer of the teacher's visual information extraction capability. Additionally, recognizing that this visual focus varies significantly across transformer layers, we design the **I**nstruction-**R**elevant **S**core (**IRS**) to quantify the relevance of a layer's attention map to the given instruction. This enables principled selection of the most instruction-relevant visual focus for IVA. Furthermore, TPA explicitly aligns the token-to-token transition probabilities, enabling the student to better learn the teacher's continuous token generation patterns. Moreover, both theoretical analysis and experimental evidence demonstrate that TPA helps mitigate the teacher-student autoregressive generation discrepancy at test time.

Extensive experiments validate the efficacy of Align-TI in distilling knowledge from large-scale MLLMs into compact models, as summarized in Fig. 1. Our distilled 1B model achieves relative improvements of 4.6% on MMBench (Liu et al., 2024c) and 6.0% on MME (Fu et al., 2024) compared to the Vanilla KD baseline. Furthermore, our Align-TI-2B outperforms both small LLaVA-MoD-2B (Shu et al., 2024) (a strong MoE-based distillation baseline) and the larger LLaVA-1.5-7B (Liu et al., 2023b), advancing the state-of-the-art in distillation strategies for parameter-efficient MLLMs.

2 Preliminaries

Multimodal Large Language Models. By integrating the capabilities of pretrained LLMs with vision encoders, MLLMs establish a connection between the visual and linguistic modalities. Given an input $\boldsymbol{x} = (X_v, X_q)$, where X_v represents the input image and X_q denotes the textual instruction, MLLMs aim to generate a response $\boldsymbol{y}$ conditioned on $\boldsymbol{x}$. A typical MLLM architecture comprises three core components: a vision encoder, a vision-language projector, and a large language model. The MLLM processing pipeline can be formally defined as:

$$\mathcal{O} = \mathrm{LLM}_\phi\left(\mathrm{Proj}_\omega\left(\mathrm{Vis}_\psi(X_v)\right), X_q\right), \tag{1}$$

where ψ, ω and ϕ denote the parameters of the vision encoder, projector and language model, respectively. The output $\mathcal{O} = \{\boldsymbol{v}, \boldsymbol{q}, \boldsymbol{y}\}$ consists of visual tokens $\boldsymbol{v}$, instruction tokens $\boldsymbol{q}$, and response tokens $\boldsymbol{y}$. Note that $\boldsymbol{v}$ and $\boldsymbol{q}$ are present in the output because the transformer decoder maintains an equal number of input and output tokens. Unlike the supervised response tokens $\boldsymbol{y}$, the tokens $\boldsymbol{v}$ and $\boldsymbol{q}$ serve as unsupervised conditional context, representing the model's comprehension of the input $\boldsymbol{x}$ that guides the generation process.

Training Scheme of MLLMs. A typical MLLM training scheme (Liu et al., 2023c) consists of two stages: pretraining and fine-tuning. The pretraining stage focuses on learning the vision-language projector Proj_ω to establish cross-modal alignment using caption data. Subsequently, the fine-tuning stage jointly optimizes both Proj_ω and LLM_ϕ using instruction-following data and limited caption data, thereby enhancing instruction-following capabilities.

Problem Definition. We explore knowledge distillation for MLLMs, aiming to distill small-scale MLLMs from powerful teacher MLLMs. Formally, given a teacher model distribution $P_T(\boldsymbol{y}|\boldsymbol{x})$ and a parameterized student distribution $P_S^\theta(\boldsymbol{y}|\boldsymbol{x})$, the knowledge distillation objective minimizes their distributional divergence on a dataset $\mathcal{D} = \{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\}_{i=1}^{|\mathcal{D}|}$.

Vanilla KD. Vanilla KD recognizes the autoregressive nature of language models and performs next-token alignment by combining ground-truth supervision with distribution matching. Given a query $\boldsymbol{x}$ and its corresponding ground-truth response sequence $\boldsymbol{y}_{1:L}^{\mathcal{D}}$, the objective is formalized as minimizing the forward KL divergence between P_T and P_S^θ at each decoding step:

$$\mathcal{L}_{\mathrm{kd}}(\theta) = \mathop{\mathbb{E}}_{(\boldsymbol{x}, \boldsymbol{y}_{1:L}^{\mathcal{D}}) \sim \mathcal{D}} \left[\sum_{k=1}^{L} D_{\mathrm{KL}}\left(P_T \parallel P_S^\theta\right)\left(y_k \mid \boldsymbol{x}, \boldsymbol{y}_{<k}^{\mathcal{D}}\right) \right], \tag{2}$$

where $\boldsymbol{y}_{<k}^{\mathcal{D}}$ denotes the ground-truth prefix tokens, and D_{KL} represents the KL divergence. This step-wise distillation transfers the teacher's generation preferences at each generation step.

3 Framework of Align-TI

Fig. 3 presents the overview of our proposed knowledge distillation framework, Align-TI, which is modeled in view of token interactions. It consists of two core components: (1) Instruction-aware

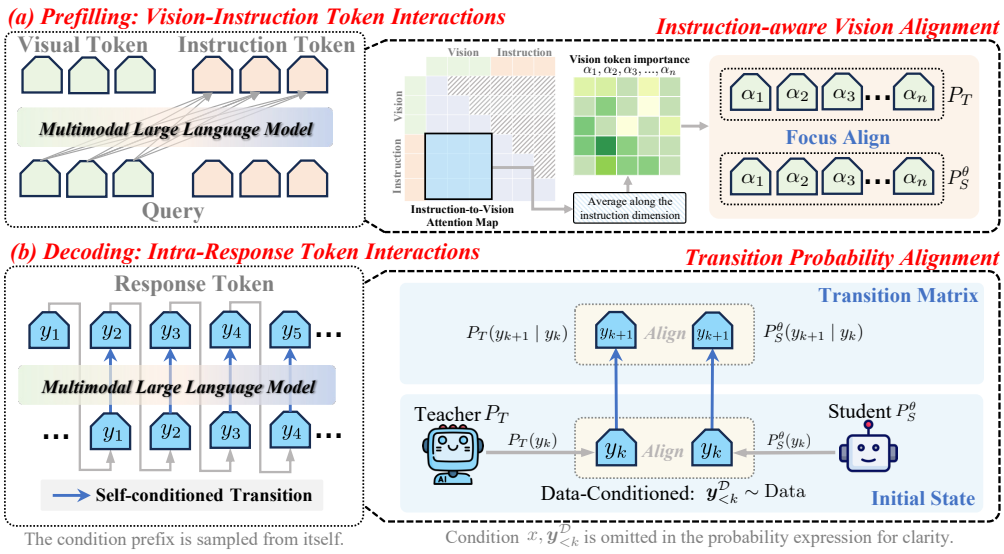


Figure 3: Overview of the proposed Align-TI. The framework explicitly models knowledge distillation in MLLMs from the perspective of token interactions.

Vision Alignment (IVA), which aligns visual tokens by incorporating instruction-aware importance weights to focus on salient visual regions. (2) Transition Probability Alignment (TPA), which aligns not only the initial token distributions conditioned on ground-truth data but also the transition probabilities to better imitate the teacher's autoregressive generation process.

3.1 Instruction-aware Vision Alignment

Unlike the conventional method (Cai et al., 2024) that applies uniform alignment for all visual tokens, our proposed IVA prioritizes alignment of salient visual regions. This targeted approach enables a more fine-grained alignment, mitigating the influence of redundant visual information.

Objective Derivation. The key insight of IVA is to exploit the focus relationships embedded in the cross-attention map between instruction and visual tokens. Given the full attention map $\boldsymbol{A}$, we first extract the instruction-to-vision sub-matrix $\boldsymbol{A}_{i\to v} \in \mathbb{R}^{N_i \times N_v}$, where N_i and N_v denote the numbers of instruction and visual tokens, respectively. We aggregate attention weights across the instruction dimension to obtain the instruction-aware importance for each visual token. Subsequently, the IVA objective is formulated by weighting the per-token alignment loss with these importance weights:

$$\mathcal{L}_{\text{iva}}(\theta) = \sum_{k=1}^{N_v} \frac{1}{N_i} \left(\sum_{u=1}^{N_i} \boldsymbol{A}_{i\to v}(u,k) \right) \cdot D_{\text{KL}}(P_T \| P_S^\theta)(v_k \mid \boldsymbol{v}_{<k}), \tag{3}$$

where $\alpha_k = \frac{1}{N_i} \sum_{u=1}^{N_i} \boldsymbol{A}_{i\to v}(u,k)$ denotes the instruction-aware importance weight for the k-th visual token, and $\boldsymbol{v}$ represents the set of visual tokens. This formulation elegantly directs the model to focus its learning on the most salient visual information as indicated by the instruction.

Remark. The ultimate goal of KD is to train a student model to generate responses aligned with the teacher model. Although IVA aligns visual tokens rather than directly establishing relationships with response tokens, it enables the student model to imitate the teacher's processing of visual tokens. As a result, the hidden representations of the student model are implicitly optimized to be more effective for generating the target response.

Definition 3.1 (Instruction-Relevance Score). *Let $\boldsymbol{\alpha}^{(l)}(\boldsymbol{x})$ denote the vectorized instruction-aware visual token importance weights, extracted from the l-th transformer layer for a given input query $\boldsymbol{x}$. The Instruction-Relevance Score (IRS) for layer l is defined as:*

$$\text{IRS}(l) = 1 - \mathbb{E}_{\substack{\boldsymbol{x}_1, \boldsymbol{x}_2 \\ i.i.d. \sim \mathcal{D}_x}} \left[\cos\left(\boldsymbol{\alpha}^{(l)}(\boldsymbol{x}_1), \boldsymbol{\alpha}^{(l)}(\boldsymbol{x}_2) \right) \right], \tag{4}$$

where $\cos(\cdot,\cdot)$ denotes the cosine similarity function, and $\mathcal{D}_x$ is the distribution of input queries.

Remark. The IRS is a metric designed to measure the relevance between attention distribution and the instruction's semantics. The core intuition of IRS is that if a layer's attention pattern exhibits minimal variation across different inputs, its functionality is likely generic and instruction-agnostic. The value range of IRS is $[0, 1]$, and a lower IRS indicates weak relevance to the instruction.

Principled Layer Selection. Each layer of MLLMs generates an attention map, from which we can extract the instruction-aware importance weights. However, the evolving focus patterns across layers present a challenge for selecting the optimal layer. Visual token pruning methods (Ye et al., 2025) similarly leverage these maps, but they often rely on empirical choices. To make a principled selection, we employ the IRS to choose the most instruction-relevant layer. Our rationale is that a higher IRS signifies the layer is more sharply focused on instruction-relevant visual regions. Fig. 4 visualizes IRS across layers in a Qwen2-7B-based MLLM. It shows that shallow layers exhibit low IRS, indicating limited semantic grounding in early attention distributions. As depth increases, IRS increases, indicating that attention progressively sharpens and concentrates on instruction-relevant visual regions. However, this trend reverses in the few final layers, where the metric begin to decrease. We find that this is because the model starts integrating broader contextual information, causing the attention to lose its sharp focus on specific objects (see Appendix C for visualization). This pattern of IRS is generalizable, and we also observe it in a Vicuna-7B-based MLLM (Appendix C). Based on this analysis, we select the layer with the highest IRS for IVA, with experiments in Sec. 4.2 validating its efficacy.

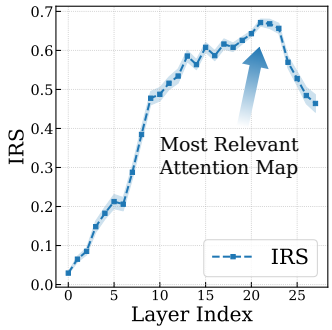


Figure 4: Analysis of IRS.

3.2 Transition Probability Alignment

In contrast to Vanilla KD, which aligns next-token predictions conditioned on ground-truth prefixes sampled from data $\mathcal{D}$, TPA aligns the token-to-token transition probability matrix. By emphasizing transition dynamics rather than only next-step predictions, TPA enables more effective transfer of sequential generation patterns and mitigates the growing discrepancy between teacher and student models during autoregressive decoding.

Objective Derivation. Given a query $\boldsymbol{x}$ and a ground-truth prefix $\boldsymbol{y}^{\mathcal{D}}_{<k}$, our objective is to align not only the immediate next-token distribution but also the subsequent transition probabilities. We formalize this objective as the minimization of the following loss:

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y}^{\mathcal{D}}_{1:L})\sim\mathcal{D}}\left[\sum_{k=1}^{L}\underbrace{D_{\mathrm{KL}}(P_T \parallel P_S^\theta)(y_k \mid \boldsymbol{x}, \boldsymbol{y}^{\mathcal{D}}_{<k})}_{\text{Initial State Align } \mathcal{L}_{\mathrm{kd}}(\theta)} + \underbrace{\mathbb{E}_{y_k\sim P_S^\theta} D_{\mathrm{KL}}(P_T \parallel P_S^\theta)(y_{k+1} \mid y_k, \boldsymbol{x}, \boldsymbol{y}^{\mathcal{D}}_{<k})}_{\text{Transition Probability Align } \mathcal{L}_{\mathrm{tpa}}(\theta)}\right]. \quad (5)$$

The first term, $\mathcal{L}_{\mathrm{kd}}(\theta)$, is the objective of Vanilla KD, which aligns the initial state distribution at step k. The second term, $\mathcal{L}_{\mathrm{tpa}}(\theta)$, is our proposed objective, which aligns the one-step transition probability matrices:

$$P_T(y_{k+1} \mid y_k), P_S^\theta(y_{k+1} \mid y_k) \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}, \quad \forall y_k, y_{k+1} \in \mathcal{V}, \quad (6)$$

where $\mathcal{V}$ denotes the vocabulary set. These matrices capture token-to-token transition dependencies conditioned on x and $\boldsymbol{y}^{\mathcal{D}}_{<k}$. Moreover, direct alignment of the full matrix is computationally infeasible due to the large vocabulary size $|\mathcal{V}|$. Fortunately, since initial probability distributions are highly long-tailed with most vocabulary entries having near-zero probability, aligning all matrix rows is unnecessary. We instead align transition probabilities conditioned on the student's distribution ($y_k \sim P_S^\theta$). Alternatively, conditioning on the teacher distribution ($y_k \sim P_T$) is possible, but it is less effective than conditioning on the student distribution. Specifically, the student distribution allows on-policy exploration of the student's predictive space, facilitating correction of potential errors while maintaining computational efficiency by avoiding additional teacher forward passes.

Remark. Consider the sequence decoding space defined over vocabulary $\mathcal{V}$. Vanilla KD aligns teacher-student distributions along $O(|\mathcal{V}|)$ generation paths by matching next-token probabilities. TPA expands this coverage to $O(|\mathcal{V}|^2)$ paths by additionally aligning the transition probability matrix, providing enhanced alignment scope. Detailed analysis is provided in Appendix D. This expanded scope enables supervision across more generation modes, achieving more exhaustive knowledge transfer. Furthermore, this expanded coverage increases the likelihood that student-generated

Table 1: Comparison with state-of-the-art MLLMs. Our Align-TI establishes new state-of-the-art results among compact ∼1B and ∼2B parameter models, and demonstrates competitive performance with some larger models. † denotes models obtained via MLLM distillation.

Method	LLM	#Params	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
LLaVA-1.5	Vicuna-7B	∼7B	62.0	66.8	58.2	85.9	75.5	64.3	68.8
Deepseek-VL	DLLM-7B		61.3	74.0	64.7	-	73.4	74.1	-
LLaVA-Next	Vicuna-7B		64.2	70.1	64.9	86.5	76.0	67.4	71.5
MobileVLM V2	Vicuna-7B		62.6	74.8	62.3	85.3	78.0	69.2	72.0
LLaVA-OV	Qwen2-7B		62.2	-	84.5	-	78.0	80.8	-
Qwen2.5-VL	Qwen2.5-7B		60.9	88.8	77.8	86.6	85.1	83.4	80.4
VILA	MLLaMA-2.7B	∼3B	61.5	69.0	60.4	85.9	72.1	63.4	68.7
MiniCPM-V-2	MiniCPM-2.4B		52.1	76.3	73.2	86.3	70.5	68.5	71.2
LLaVADI†	MLLaMA-2.7B		61.4	64.1	50.7	86.7	68.8	62.5	65.7
MobileVLM V2	MLLaMA-2.7B		61.1	66.7	57.5	84.7	72.0	63.2	67.5
TinyLLaVA	Phi2-2.7B		62.0	69.1	59.7	86.4	73.2	66.9	69.6
MoVE-KD†	MLLaMA-1.4B	∼2B	57.7	57.3	44.3	86.1	59.4	48.8	58.9
Align-KD†	MLLaMA-1.4B		60.1	67.7	53.1	87.0	65.2	57.5	65.1
Mini-Gemini	Gemma-2B		60.7	63.1	56.2	85.6	67.0	59.8	65.4
MoE-LLaVA	Qwen1.5-1.8B		61.5	63.1	48.0	87.0	64.6	59.7	64.0
LLaVA-KD†	Qwen1.5-1.8B		62.3	64.7	53.4	86.3	69.1	64.0	66.6
LLaVA-MoD†	Qwen2-1.5B		58.8	69.2	59.9	**87.2**	69.2	68.9	68.9
Align-TI†	Qwen2-1.5B		**62.9**	71.4	65.1	86.1	**75.6**	71.8	72.2
Align-TI†	Qwen3-1.7B		62.6	**76.5**	**67.1**	86.6	73.4	**75.2**	**73.6**
SPHINX-Tiny	TLLaMA-1.1B	∼1B	58.0	21.5	57.8	82.2	63.1	56.6	56.5
LLaVA-KD†	Qwen1.5-0.5B		59.6	60.6	49.9	85.9	64.5	60.1	63.4
LLaVA-MoD†	Qwen2-0.5B		56.6	61.1	57.1	-	67.0	58.7	-
Align-TI†	Qwen2-0.5B		60.4	60.7	59.9	86.8	68.9	63.2	66.7
Align-TI†	Qwen3-0.6B		**61.2**	**68.4**	**64.1**	**86.9**	**70.0**	**67.6**	**69.7**

trajectories during inference fall within the aligned distribution space, thereby mitigating the exposure bias caused by the train-test gap (described in Fig. 2 (Right)).

Objective Estimation. We employ a Monte Carlo approach to estimate the expectation of $\mathcal{L}_{\text{tpa}}(\theta)$. First, a forward pass through the student model yields the initial state probability distribution $P_S^\theta(y_k)$. We then sample a small set of d candidate tokens $\{y_k^{(u)}\}_{u=1}^d$ from this distribution. Alignment is performed only on transition matrix rows associated with $\{y_k^{(u)}\}_{u=1}^d$, formalized as:

$$\mathcal{L}_{\text{tpa}}(\theta) \simeq \mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y}^{\mathcal{D}}_{1:L})\sim\mathcal{D}} \sum_{k=1}^{L} \frac{1}{d} \sum_{u=1}^{d} D_{\text{KL}}\left(P_T \parallel P_S^\theta\right)\left(y_{k+1} \mid y_k^{(u)}, \boldsymbol{x}, \boldsymbol{y}^{\mathcal{D}}_{<k}\right). \tag{7}$$

Parallelized Calculation. A naive implementation of Eq. 7 would require d separate forward passes on both the student model and the teacher model, incurring excessive computational cost. We propose an efficient parallel calculation method that computes these values in a single pass. Our approach begins by reorganizing the input response sequence as follows: $\hat{\boldsymbol{y}} = \{y^{\mathcal{D}}_{k-1}, y_k^{(1)}, y_k^{(2)}, \ldots, y_k^{(d)}\}_{k=1}^{L}$. During the forward pass of $\hat{\boldsymbol{y}}$, we apply a specially designed ribbon attention mask, as illustrated in Fig. 5. Specifically, the mask ensures $y_k^{(u)}$ can only attend to the common context of y_{k-1}, but not to any other candidate $y_k^{(v)}$ $(v \neq u)$. This strategy effectively creates d parallel causal pathways, enabling simultaneous estimation of the required transition probabilities and significantly improving computational efficiency.

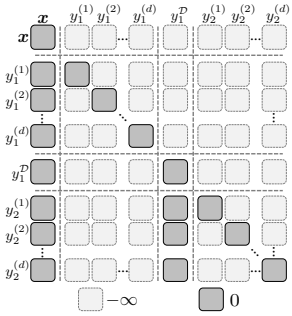


Figure 5: Visualization of ribbon attention mask.

3.3 Overall Objective for Align-TI

The overall objective for Align-TI integrates the standard SFT loss with our proposed distillation losses. The final objective $\mathcal{L}(\theta)$ is a summation of all components:

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{sft}}(\theta) + \mathcal{L}_{\text{iva}}(\theta) + \mathcal{L}_{\text{kd}}(\theta) + \mathcal{L}_{\text{tpa}}(\theta) \tag{8}$$

4 EXPERIMENTAL RESULTS

Implementation Details. We utilize the Qwen2 (Bai et al., 2025) and Qwen3 (Yang et al., 2025) series as the LLMs for our student and teacher models. The teacher model comprises approximately 7-8B parameters, while the student model contains 1-2B parameters. The performance of teacher models is presented in Tab. 2. Models within the same series are paired for distillation (e.g., Qwen2-7B serves as the teacher for Qwen2-0.5B/1.5B students). We follow the MobileVLM V2 (Chu et al., 2024b) to organize our training data, with 1.2M captioning samples for pretraining and 2.4M mixed captioning and VQA samples for fine-tuning. Due to the limited learnable parameters, the KD is only adopted in the fine-tuning stage. The number of sampled tokens d is set to 4. More implementation details and hyperparameters are illustrated in Appendix B.1. We mainly compare on six benchmarks to evaluate the multimodal understanding and VQA capabilities, more details about our evaluation benchmark are provided in Appendix B.2.

Table 2: Performance of our Teacher Models.

LLM	VisEnc	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
Qwen2-7B	SigLIP-B	64.6	80.7	64.1	86.1	78.6	76.0	75.1
Qwen3-8B	SigLIP-B	64.0	83.4	64.1	86.4	80.4	77.3	76.0

4.1 MAIN RESULTS

Comparison with State-of-the-art MLLMs. As presented in Tab. 1, we benchmark our models against state-of-the-art MLLMs, including both models trained from scratch and those derived via distillation. Our distilled Align-TI models achieve the best performance within the $\sim$1B and $\sim$2B parameter scales. Notably, Align-TI-2B surpasses substantially larger counterparts, outperforming LLaVA-1.5-7B and MobileVLM-V2-7B (Chu et al., 2024b) by relative 7.0% and 2.2%, respectively. Furthermore, Align-TI-2B achieves a significant 4.8% performance gain over LLaVA-MoD-2B (Shu et al., 2024), a superior MoE-based MLLM distillation baseline. These results demonstrate the efficacy of our distillation approach for transferring knowledge from large-scale MLLMs and developing high-performing small-scale MLLMs.

Comparison with Distillation Strategy Designed for LLMs. We further compare our method with distillation strategies specifically designed for LLMs, with results summarized in Tab. 3. Classical distillation approaches for LLMs primarily employ diverse divergences, such as Forward KL (FKL), Jensen-Shannon Divergence (JSD) and Reverse KL in MiniLLM (Gu et al., 2023). Our experiments show that FKL yields superior performance over other KL variants, which aligns with prior findings that the optimal divergence is task-dependent (Agarwal et al., 2024; Xu et al., 2024a). Additionally, GKD (Agarwal et al., 2024) exhibits an average performance degradation of 2.3 relative to FKL. This may be attributed to GKD aligning on student-generated on-policy responses, which can lead to incorrect answers, particularly in more challenging multi-modal scenarios. Nevertheless, all these LLM-centric strategies are substantially outperformed by Align-TI, as the gap between LLM and MLLM is significant.

Table 3: Comparison with knowledge distillation strategies designed for LLM.

Model	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
FKL	59.3	59.7	59.2	86.2	65.0	60.4	65.0
MiniLLM	59.4	59.9	58.7	85.5	65.6	57.9	64.5
JSD	57.4	58.9	57.9	85.9	66.2	56.1	63.7
GKD	55.4	58.5	57.3	85.6	61.5	57.9	62.7
Align-TI	**60.4**	**60.7**	**59.9**	**86.8**	**68.9**	**63.2**	**66.7**

Table 4: Ablation Study on IVA and TPA

IVA	TPA	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
		57.6	59.0	59.1	86.2	66.1	57.9	64.3
✓		59.6	58.0	**61.3**	86.5	66.8	58.3	65.1
	✓	60.3	**61.0**	59.6	86.5	68.1	63.0	66.4
✓	✓	**60.4**	60.7	59.9	**86.8**	**68.9**	**63.2**	**66.7**

Table 5: Inference Efficiency compared with LLaVA-1.5-7B (Liu et al., 2023b).

Metrics	LLaVA-1.5	Align-TI
Params (B)	7	2
Peak Memory (GiB)	14.0	4.8
Time to First Token (ms)	90	57
Throughput (token/s)	33.8	64.8
AVG Performance (%)	68.8	73.6

Table 6: Training Efficiency.

Metrics	Vanilla KD	GKD	Align-TI		
			IVA	TPA	IVA+TPA
Training Time (H)	355	962	359	504	509
Memory (GiB)	70.6	76.8	70.7	75.3	75.6

Efficiency Analysis. We assess the computational efficiency of our method from two perspectives: inference speed and training overhead. As shown in Tab. 5, Align-TI-2B not only surpasses the much larger LLaVA-1.5-7B in performance but also exhibits significant computational advantages. Specifically, it achieves $1.7\times$ faster first-token generation, $1.9\times$ higher decoding throughput, and consumes only 4.8 GiB of peak memory. These characteristics make it highly suitable for deployment on resource-constrained edge devices. Furthermore, Tab. 14 analyzes the training efficiency of our approach. It demonstrates that IVA incurs negligible additional computational overhead, with the majority of the cost arising from TPA. Compared to Vanilla KD, our Align-TI increases training time by $1.4\times$, whereas GKD results in a $2.7\times$ training time overhead.

4.2 Ablation Study

Component Analysis. To evaluate the contributions of IVA and TPA, Tab. 4 presents an ablation study on their individual and combined effects. When neither IVA nor TPA is employed, the baseline model achieves an average performance of 64.3. In contrast, the combined integration of TPA and IVA yields an average performance of 66.7, which represents an improvement of 2.4 over the baseline. When applied separately, TPA and IVA improve the baseline by 2.1 and 0.8, respectively, confirming the efficacy of each module. Notably, the performance gain from TPA is larger than that from IVA. This observation aligns with their distinct mechanisms: TPA imposes an explicit constraint by directly aligning output distributions, whereas IVA operates indirectly by matching latent feature representations for response generation. Moreover, as illustrated in Tab. 14, the training time overhead of TPA is significantly higher than that of IVA, while IVA can be integrated with TPA at nearly no additional cost.

Design of Important Weights. Tab. 7 presents our investigation into the optimal layer depth for extracting importance weights. By evaluating five equidistant layers, we find that optimal performance is achieved with the 21st layer, located approximately three-quarters of the model's depth. Notably, this layer coincides with the model region exhibiting the highest IRS. In contrast, weights from layers with low instruction relevance degrade model performance. This finding underscores that IVA's effectiveness originates from its ability to focus student model's limited capability on instruction-salient regions while filtering out the impact of redundant visual tokens. Furthermore, we compare the effects of using importance weights derived from the student and teacher models, as detailed in Tab. 8. The results reveal that aligning with the teacher's attention focus yields an average performance improvement of 0.4. This improvement can be attributed to the fact that the student model, with its limited capacity and lack of SFT, does not yet possess a robust ability to accurately focus on instruction-relevant regions.

Table 7: Comparison of importance weights extracted from different layer depths for IVA.

Layer	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
0 (0/4)	58.1	56.9	58.9	86.3	66.3	55.8	63.7
7 (1/4)	55.4	56.1	40.8	85.1	62.6	53.0	58.8
14 (2/4)	58.3	57.1	59.6	85.8	66.0	56.7	63.9
21 (3/4)	**59.6**	58.0	**61.3**	86.5	66.8	**58.3**	**65.1**
27 (4/4)	58.8	**58.2**	59.3	**86.8**	**67.1**	57.5	64.6

Table 8: Comparison of importance weights extracted from teacher and student models for IVA.

Method	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
Student	59.3	57.5	61.1	86.3	65.4	**58.3**	64.7
Teacher	**59.6**	**58.0**	**61.3**	**86.5**	**66.8**	**58.3**	**65.1**

Comparison with Uniform Alignment. To demonstrate the efficacy of our proposed IVA, we benchmark it against the uniform alignment strategy from (Cai et al., 2024), which assigns equal weights to all visual tokens. As shown in Tab. 9, IVA outperforms uniform alignment on all six benchmarks, achieving an average improvement of 0.6. This highlights that IVA's instruction-aware weighting mechanism facilitates a more effective and targeted alignment.

Table 9: Compare with visual token alignment with uniform weights (Cai et al., 2024).

Method	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
Uniform	59.4	57.3	61.0	86.3	65.1	58.0	64.5
Ours	**59.6**	**58.0**	**61.3**	**86.5**	**66.8**	**58.3**	**65.1**

Impact of Different TPA Designs. We compare two sampling strategies for TPA: (i) greedy sampling, which selects the top-d most probable tokens, and (ii) nucleus sampling, which stochastically draws d tokens from the student's predictive distribution. Fig. 6 illustrates the model performance as a function of sampled token number d. For both strategies, performance improves as d increases, eventually plateauing around $d = 4$. This trend aligns with the expectation that sampling the high-

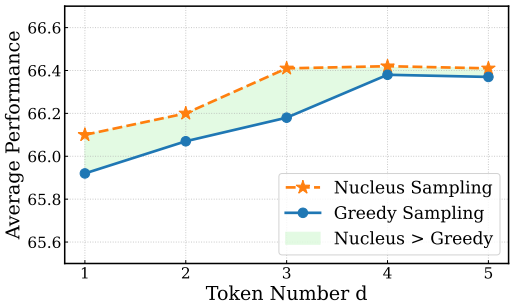


Figure 6: Ablation study on TPA design choices: comparing different sampling strategies and sampled token number d.

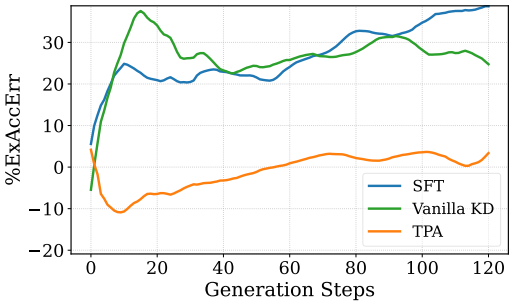


Figure 7: Evolution of %ExAccErr across generation steps, illustrating the effect of TPA on mitigating exposure bias.

probability part of the output distribution is adequate since language model output distributions are typically long-tailed. More importantly, nucleus sampling consistently outperforms greedy sampling in the low-d regime. We attribute this to the diversity inherent in nucleus sampling, which encourages the student to learn a broader range of state transitions. In contrast, greedy sampling focuses on the model's most confident, high-probability tokens, which often overlap with the ground truth, thereby providing redundant supervision.

4.3 Analysis on IVA and TPA

Analysis of IVA on Enhancing Visual Focus. We qualitatively examine how IVA strengthens a student model's ability to attend to instruction-relevant visual regions. As illustrated in Fig. 8, we visualize the instruction-to-vision attention maps for the student model (with and without IVA) alongside the teacher model. These maps are sourced from the layer exhibiting the highest IRS. We observe that equipping the student with IVA makes its attention patterns more closely align with the teacher's. Specifically, we identify two primary improvements: **(1) Focus correction**: Without IVA, the student may incorrectly attend to unrelated objects. For instance, when asked about a "green logo," it focuses on an entirely different logo (top row). IVA helps redirect its attention to the correct target. **(2) Focus sharpening**: Even when the student localizes the correct general area without IVA, its attention can be dispersed across irrelevant regions. IVA refines this into a concentrated map that closely follows the teacher's precise focus (bottom row). These findings demonstrate that IVA effectively distills the teacher's ability to extract visual information.

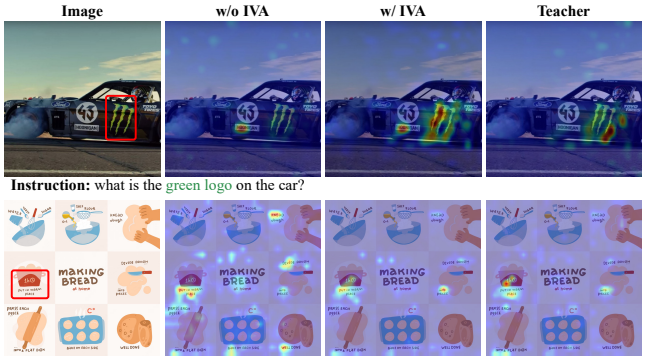


Figure 8: Qualitative analysis of IVA. IVA enhances student attention by correcting misdirected focus (top row) and sharpening diffuse attention maps into precise ones (bottom row).

Analysis of TPA on Mitigating Exposure Bias. Fig.7 depicts the analysis of %ExAccErr, a metric for assessing exposure bias with definition and calculation details outlined in Appendix B.4.2. When exposure bias is eliminated, the teacher and student models generate the same prediction distribution across different prefixes, resulting in a %ExAccErr being zero. As shown in Fig. 7, models trained with SFT and Vanilla KD both exhibit significant exposure bias, with their %ExAccErr values quickly rising and then stabilizing at around 30%. In contrast, the model distilled through our TPA approach exhibits %ExAccErr within the range of $(-10\%, 10\%)$. Moreover, the %ExAccErr even becomes negative in the early stages, indicating that the gap between student and teacher is smaller in the condition of a prefix generated by the student model. This finding strongly suggests that TPA successfully forces the student to learn the teacher's underlying transition dynamics, thereby aligning their output distributions across different contexts and effectively reducing exposure bias.

4.4 Scaling Analysis

Data Scaling. Fig. 9 presents the performance of SFT, Vanilla KD and Align-TI as a function of training data size, with Qwen2-0.5B as the LLM of student. All three approaches demonstrate consistent improvements as the amount of training data increases. Notably, Align-TI consistently outperforms Vanilla KD across all data scales. Furthermore, the results indicate that SFT and Vanilla KD perform similarly when trained on either a limited dataset (25%) or the full dataset (100%), whereas Align-TI delivers substantial gains in both settings. This highlights two key strengths of Align-TI: (1) it facilitates highly effective knowledge transfer even in data-scarce scenarios, and (2) it continues to distill supplementary knowledge when data is abundant, enabling further performance gains.

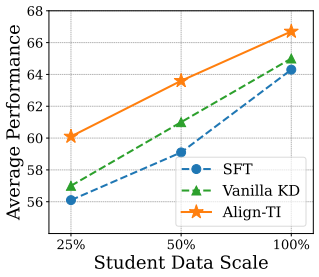


Figure 9: The scaling law for student training data.

Teacher Scaling. Tab. 10 presents the impact of teacher model size on Align-TI, using Qwen3-1.7B/4B/8B and Qwen3-0.6B as the LLM of teacher and student. Increasing the teacher size from 2B to 4B results in a notable average performance improvement, as indicated by 1.4 on SQA and 1.3 on GQA. However, further increasing the scale to 8B results in only marginal gains, with the average performance improving by a mere 0.1. This suggests that the benefits of a larger teacher exhibit diminishing returns, likely constrained by the representation capacity of the student model, a phenomenon also observed in (Mirzadeh et al., 2020).

Table 10: Scaling analysis of teacher model size.

Teacher Size	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
~2B	60.2	67.4	63.5	86.4	**70.5**	67.5	69.3
~4B	**61.5**	**68.8**	63.7	86.2	69.7	67.5	69.6
~8B	61.2	68.4	**64.1**	**86.9**	70.0	**67.6**	**69.7**

Model Architecture. To validate the effectiveness of Align-TI across diverse architectures, we conduct additional experiments using MobileLLaMA-1.4B (Chu et al., 2024a) as the student model, and MLLM with Vicuna-7B (Chiang et al., 2023) as the teacher. As shown in Tab. 11, Vanilla KD yields merely a 0.2 average improvement over SFT, while exhibiting a 1.5 performance drop on MME. In contrast, our method achieves an average gain of 2.1 relative to SFT, and achieves the best performance on all benchmarks. These results demonstrate the robustness of Align-TI across various model architectures.

Table 11: Distillation performance comparison on MobileLLaMA-1.4B as the student LLM.

Method	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
Teacher	62.0	70.4	61.2	85.9	75.5	64.3	67.3
SFT	58.4	64.8	60.6	85.6	66.6	56.2	65.4
Vanilla KD	58.9	65.6	61.4	86.0	65.1	56.6	65.6
Align-TI	**60.4**	**67.7**	**61.9**	**86.9**	**67.5**	**60.7**	**67.5**

5 Conclusion

This paper introduces Align-TI, a novel token-level knowledge distillation framework for transferring knowledge from large-scale to parameter-efficient MLLMs. Align-TI models knowledge distillation from the perspective of token interactions, including both vision-instruction token interactions and intra-response token interactions. We present an in-depth analysis of these two types of interactions, and propose two corresponding components: Instruction-aware Vision Alignment (IVA) and Transition Probability Alignment (TPA). IVA aligns visual tokens on the instruction-aware salient regions to learn the teacher's visual information extraction capability, and TPA distills the token-to-token transition probability to transfer the dynamics of autoregressive generation. Comprehensive experiments demonstrate the effectiveness of Align-TI in distilling MLLMs.

Limitations and Future Work. Due to limited computational resources, we validate the effectiveness of Align-TI only on image–text benchmarks. However, we believe Align-TI can be effectively extended to other modalities (e.g., video), as such tasks also produce token-level outputs, which can be modeled by Align-TI's objective. Exploring its potential in continuous spaces represents a promising direction, enabling application to diverse models such as unified frameworks and latent reasoning architectures. Moreover, exploring the vision-language alignment via distillation of vision-language projector could be a promising avenue.

6 Reproducibility Statement

Reproducibility is a priority in our research. In this statement, we outline the measures taken to ensure our work can be reproduced.

Source Code. The source code and pre-trained model weights will be made publicly available.

Experimental Setup and Details. In the main paper, the basic experimental configurations are presented in Sec. 4 (Implementation Details). For comprehensive documentation, Appendix B provides further implementation specifics, covering: the training details for teacher models (Appendix B.1.1), training details for student models(Appendix B.1.2), benchmark details (Appendix B.2), descriptions of comparison methods (Appendix B.3), calculation methodologies for Exposure Bias (Appendix B.4) and calculation details for IRS (Appendix B.5).

Datasets. All datasets used for training and evaluation are open-sourced and publicly accessible, and their details are provided in Appendix B.1 and B.2.

By highlighting these references, we intend to improve the reproducibility of our work, helping other researchers verify and build on our findings. We're open to any questions or requests for more information about our methods, as we aspire to ensure our research is transparent and reliable.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

Kushal Arora, Layla El Asri, Hareesh Bahuleyan, and Jackie Chi Kit Cheung. Why exposure bias matters: An imitation learning perspective of error accumulation in language generation. *arXiv preprint arXiv:2204.01171*, 2022.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Yuxuan Cai, Jiangning Zhang, Haoyang He, Xinwei He, Ao Tong, Zhenye Gan, Chengjie Wang, and Xiang Bai. Llava-kd: A framework of distilling multimodal large language models. *arXiv preprint arXiv:2410.16236*, 2024.

Jiajun Cao, Yuan Zhang, Tao Huang, Ming Lu, Qizhe Zhang, Ruichuan An, Ningning Ma, and Shanghang Zhang. Move-kd: Knowledge distillation for vlms with mixture of visual encoders. In *Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 19846–19856, 2025.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. In *European Conference on Computer Vision*, pp. 370–387. Springer, 2024a.

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.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to

commercial multimodal models with open-source suites. *Science China Information Sciences*, 67 (12):220101, 2024b.

Wei-Lin Chiang, Zhuohan Li, Ziqing Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3):6, 2023.

Xiangxiang Chu, Limeng Qiao, Xinyu Zhang, Shuang Xu, Fei Wei, Yang Yang, Xiaofei Sun, Yiming Hu, Xinyang Lin, Bo Zhang, et al. Mobilevlm v2: Faster and stronger baseline for vision language model. *arXiv preprint arXiv:2402.03766*, 2024a.

Xiangxiang Chu, Limeng Qiao, Xinyu Zhang, Shuang Xu, Fei Wei, Yang Yang, Xiaofei Sun, Yiming Hu, Xinyang Lin, Bo Zhang, et al. Mobilevlm v2: Faster and stronger baseline for vision language model. *arXiv preprint arXiv:2402.03766*, 2024b.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Yu Wu, et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. *arXiv preprint arXiv:2401.06066*, 2024.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 326–335, 2017.

Qianhan Feng, Wenshuo Li, Tong Lin, and Xinghao Chen. Align-kd: Distilling cross-modal alignment knowledge for mobile vision-language large model enhancement. In *Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 4178–4188, 2025.

Luciano Floridi and Massimo Chiriatti. Gpt-3: Its nature, scope, limits, and consequences. *Minds and machines*, 30(4):681–694, 2020.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, and Rongrong Ji. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2024.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. *arXiv preprint arXiv:2306.08543*, 2023.

Etash Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, et al. Openthoughts: Data recipes for reasoning models. *arXiv preprint arXiv:2506.04178*, 2025.

Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, et al. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*, 2025.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Runhui Huang, Xinpeng Ding, Chunwei Wang, Jianhua Han, Yulong Liu, Hengshuang Zhao, Hang Xu, Lu Hou, Wei Zhang, and Xiaodan Liang. Hires-llava: Restoring fragmentation input in high-resolution large vision-language models. In *Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 29814–29824, 2025.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 6700–6709, 2019.

Hugging Face. Open r1: A fully open reproduction of deepseek-r1, January 2025. URL https://github.com/huggingface/open-r1.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Jie Jiang, Qi Yang, Bolin Ni, Shiming Xiang, Han Hu, and Houwen Peng. R-4b: Incentivizing general-purpose auto-thinking capability in mllms via bi-mode annealing and reinforce learning, 2025. URL https://arxiv.org/abs/2508.21113.

Gyeongman Kim, Doohyuk Jang, and Eunho Yang. Promptkd: Distilling student-friendly knowledge for generative language models via prompt tuning. *arXiv preprint arXiv:2402.12842*, 2024.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1317–1327, 2016.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: Towards streamlined distillation for large language models. *arXiv preprint arXiv:2402.03898*, 2024.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024a.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023a.

Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. Mini-gemini: Mining the potential of multi-modality vision language models. *arXiv preprint arXiv:2403.18814*, 2024b.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023b.

Bin Lin, Zhenyu Tang, Yang Ye, Jiaxi Cui, Bin Zhu, Peng Jin, Jinfa Huang, Junwu Zhang, Yatian Pang, Munan Ning, et al. Moe-llava: Mixture of experts for large vision-language models. *arXiv preprint arXiv:2401.15947*, 2024a.

Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. In *Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 26689–26699, 2024b.

Dongyang Liu, Renrui Zhang, Longtian Qiu, Siyuan Huang, Weifeng Lin, Shitian Zhao, Shijie Geng, Ziyi Lin, Peng Jin, Kaipeng Zhang, et al. Sphinx-x: Scaling data and parameters for a family of multi-modal large language models. *arXiv preprint arXiv:2402.05935*, 2024a.

Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651, 2023a.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023b.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023c.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llavanext: Improved reasoning, ocr, and world knowledge, 2024b.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pp. 216–233. Springer, 2024c.

Yuqi Liu, Bohao Peng, Zhisheng Zhong, Zihao Yue, Fanbin Lu, Bei Yu, and Jiaya Jia. Seg-zero: Reasoning-chain guided segmentation via cognitive reinforcement. *arXiv preprint arXiv:2503.06520*, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Yaofeng Sun, et al. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*, 2024.

Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. *arXiv preprint arXiv:2110.13214*, 2021.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022.

Ben Mann, N Ryder, M Subbiah, J Kaplan, P Dhariwal, A Neelakantan, P Shyam, G Sastry, A Askell, S Agarwal, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 1:3, 2020.

Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 34, pp. 5191–5198, 2020.

Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *Advances in Neural Information Processing Systems*, 24, 2011.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, et al. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*, 2025.

Fangxun Shu, Yue Liao, Le Zhuo, Chenning Xu, Lei Zhang, Guanghao Zhang, Haonan Shi, Long Chen, Tao Zhong, Wanggui He, et al. Llava-mod: Making llava tiny via moe knowledge distillation. *arXiv preprint arXiv:2408.15881*, 2024.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 8317–8326, 2019.

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.

Bin Wang, Fan Wu, Xiao Han, Jiahui Peng, Huaping Zhong, Pan Zhang, Xiaoyi Dong, Weijia Li, Wei Li, Jiaqi Wang, et al. Vigc: Visual instruction generation and correction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 5309–5317, 2024.

Yuqiao Wen, Zichao Li, Wenyu Du, and Lili Mou. F-divergence minimization for sequence-level knowledge distillation. *arXiv preprint arXiv:2307.15190*, 2023.

Shicheng Xu, Liang Pang, Yunchang Zhu, Jia Gu, Zihao Wei, Jingcheng Deng, Feiyang Pan, Huawei Shen, and Xueqi Cheng. Distilling the implicit multi-branch structure in llms' reasoning via reinforcement learning. *arXiv preprint arXiv:2505.16142*, 2025.

Shilin Xu, Xiangtai Li, Haobo Yuan, Lu Qi, Yunhai Tong, and Ming-Hsuan Yang. Llavadi: What matters for multimodal large language models distillation. *arXiv preprint arXiv:2407.19409*, 2024a.

Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen, Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao, and Tianyi Zhou. A survey on knowledge distillation of large language models. *arXiv preprint arXiv:2402.13116*, 2024b.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*, 2024.

Xubing Ye, Yukang Gan, Yixiao Ge, Xiao-Ping Zhang, and Yansong Tang. Atp-llava: Adaptive token pruning for large vision language models. In *Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 24972–24982, 2025.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11975–11986, 2023.

Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. Llava-mini: Efficient image and video large multimodal models with one vision token. *arXiv preprint arXiv:2501.03895*, 2025.

Jiaxing Zhao, Xihan Wei, and Liefeng Bo. R1-omni: Explainable omni-multimodal emotion recognition with reinforcement learning. *arXiv preprint arXiv:2503.05379*, 2025.

Baichuan Zhou, Ying Hu, Xi Weng, Junlong Jia, Jie Luo, Xien Liu, Ji Wu, and Lei Huang. Tinyllava: A framework of small-scale large multimodal models. *arXiv preprint arXiv:2402.14289*, 2024.

APPENDIX

In Sec. A, we review existing works relevant to this study. Sec. B provides more implementation details of this study, including training details, benchmark details, comparison methods and the calculation of some metrics used in this study. Additional technical details for our proposed IVA and TPA are presented in Sec. C and Sec. D, respectively. Sec. E presents additional experiments. Sec. F provides claims regarding the use of large language models. Finally, we provide several case studies in Sec. G.

A RELATED WORK

Multimodal Large Language Models. The success of large language models, driven by self-supervised next-token prediction, has significantly advanced multimodal learning by unifying vision and language modalities within the LLM framework. To achieve this, aligning visual and textual representations is essential. Key alignment strategies include: Flamingo (Alayrac et al., 2022), which integrates visual features via cross-attention adapters; BLIP-2 (Li et al., 2023a), employing a Querying Transformer for visual-language pre-alignment; and LLaVA (Liu et al., 2023c), which demonstrated that a simple MLP layer suffices for effective modality alignment and is widely adopted in subsequent works (Bai et al., 2025; Chen et al., 2024b). Beyond LLMs, MLLMs also rely on pre-trained vision encoders to process vision input. Thus, (Li et al., 2024b) introduces a more powerful vision encoder, and (Huang et al., 2025) supports higher-resolution inputs. Furthermore, MLLM capabilities are expanding beyond text generation to tasks like segmentation (Liu et al., 2025) and detection (Shen et al., 2025). Recent works (Zhao et al., 2025; Jiang et al., 2025) introduce the reasoning capabilities into the MLLMs. In addition, considering the substantial computational demands of MLLMs, significant efforts (Chu et al., 2024a; Zhou et al., 2024; Zhang et al., 2025) are also focused on improving model efficiency. This work also focuses on compact MLLMs and develops a more powerful knowledge distillation method for transferring knowledge in large-scale MLLMs.

Knowledge Distillation for LLM. Recent years have witnessed remarkable successes in LLMs trained on extensive datasets with numerous parameters. However, their substantial computational requirements limit deployment in resource-constrained scenarios, motivating extensive research into model compression via KD (Hinton et al., 2015). KD transfers knowledge from powerful teacher models to compact student models, categorized into black-box (output-only access) and white-box (access to intermediate features and logits) paradigms (Xu et al., 2024b). White-box KD, with access to intermediate features and logits, shows superior knowledge transfer capabilities compared to black-box KD methods that synthesize data through the teacher model (Kim & Rush, 2016; Guha et al., 2025; Hugging Face, 2025). This study primarily focuses on white-box distillation.

Early efforts in LLM distillation centered on refining the distillation objective. For instance, MiniLLM (Gu et al., 2023), f-Distill (Wen et al., 2023) and DistiLLM (Ko et al., 2024) proposed using reverse KL divergence, f-divergence and skew KL divergence, respectively, to better align the student's output distribution with the teacher's. More recent advancements have focused on transferring more complex forms of knowledge. GKD (Agarwal et al., 2024) enables the student to learn from the teacher's rationale on self-generated mistakes. PromptKD (Kim et al., 2024) pioneered the use of prompt tuning to adapt the teacher, making its knowledge more student-friendly. Furthermore, RLKD (Xu et al., 2025) introduced a reinforcement learning framework guided by a novel reward model, allowing the student LLM to internalize the teacher's complex, multi-branch reasoning pathways.

Knowledge Distillation for MLLM. Since LLMs serve as the backbone of MLLMs, KD techniques developed for LLMs provide a foundational basis. However, their direct application is suboptimal, as MLLMs introduce the additional knowledge from a vision encoder and preserving cross-modal alignment. The exploration of MLLM distillation is still in its early stages. A recent study by LLaVADI (Xu et al., 2024a) revealed that features from intermediate layers, attention mechanisms, and token relationships are ineffective for MLLM distillation. Moreover, LLaVA-KD (Cai et al., 2024) established a framework incorporating both multimodal content and relational distillation. To address limitations in student capacity, LLaVA-MoD (Shu et al., 2024) enhances the student's representational power by integrating a Mixture-of-Experts (MoE) architecture. Align-KD (Feng et al., 2025) focuses on modeling the cross-modal alignment process during distillation. Addition-

ally, some research concentrates on distilling vision encoders, such as MoVE-KD (Cao et al., 2025), which utilizes multi-teacher distillation to efficiently compress the vision encoder. Furthermore, our work proposes a novel MLLM distillation framework centered on the interactions in the prefilling and decoding stages.

B Implementation Details

B.1 Training Details

B.1.1 Training Details for Teacher Models

In this study, we employ token-level knowledge distillation to train small-scale MLLMs. To maintain architectural consistency with the target small-scale models and facilitate effective knowledge transfer, we follow established protocols (Liu et al., 2023c) for training large-scale teacher MLLMs. Our teacher models utilize Qwen2-7B and Qwen3-8B as the backbone language models, and following recent best practices, we employ the SigLIP-B/14 (Zhai et al., 2023) as the visual encoder. The vision-language projector consists of a two-layer MLP with GeLU activation.

We adopt the two-stage training paradigm from LLaVA (Liu et al., 2023c): (1) Pretraining stage: Models are trained on the LLaVA1.5-558k caption dataset (Liu et al., 2023c) for one epoch with a learning rate of 10^{-3} and batch size of 256. (2) Fine-tuning stage: Models are trained on the LLaVA-mix-665k dataset (Liu et al., 2023c), which combines caption and VQA data, for one epoch with a learning rate of 2×10^{-5} and batch size of 128. Both stages employ the AdamW optimizer (Loshchilov & Hutter, 2017) with cosine decay learning rate scheduling and warmup, utilizing full parameter fine-tuning. Our implementation builds upon the open-source LLaVA codebase and is conducted on 8 NVIDIA A100 GPUs. Detailed training hyperparameters are provided in Tab. 12.

Table 12: Configuration for training teacher model.

Configuration	Pretraining	Fine-tuning
Trainable components	VL-projector	VL-projector, LLM
Epochs	1	1
Batch size	256	128
Learning rate	1×10^{-3}	2×10^{-5}
LR scheduler type	Cosine	Cosine
Optimizer	AdamW	AdamW
Weight decay	0	0
Warmup ratio	0.03	0.03
BF16	True	True
Model max length	2048	2048
Fine-tuning type	Full	Full
Model parallelism	ZeRO-2	ZeRO-2

B.1.2 Training Details for Student Models

Aligned with the teacher models' two-stage training strategy, student models employ identical hyperparameter configurations for learning rate, training epochs, and optimizer. Two key distinctions exist: First, student models utilize more compact large language architectures, specifically Qwen2-0.5B/1.5B and Qwen3-0.6B/1.7B. Second, during fine-tuning, we introduce auxiliary supervision from the teacher model via knowledge distillation. The training objective thus combines standard supervised fine-tuning loss with our proposed distillation loss.

To address the constrained capacity of compact models, which typically require expanded training corpora, we adopt established small-model training methodologies (Chu et al., 2024b). This motivates our use of an augmented dataset containing 3.6M samples: 1.2M captioning samples during pretraining and 2.4M mixed captioning and VQA samples during fine-tuning. Data sources and functional allocations are detailed in Tab. 13. All experiments were conducted across 16 NVIDIA H20 GPUs.

Table 13: Detailed description for data adopted to train the student models.

Stage	Datasets	#Samples	Description
Pretraining	ShareGPT4V-PT (Chen et al., 2024a)	1.2M	Caption
Fine-tuning	VSR (Liu et al., 2023a)	13K	VQA
	SQA (Lu et al., 2022)	13K	VQA
	Text-VQA (Singh et al., 2019)	35K	VQA
	VIGC (Wang et al., 2024)	37K	VQA
	IConQA (Lu et al., 2021)	107K	VQA
	Visual Dialog (Das et al., 2017)	123K	Conversation
	COCO (Chen et al., 2015)	592K	Caption
	ShareGPT4V (Chen et al., 2024a)	665K	Mixed
	SBU (Ordonez et al., 2011)	844K	Caption
Total	-	3.6M	-

B.2 Benchmark Details

Our comprehensive evaluation encompasses six carefully curated benchmarks designed to assess diverse visual-language understanding and generation capabilities. The key characteristics of each benchmark are detailed below:

GQA (Hudson & Manning, 2019) (*Question Answering on Image Scene Graphs*): A VQA benchmark designed for real-world visual reasoning and compositional question answering, containing 12,578 samples for evaluation.

SQA (Lu et al., 2022) (*Scientific question answering*): A VQA benchmark focusing on scientific tables and text with diverse reasoning types, containing 4,241 samples for evaluation.

TextVQA (Singh et al., 2019) (*Text Visual Question Answering*): A VQA benchmark for visual reasoning based on text in images, containing 5,000 samples for evaluation.

POPE (Li et al., 2023b) (*Polling-based Object Probing Evaluation*): A Hallucination detection benchmark focusing systematically evaluating object hallucination tendencies, containing 8,910 samples for evaluation.

MME (Fu et al., 2024) (*Multimodal Model Evaluation*): Comprehensive evaluation suite covering measures both perception and cognition abilities on a total of 14 subtasks with 2,374 manually curated samples.

MMB (Liu et al., 2024c) (*MultiModal Benchmark*): Large-scale multi-choice VQA benchmark containing questions requiring advanced reasoning across 20 task categories, with 4,377 English questions for MMB.

B.3 Details of Comparison Methods

We evaluate our Align-TI method by distilling MLLMs at two different scales: $\sim$1B and $\sim$2B parameters. Our comparison encompasses both models of similar parameter counts and larger-scale models to demonstrate the efficiency and effectiveness of our approach.

For models with comparable parameter counts ($\sim$1B and $\sim$2B), we compare against a comprehensive set of baselines spanning different training paradigms. These include models trained end-to-end from scratch, such as SPHINX-Tiny (Liu et al., 2024a), Mini-Gemini (Li et al., 2024b), and MoE-LLaVA (Lin et al., 2024a). Additionally, we compare with models developed using specialized knowledge distillation techniques for MLLMs, including MoVE-KD (Cao et al., 2025), Align-KD (Feng et al., 2025), LLaVA-KD (Cai et al., 2024) and LLaVA-MoD (Shu et al., 2024).

To further demonstrate the parameter efficiency of our method, we extend our comparison to larger-scale models. This includes $\sim$3B parameter models such as TinyLLaVA (Zhou et al., 2024), MobileVLM V2 (Chu et al., 2024b), LLaVADI (Xu et al., 2024a), MiniCPM-V-2 (Yao et al., 2024) and VILA (Lin et al., 2024b), as well as $\sim$7B parameter models including LLaVA-1.5 (Liu et al., 2023b),

LLaVA-Next (Liu et al., 2024b), DeepSeek-VL (Lu et al., 2024), LLaVA-OV (Li et al., 2024a) and Qwen2.5-VL (Bai et al., 2025).

B.4 Details of Analysis on Exposure Bias

B.4.1 Training-time and Test-time Accumulated Error

In Fig. 2, we visualize the training-time and test-time accumulated errors to reveal the expanding gap with increasing generation length. This section provides the detailed computation. To ensure analysis over sufficiently long sequences, we construct an evaluation set $\mathcal{D}^e$ by randomly sampling 1K samples from the original training set where response length exceeds 100 tokens. These samples are subsequently removed from the training data, ensuring $\mathcal{D}^e$ follows the same distribution as the training set. The student model analyzed here is trained using Vanilla KD.

The training-time accumulated error $E_{\text{train}}(l)$ measures the cumulative divergence under teacher forcing, where the model is conditioned on the ground-truth prefix $y_{<t}$ at each generation step:

$$E_{\text{train}}(l) = \sum_{t=1}^{l} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^e} \left[D_{\text{KL}}(P_T \| P_S^\theta)(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}) \right]. \tag{9}$$

In contrast, the test-time accumulated error $E_{\text{test}}(l)$ simulates realistic autoregressive inference by conditioning on the model's own generated prefix:

$$E_{\text{test}}(l) = \sum_{t=1}^{l} \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_x^e,\, \boldsymbol{y}_{<t}\sim P_S^\theta} \left[D_{\text{KL}}(P_T \| P_S^\theta)(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}) \right]. \tag{10}$$

The train-test gap illustrated in Fig. 2 corresponds to the difference $E_{\text{test}}(l) - E_{\text{train}}(l)$, which directly quantifies the performance degradation caused by error propagation during inference.

B.4.2 Excess Accumulated Error

Definition B.1 (Excess Accumulation Error)**.** *Given a target distribution P_T and a parameterized student model P_S^θ, the Excess Accumulated Error ($\%\text{ExAccErr}_{\leq}(l)$) (Arora et al., 2022) quantifying exposure bias over sequences is formally defined as:*

$$\%\text{ExAccErr}_{\leq}(l) = \frac{R(l) - E(l)}{E(l)} \times 100\%, \tag{11}$$

where $R(l)$ denotes the accumulated regret of imitating the teacher's generation logic up to l time steps, and $E(l)$ is the baseline error conditioned on the oracle context sampled from teacher distribution P_T. A value near zero implies mitigation of exposure bias.

The excess accumulated error is estimated using the 1K dataset $\mathcal{D}_x^e$ described in Sec. B.4.1. Here, $R(l)$ represents the accumulated teacher-student error up to generation step l, conditioned on low-quality prefixes sampled from the student model P_S^θ, and is estimated using KL divergence:

$$R(l) = \sum_{t=1}^{l} \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_x^e,\boldsymbol{y}_{<t}\sim P_S^\theta} \left[D_{\text{KL}}(P_T \| P_S^\theta)(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}) \right] \tag{12}$$

Similarly, $E(l)$ denotes the baseline teacher-student error up to generation step l, conditioned on oracle contexts sampled from the teacher distribution P_T:

$$E(l) = \sum_{t=1}^{l} \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_x^e,\boldsymbol{y}_{<t}\sim P_T} \left[D_{\text{KL}}(P_T \| P_S^\theta)(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}) \right] \tag{13}$$

Thus, $\%\text{ExAccErr}_{\leq}(l)$ quantifies the relative error induced by exposure bias. Under ideal conditions where exposure bias is effectively mitigated, the student model should exhibit nearly identical distribution gaps regardless of the source model generating the response, resulting in $\%\text{ExAccErr}_{\leq}(l) \to 0$. Notably, due to the uncertain relationship between $R(l)$ and $E(l)$, $\%\text{ExAccErr}_{\leq}(l)$ can assume negative values. A negative value indicates that the distribution gap between student and teacher becomes particularly small when conditioned on student-generated responses, while a larger gap exists when conditioned on teacher-generated oracle responses. This phenomenon suggests the persistent presence of exposure bias.

B.5 Calculation Details for IRS.

The Instruction-Relevance Score (IRS) is formally defined as the expected cosine similarity between the vision token importance vectors extracted from attention maps for two different inputs. We estimate this expectation by constructing a set of 1K input pairs, each containing two different queries. Our empirical results demonstrate that the IRS is a stable metric that converges rapidly with a modest number of sample pairs.

B.6 Implementation details for TPA.

Algorithm 1 outlines the procedure for Transition Probability Alignment (TPA), providing further implementation details.

Algorithm 1 Transition Probability Alignment (TPA)

Input: Frozen teacher model $P_T(\boldsymbol{y}|\boldsymbol{x})$
Student model $P_S^\theta(\boldsymbol{y}|\boldsymbol{x})$ with learnable parameters θ.
Training dataset $\mathcal{D}$
Hyperparameters: number of sampled tokens d, learning rate η
repeat
  Sample a mini-batch $\mathcal{B} \sim \mathcal{D}$.
  Initialize total losses $\mathcal{L}_{\text{kd}}(\theta) \leftarrow 0$ and $\mathcal{L}_{\text{tpa}}(\theta) \leftarrow 0$.
  for each $(\boldsymbol{x}, \boldsymbol{y}^{\mathcal{D}})$ in $\mathcal{B}$ **do**
    Initialize augmented sequence $\hat{\boldsymbol{y}} \leftarrow \emptyset$.
    Perform single forward pass of P_S^θ on $(\boldsymbol{x}, \boldsymbol{y}^{\mathcal{D}})$.
    for $k = 1$ to $|\boldsymbol{y}^{\mathcal{D}}|$ **do**
      Sample d candidate tokens $\{y_{k+1}^{(i)}\}_{i=1}^{d} \sim P_S(\cdot \mid \boldsymbol{x}, \boldsymbol{y}_{<k}^{\mathcal{D}})$
      Concatenate tokens: $\hat{\boldsymbol{y}} \leftarrow \hat{\boldsymbol{y}} \circ y_k^{\mathcal{D}} \circ \left[y_{k+1}^{(1)}, \ldots, y_{k+1}^{(d)}\right]$
    end for
    Construct the ribbon attention mask M for $\hat{\boldsymbol{y}}$ ▷ Ensures parallel causal paths, Fig. 5
    Perform single forward passes of P_T and P_S^θ on $(\boldsymbol{x}, \hat{\boldsymbol{y}})$ using attention mask M.
    Compute the initial state alignment loss (Vanilla KD): ▷ Eq. 2

$$\mathcal{L}_{\text{kd}}(\theta) \leftarrow \mathcal{L}_{\text{kd}}(\theta) + \sum_{k=1}^{|\boldsymbol{y}^{\mathcal{D}}|} D_{\text{KL}}\big(P_T \parallel P_S\big)(y_k \mid \boldsymbol{x}, \boldsymbol{y}_{<k}^{\mathcal{D}})$$

    Compute the transition probability alignment loss: ▷ Eq. 7

$$\mathcal{L}_{\text{tpa}}(\theta) \leftarrow \mathcal{L}_{\text{tpa}}(\theta) + \sum_{k=1}^{|\boldsymbol{y}^{\mathcal{D}}|} \frac{1}{d} \sum_{i=1}^{d} D_{\text{KL}}\big(P_T \parallel P_S\big)(y_{k+1} \mid y_k^{(i)}, \boldsymbol{x}, \boldsymbol{y}_{<k}^{\mathcal{D}})$$

  end for
  Update model parameters: $\theta \leftarrow \theta - \eta \nabla_\theta (\mathcal{L}_{\text{kd}}(\theta) + \mathcal{L}_{\text{tpa}}(\theta))$
until convergence
Return: Distilled student model $P_S^\theta(\boldsymbol{y}|\boldsymbol{x})$

C Additional Details for IVA

C.1 Visualization Analysis for Instruction-to-Vision Attention Map.

As quantitatively illustrated in Fig. 4, the IRS varies significantly across different layers of the teacher model. Specifically, IRS is relatively low in the early layers, decreases to its maximum in the middle layers, and then gradually decreases in the deeper layers. To qualitatively interpret this behavior, we visualize visual token importance maps across layers in Fig. 11. Our analysis reveals a clear evolution of the model's attention mechanism. Initially, in the shallow layers (e.g., Layers 0 and 5), the model demonstrates instruction-agnostic behavior, with attention maps focusing on

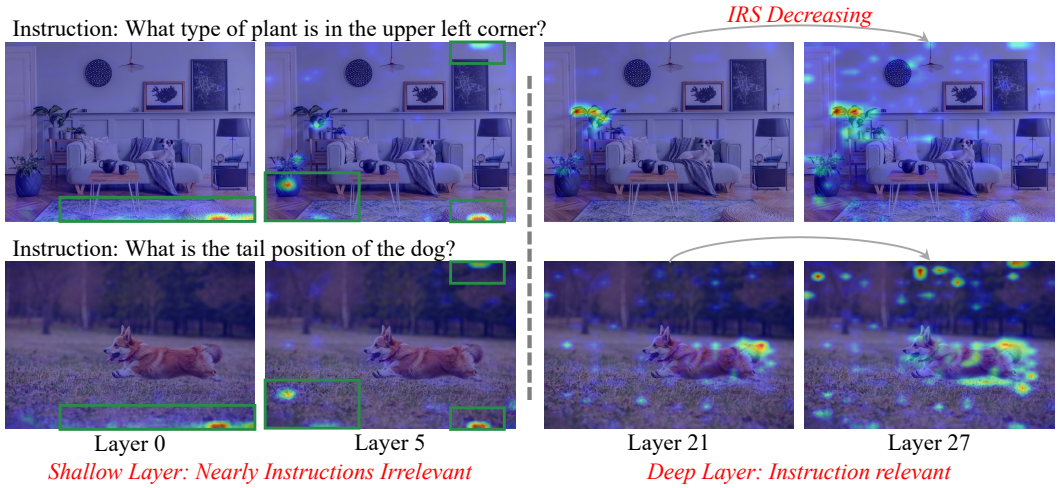


Figure 11: Qualitative analysis of instruction-to-vision attention map evolution across layers. In shallow layers, the attention is largely instruction-agnostic, with different instructions causing the model to attend to similar visual regions (highlighted in gree boxes). In deep layers, the attention maps become highly instruction-specific, with model focusing on instruction-relevant visual regions.

general salient patterns regardless of the instruction. This corresponds to an initial phase of low-level feature extraction, resulting in low IRS. As information propagates to the middle layers (e.g., Layer 21), the attention transitions to an instruction-specific mode, sharply focusing on semantically relevant regions for each task. This semantic filtering process causes a significant drop in IRS to their maximum. Interestingly, towards the final layers (e.g., from Layer 21 to 27), the focused attention begins to diffuse, re-incorporating contextual information from surrounding areas. This final stage of contextual refinement, essential for generating rich responses, leads to a subsequent decreasing in IRS.

C.2 IRS Analysis with Vicuna-7B as LLM.

In Fig. 4 we analyzed the variation of IRS across different layers using Qwen2-7B as the LLM, observing a characteristic trend where IRS initially increases and subsequently decreases, with the peak occurring at approximately three-quarters of the model depth. Extending this analysis in Fig. 10, we evaluate the IRS behavior when using Vicuna-7B as the LLM. Consistent with prior observations, the IRS also exhibits an initial increase followed by a decrease, achieving its maximum near the three-quarters of the layer depth. These results suggest that the observed IRS phenomenon is general across different model architectures.

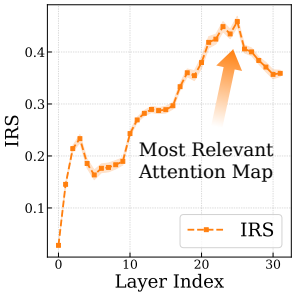


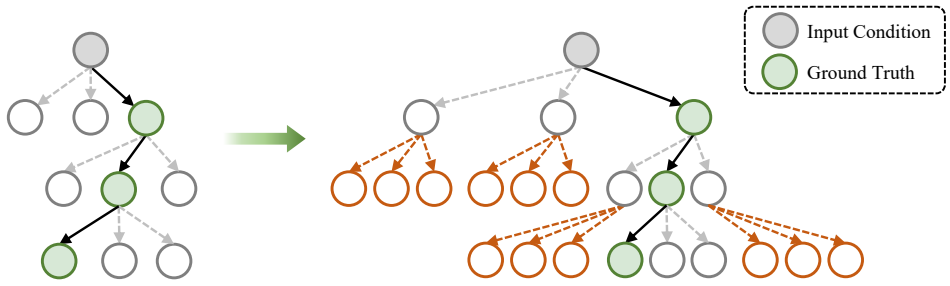
Figure 10: Analysis of IRS.

D Additional Details for TPA

D.1 Derivation of Alignment Scope Expanded in TPA

We analyze the alignment scope by conceptualizing the generation process as a tree structure where each node represents a token state and edges represent transitions between states.

Vanilla KD aligns next-token probabilities conditioned on ground-truth prefixes. At each timestep k, it minimizes $D_{\mathrm{KL}}(P_T \| P_S^\theta)(y_k | x, \boldsymbol{y}_{<k}^{\mathcal{D}})$ over all $|\mathcal{V}|$ possible next tokens. However, the alignment follows a single trajectory determined by the ground-truth sequence $\boldsymbol{y}_{<k}^{\mathcal{D}}$. In the tree representation, Vanilla KD aligns all child nodes at each level but only traverses one path from root to leaf. For a sequence of length L, this results in alignment over $L \times |\mathcal{V}|$ nodes, yielding $O(|\mathcal{V}|)$ path coverage as illustrated in Fig. 12 (A).



(A) Vanilla KD *(B) Transition Probability Alignment (Ours)*

Figure 12: Visualization of alignment spaces achieved by Vanilla KD and Transition Probability Alignment. Starting from the root node, probability distributions at internal nodes and leaf nodes are aligned to transfer knowledge from teacher to student models. Each non-leaf node has children spanning the entire vocabulary $\mathcal{V}$.

TPA extends this alignment by incorporating transition probability matching through $\mathbb{E}_{y_k \sim P_S^\theta} D_{\mathrm{KL}}(P_T \| P_S^\theta)(y_{k+1}|y_k)$. Rather than expanding only from ground-truth tokens, TPA samples candidate tokens and aligns transitions from each sampled state. This corresponds to aligning a $|\mathcal{V}| \times |\mathcal{V}|$ transition matrix at each timestep, where both predecessor and successor tokens span the entire vocabulary. For a sequence of length L, TPA aligns $|\mathcal{V}| + (L-1) \times |\mathcal{V}|^2$ nodes, achieving $O(|\mathcal{V}|^2)$ path coverage as shown in Fig. 12 (B).

D.2 Discussion on Training Efficiency of TPA

Tab. 14 presents the training efficiency analysis of our proposed TPA, specifically evaluating the computational overhead in terms of forward propagation frequency. We consider a dataset consisting of N training samples with an average response sequence length of L, and denote the number of sampled tokens as d. As shown in the comparison, Vanilla KD requires N forward pass for both the student and teacher models per epoch. Utilizing a parallelized calculation strategy with carefully designed attention masks, our proposed TPA necessitates $2N$ forward passes for the student and N for the teacher. This mechanism ensures that each token attends solely to its valid prefix, enabling efficient batch processing.

Without this parallelization, the computational cost becomes prohibitively expensive, scaling to $dLN + N$ for the student and dLN for the teacher, as the output distribution must be computed iteratively for each sampled token. Consequently, our parallelized approach significantly reduces the overhead, ensuring that Align-TI maintains a training efficiency comparable to standard KD methods while achieving superior alignment performance.

Table 14: Comparison of computational overhead in terms of forward passes per epoch. Here, N denotes the number of training samples, L represents the average response sequence length, and d indicates the number of sampled tokens.

Method	Number of Forward Passes	
	Student Model	Teacher Model
Vanilla KD	N	N
TPA (w/ Parallelization)	$2N$	N
TPA (w/o Parallelization)	$dLN + N$	dLN

D.3 Discussion on impact of TPA on Sequence-level Alignment.

In this section, we aim to gain deeper insight by examining how our proposed TPA relates to the ideal objective of sequence-level alignment. The ultimate goal of knowledge distillation is to minimize

the KL divergence between the teacher's and the student's full sequence distributions:

$$\min_{\theta} \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_x} \left[D_{\mathrm{KL}} \left(P_T \parallel P_S^{\theta} \right) \left(\boldsymbol{y}_{1:L} \mid \boldsymbol{x} \right) \right] \tag{14}$$

where L is the sequence length and $\mathcal{D}_x$ denotes the set of input problems. Our analysis reveals that, compared to Vanilla KD, TPA promotes this sequence-level alignment. We present evidence from both theoretical and experimental perspectives.

Theoretically. Strictly optimizing the formulation in Eq. 14 necessitates alignment within a joint probability space of complexity $O(|V|^L)$. In practice, this space is computationally intractable and highly sparse, with numerous combinations being semantically meaningless or contextually irrelevant. As discussed in Sec. D.1, Vanilla KD simplifies this objective by performing alignment in an $O(|V|)$ space. In contrast, TPA operates within an $O(|V|^2)$ space. This implies that TPA exposes the student to richer structural patterns and transition dynamics during training. Such alignment provides a more superior approximation of the full sequence distribution compared to Vanilla KD, thereby facilitating the optimization of Eq. 14.

Empirically. A primary motivation for TPA is to mitigate the Exposure Bias arising from the training-test distribution shift. We employ the Excess Accumulated Error metric (%ExAccErr) to quantify the severity of this bias. Fig. 7 illustrates the trajectory of %ExAccErr as the number of generation steps increases. The results indicate that the model distilled via TPA maintains a remarkably low error rate (ranging between 0 and 10%), which is significantly lower than the approximately 30% observed with Vanilla KD. This substantial reduction suggests that the cumulative error between the student and teacher is effectively suppressed during autoregressive generation. Consequently, the sequences generated by the student exhibit higher fidelity to the teacher's distribution, empirically confirming that Eq. 14 is better optimized under TPA.

Moreover, better alignment with Eq. 14 implies that the student model internalized more teacher's underlying continuous generative logic, thereby improving its ability to capture long-range dependencies and transcending the limitations of simple token-level mimicry.

E Additional Experiments

E.1 Details of Figure 1 (Right).

Fig. 1 (Right) provides a bar chart comparison to highlight the performance differences between our proposed Align-TI, SFT, and Vanilla KD. The data for this visualization is drawn from the comprehensive results presented in Tab. 3 and 4. For a more direct examination, we reproduce the exact numerical values in Tab. 15.

Table 15: Comparison with standard SFT and Vanilla KD.

Model	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
SFT	57.6	59.0	59.1	86.2	66.1	57.9	64.3
Vanilla KD	59.3	59.7	59.2	86.2	65.0	60.4	65.0
Align-TI	**60.4**	**60.7**	**59.9**	**86.8**	**68.9**	**63.2**	**66.7**

E.2 Loss Contribution Analysis.

Table 16 illustrates the impact of each loss term on final performance, providing a clearer understanding of their respective contributions. The overall objective of our proposed Align-TI consists of four components: Supervised Fine-Tuning (SFT) loss $\mathcal{L}_{\mathrm{sft}}$, Instruction-aware Vision Alignment (IVA) loss $\mathcal{L}_{\mathrm{iva}}$, Vanilla KD loss $\mathcal{L}_{\mathrm{kd}}$ and Transition Probability Alignment (TPA) loss $\mathcal{L}_{\mathrm{tpa}}$. Starting from the baseline with only $\mathcal{L}_{\mathrm{sft}}$, adding $\mathcal{L}_{\mathrm{iva}}$ or $\mathcal{L}_{\mathrm{kd}}$ individually yields average improvements of 0.8 and 0.7, respectively, while combining the two results in a larger gain of 1.3. When initiating from a Vanilla KD configuration, incorporating $\mathcal{L}_{\mathrm{iva}}$ and $\mathcal{L}_{\mathrm{tpa}}$ enhances performance by 0.6 and 1.4, respectively. Applying both losses together achieves a total enhancement of 1.7. Notably, the absence of $\mathcal{L}_{\mathrm{kd}}$ in this setup only slightly decreases performance by 0.1, suggesting that the other components can effectively compensate for its omission.

Table 16: Impact of each loss term to the final performance.

Loss	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
$\mathcal{L}_{\text{sft}}$	57.6	59.0	59.1	86.2	66.1	57.9	64.3
$\mathcal{L}_{\text{sft}} + \mathcal{L}_{\text{iva}}$	59.6	58.0	**61.3**	86.5	66.8	58.3	65.1
$\mathcal{L}_{\text{sft}} + \mathcal{L}_{\text{kd}}$	59.3	59.7	59.2	86.2	65.0	60.4	65.0
$\mathcal{L}_{\text{sft}} + \mathcal{L}_{\text{iva}} + \mathcal{L}_{\text{kd}}$	59.8	59.4	60.5	86.5	66.3	61.3	65.6
$\mathcal{L}_{\text{sft}} + \mathcal{L}_{\text{kd}} + \mathcal{L}_{\text{tpa}}$	60.3	**61.0**	59.6	86.5	68.1	63.0	66.4
$\mathcal{L}_{\text{sft}} + \mathcal{L}_{\text{tpa}} + \mathcal{L}_{\text{iva}}$	60.3	60.4	60.6	86.4	68.5	63.1	66.6
$\mathcal{L}_{\text{sft}} + \mathcal{L}_{\text{iva}} + \mathcal{L}_{\text{kd}} + \mathcal{L}_{\text{tpa}}$	**60.4**	60.7	59.9	**86.8**	**68.9**	**63.2**	**66.7**

E.3 Performance Comparison Between Teacher and Student.

In Tab. 17, we compare the performance of the teacher model with that of its distilled student counterparts. The results show that student models trained via distillation achieve performance comparable to that of the teacher. Moreover, increasing the student model's parameter count from 1B to 2B leads to consistent improvements, narrowing the gap with the teacher. Notably, both the 1B and 2B student models exhibit lower hallucination rates than the teacher. On TextVQA, the student models perform on par with or even surpass the teacher. We attribute this to the benchmark's relative simplicity and the teacher's potential over-parameterization in this context. A similar trend of improvement is also observed in LLaVA-MoD (Shu et al., 2024). However, on more complex benchmarks such as SQA, MME, and MMB, a substantial performance gap remains.

Table 17: Comparison between teacher models and distilled student models

Type	LLM	GQA	SQA	TextVQA	POPE	MME	MMB	AVG
Teacher	Qwen2-7B	64.6	80.7	64.1	86.1	78.6	76.0	75.1
Student	Qwen2-1.5B	62.9	71.4	65.1	86.1	75.6	71.8	72.2
	Qwen2-0.5B	60.4	60.7	59.9	86.8	68.9	63.2	66.7
Teacher	Qwen3-8B	64.0	83.4	64.1	86.4	80.4	77.3	76.0
Student	Qwen3-1.7B	62.6	76.5	67.1	86.6	73.4	75.2	73.6
	Qwen3-0.6B	61.2	68.4	64.1	86.9	70.0	67.6	69.7

F The Use of Large Language Models

In preparing this manuscript, we leveraged AI assistants to improve academic writing and debug code. These tools served a function similar to a human editor, dedicated solely to enhancing clarity, grammatical correctness, and code efficiency. All conceptual insights, technical assertions, and critical analyses remain the original contributions of the authors.

G CASE STUDY

In this section, we provide a series of case studies that qualitatively illustrate the effectiveness of our distilled Align-TI-2B model. These examples highlight the model's performance across a wide range of vision-language tasks, including image understanding (Fig. 13), object counting (Fig. 14), chart description (Fig. 15), solving scientific questions (Fig. 16), optical character recognition (Fig. 17), and spatial relationship understanding (Fig. 18).

Case: Image Understanding

User:
Describe this image in detail.

Assistant:
The scene features an airplane flying low over a sea port, with its shadow on the water. The airplane is situated in the middle of the image, creating a striking contrast between the air and water. In the background, there is a large cargo yard filled with numerous colorful shipping containers. The containers are of various sizes and colors, creating an impressive view of the port activities. The arrangement of the containers appears to be organized and well-maintained.

User:
Is this image real? If so, how was it taken?

Assistant:
Yes, this image is real. It was taken using a drone, which allows for an aerial view of the sea port and cargo yard from a high vantage point. The drone captured the airplane flying low over the water, with the shadow on the water, and the large cargo yard filled with various shipping containers in the background.

Figure 13: An example of Align-TI-2B's ability in image understanding.

Case: Object Counting

User:
How many capybaras are there and what are they doing?

Assistant:
There are five capybaras, and they are sitting next to each other near a body of water, possibly a lake.

Case: Object Counting

User:
How many birds are there?

Assistant:
There are six birds in the image.

Figure 14: Examples of Align-TI-2B's capability in object counting.

Case: Chart Description

Percentage of national consumer expenditure by category - 2002

Country	Food/Drinks/ Tobacco	Clothing/ Footwear	Leisure/ Education
Ireland	28.91%	6.43%	2.21%
Italy	16.36%	9.00%	3.20%
Spain	18.80%	6.51%	1.98%
Sweden	15.77%	5.40%	3.22%
Turkey	32.14%	6.63%	4.35%

User:
The table below gives information on consumer spending on different items in five different countries in 2002.

Summarise the information by selecting and reporting the main features, and make comparisons where relevant.

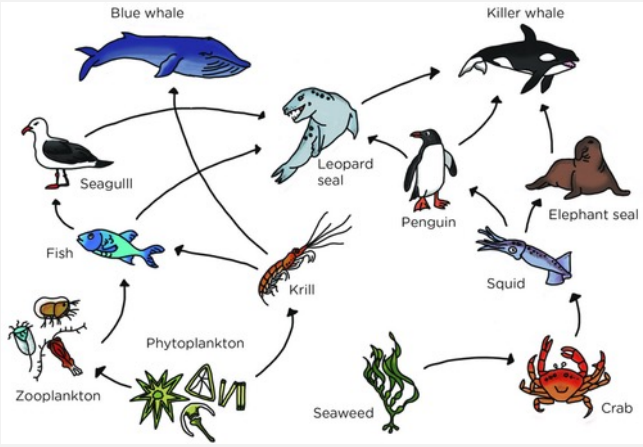
You should write at least 150 words.

Assistant:
For instance, food and drinks are the largest categories, accounting for a significant portion of the overall spending. Clothing, on the other hand, represents a smaller share of the total expenditure. Turkey, Italy, and Spain are among the countries with higher percentages for clothing, while Ireland and Turkey have lower percentages for food and drinks. The United Kingdom has a moderate percentage for clothing but a higher percentage for food and drinks.

This comparison provides insights into the spending patterns and preferences of people in different countries, highlighting the differences in how they invest their money in various categories of consumer goods.

Figure 15: An example demonstrating Align-TI-2B solves chart description problem.

Case: Scientific Question



User:
Which are producers in this food web? Give their names.

Assistant:
In this food web, the producers are the phytoplankton and seaweed.

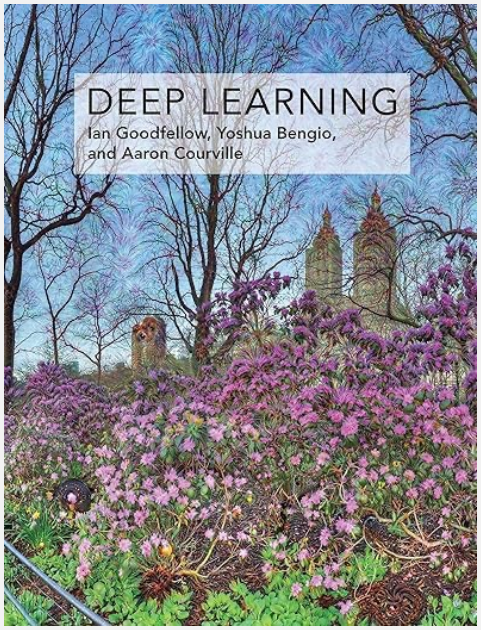
Case: Scientific Question

User:
Which continent is highlighted?

Assistant:
The continent of Asia is highlighted in the map, with red shading indicating its boundaries.

Figure 16: Examples of Align-TI-2B's capability in solving scientific questions.

Case: Optical Character Recognition



User:
How many authors does this book have?

Assistant:
3

User:
Please list all author names.

Assistant:
Ian Goodfellow, Yoshua Bengio, and Aaron Courville

Figure 17: An example of Align-TI-2B performing Optical Character Recognition (OCR).

Case: Spatial Relationships

User:
What is the spatial relation between the frisbee and the man?

Assistant:
The frisbee is above the man, as he is trying to catch it.

Figure 18: An example demonstrating Align-TI-2B's understanding of spatial relationships.