ONLINE IN-CONTEXT DISTILLATION FOR LOW-RESOURCE VISION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

As the field continues its push for ever more resources, this work turns the spotlight on a critical question: how can vision-language models (VLMs) be adapted to thrive in low-resource, budget-constrained settings? While large VLMs offer strong performance, they are impractical to deploy in such settings. Small VLMs, on the other hand, are efficient but typically require costly fine-tuning to close the performance gap with larger models in the deployment domain. Inspired by the in-context learning framework, we propose an online In-Context Distillation (ICD) method, in which a small VLM collaborates with a stronger teacher model at inference time, distilling its knowledge via sparse demonstrations to efficiently bridge the gap between them. Our method is built on an in-depth analysis that identifies the scale and the choice of models for which vision-language ICL is currently feasible, and demonstrates the advantage of ICL over fine-tuning under constrained compute budgets. We enhance our method with a novel cross-modal demonstration selection strategy, teacher test-time scaling to reduce noise, and student uncertainty conditioning to dynamically populate a demonstration pool and minimize teacher queries. Our ICD method significantly boosts the performance of small models (up to 33%) using scarce teacher annotations (as low as 4%), and competes with the teacher's zero-shot performance.

1 INTRODUCTION

Although state-of-the-art vision-language models (VLMs) (Achiam et al., 2023; Team et al., 2023; Li et al., 2024a) continue to scale in size for improved performance, smaller VLMs remain more practical for several real-world applications and users, due to their accessibility and efficiency. In online deployment scenarios with strict constraints on compute, memory, and latency (e.g., edge devices or mobile applications), small models may even represent the only feasible option. However, reducing the model size typically comes at the expense of a drop in performance, leading to a significant gap between small and large models, especially for unseen tasks during pretraining.

Given an online data stream of a specific task, a common strategy to enhance small models is fine-tuning (Zhou et al., 2022; Rasheed et al., 2023), which in general consists of two steps: data annotation and model training. Human annotation is inefficient for real-time applications as it is expensive and time-consuming (Douglas et al., 2023; Ding et al., 2022). On the other hand, fine-tuning a pretrained model for a new task in an online manner is challenging due to not only the additional computational cost, but also the risk of catastrophic forgetting (French, 1999), which degrades the model's original capabilities (Kumar et al., 2022). An alternative is to apply in-context learning (ICL) (Brown et al., 2020; Dong et al., 2022) techniques to enhance the model. ICL leverages demonstrations as context at inference time to guide models to generate task-specific output without parameter update. Nevertheless, demonstrations still require annotations.

We propose a novel online in-context distillation (ICD) framework to bridge the gap between large and small VLMs in an annotation- and training-free manner. As shown in Fig. 1 (left), our ICD consists of two key components: a powerful teacher VLM that generates low-cost, on-the-fly demonstrations without the need for human supervision, and a lightweight student VLM that leverages these demonstrations via in-context learning at inference time. This approach enables the student to distill knowledge from the teacher in an online manner, improving the performance without retraining and thus satisfying the low compute budget requirement, as shown in Fig. 1 (right). Leveraging larger models as an inexpensive instant source of supervision (Wang et al., 2024; Gilardi et al., 2023)

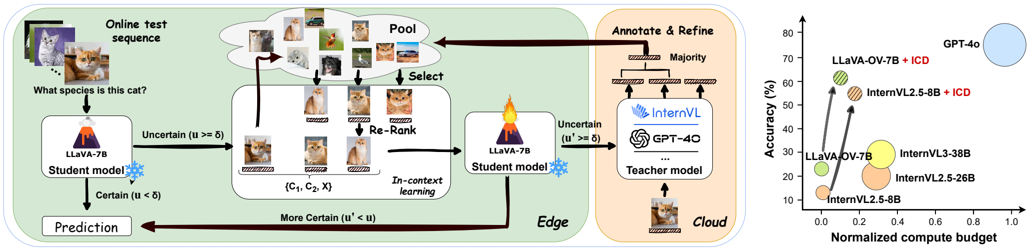


Figure 1: *Left: Overview of our online In-Context Distillation framework. The teacher model in the cloud receives demonstration requests from the student and provides refined answers with test-time scaling. The small student model accesses the dynamically populated pool for answering queries and upon uncertainty, it queries the teacher for new demonstrations. Right: Performance vs. compute budget normalized by that of GPT-4o. Our ICD boosts small VLMs toward the performance of large teacher models with moderate compute overhead.*

is efficient, automatic, and does not bring compute overhead for the student at inference time. We further equip ICD with three modules: (1) test-time scaling techniques (TTS) (Snell et al., 2024) of the teacher to reduce annotation noise and hallucination, (2) cross-modal demonstration retrieval for robust demonstration selection, and (3) uncertainty-based querying that significantly caps annotation budget. Overall, ICD features a modular framework for resource-constrained scenarios of small-large model collaboration, operating on an online pipeline.

A prerequisite for our ICD framework is the effectiveness of ICL in small VLMs. While prior work has demonstrated the success of ICL in large VLMs (Sun et al., 2024; Awadalla et al., 2023) and large language models (LLMs) (Liu et al., 2021; Li et al., 2024c), its utility for small VLMs remains unclear (Zong et al., 2024). To address this gap, we conduct an in-depth analysis of how small VLMs can benefit from ICL. Our findings show that its effectiveness depends heavily on the model's inherent capabilities. Specifically, we find that while models with fewer than 4 billion parameters (denoted as *tiny models*) struggle to benefit from ICL, potentially lacking the fundamental capabilities required for effective ICL, small models (between 4B and 12B parameters) offer a unique advantage of strong ICL capability and a budget-friendly compute. We further trace back ICL failures to a lack of fundamental capabilities such as vision-text alignment, single- and multi-modal reasoning, and visual perception. Interestingly, we find that while the underlying LLM of a VLM can solve certain ICL reasoning tasks, the full VLM fails to do so, suggesting that ICL capabilities in LLMs do not transfer directly to VLMs. We finally show that under a limited compute budget, ICL can bring stronger performance gains compared to finetuning alternatives. These observations serve as the foundation of our efficient and effective approach to boost the performance of a small VLM.

We validate our method with several small VLMs for a range of tasks, including classification, VQA, and image captioning. Our ICD provides a significant performance boost without requiring additional human annotations or model retraining, as well as without frequently querying the heavy-weight teacher. For instance, ICD boosts student performance of 7B model LLaVA-OneVision from 42.6% to 70.8%, compared to zero-shot or TTS-based baselines with as little as 4.4% annotation rate, further outperforming GPT-4o teacher scoring 68.1% Our contributions can be summarized as follows:(1) we conduct an in-depth analysis of ICL capabilities for small VLMs, identifying crucial factors for effective ICL; (2) based on our analysis, we propose a novel online in-context distillation (ICD) framework, designed for real-time compute-constrained scenarios, that combines sparse demonstration querying of a strong model with the inherent ICL capabilities of the student model; and (3) we test our framework across diverse tasks—from classification to generation—demonstrating its broad applicability while boosting base models performance by an average of 14.8% with minimal computate overhead, requiring teacher queries for only 14.7% average of inputs on average.

2 Related Work

In-context learning. ICL was first proposed for enhancing LLM by providing demonstrations at inference time (Dong et al., 2022), resulting from the emerging properties (Wei et al., 2022a;b). Specifically, it prepends demonstrations to the query sample as illustrative guidance, leading to a significant performance boost in the final generation. Various investigations have been conducted to better understand this mechanism (Dai et al., 2022; Akyürek et al., 2022; Wang et al., 2023b; Pan, 2023) and to enhance this capability (Lu et al., 2021; Liu et al., 2024; Chen et al., 2023a).

In-context learning for VLMs. ICL has gained traction in the computer vision community in recent years (Awadalla et al., 2023; Wang et al., 2023a; Chen et al., 2025b) where its potential in enhancing visual understanding is actively explored. The model leverages image-text pairs as context to infer the desired output for the query. While this ICL capability is believed to be inherited from the LLMs in the VLMs, its effectiveness is less clear in VLMs. It has been shown that additional in-context tuning can further improve the ICL capability (Zhao et al., 2023; Chen et al., 2023b). Other works (Zhang et al., 2023b; Zhou et al., 2024b; Baldassini et al., 2024) investigated the strategies to informative select demonstrations for visual ICL and (Chen et al., 2024) discovered a negative contribution of images in the demonstration for visual ICL. A recent benchmark VL-ICL (Zong et al., 2024) is proposed to better evaluate the ICL capabilities of large VLMs, highlighting the challenges in visual ICL. Nevertheless, the analysis for small VLMs remains underexplored.

VLMs distillation. Knowledge distillation (Xu et al., 2024) allows for transferring the knowledge from a large capable model to a smaller efficient model. Such properties are attractive for VLMs where their substantial size and computational demands pose challenges for deployment in resource-constrained environments (Li et al., 2023b; Hu et al., 2024; Wang et al., 2022). Recent works discovered that VLMs models can benefit from the teacher's output, in forms of instructions(Zhou et al., 2024a) or chain of thoughts (Li et al., 2023a). However, existing methods mostly rely on additional training for distillation, which requires annotated data and model training and leads to catastrophic forgetting(French, 1999). A training-free distillation is not yet proposed for VLMs.

3 Can we apply ICL to small VLMs?

In this section, we analyze the ICL capabilities of various families of small VLMs. We focus on three key questions: (1) which models (and at what scale) can benefit from ICL; (2) using a reasoning task as a use case, what failure modes affect Vision-ICL compared to its pure language counterpart; and (3) whether ICL can be advantageous over finetuning under tight compute constraints.

3.1 Definition of In-Context Learning

We consider a general VQA scenario where ICL is performed using a pretrained VLM, denoted as S . Let each triplet (I, Q, A) be a data sample consisting of an image I and a question Q, and A is the corresponding answer. At inference time, the model is given a test query $\hat{x} = (\hat{I}, \hat{Q}) \sim \mathcal{D}$ to return an answer $\hat{A}$, where $\mathcal{D}$ is the test distribution. ICL involves a context set of length k as $\mathcal{G} = \{(I_i, Q_i, A_i)\}_{i=1}^{k}$, where each entry is a demonstration sampled from a support pool $\mathcal{P}$. We propose that a VLM is capable of performing in-context learning if it achieves lower expected task risk with in-context demonstrations compared to without context demonstrations, i.e.,

$$\mathbb{E}_{\hat{x}\sim\mathcal{D}}\left[\ell(S(\hat{x} \mid \mathcal{G}), \hat{y})\right] < \mathbb{E}_{\hat{x}\sim\mathcal{D}}\left[\ell(S(\hat{x}), \hat{y})\right] + \epsilon, \tag{1}$$

where ℓ is a task-specific loss function and $\hat{y}$ is the ground-truth label, and ϵ is a small negative value controlled by significance test. The left-hand side corresponds to in-context prediction and the right-hand side represents zero-shot prediction. This comparative formulation reflects the effectiveness of ICL is equivalent to improving task performance over zero-shot. While ICL has been shown to be effective in improving performance for LLMs and large VLMs, its effectiveness in small VLMs or even tiny VLMs remains unclear (Zong et al., 2024).

3.2 A Feasibility Test of Small VLMs

We investigate ICL capability of VLMs at different scales in a set of tasks. We first categorize VLMs into tiny (N≤4B), small (4B<N≤12B), medium (12B<N≤40B), and large (N>40B), where N denotes the number of model's parameters. We use a standard image similarity (Zhang et al., 2023b) to select demonstrations. Details of the models tested can be found in Appendix A.1 We consider a challenging ICL reasoning benchmark (VL-ICL (Zong et al., 2024)) and a simpler classification task (CUB (Wah et al., 2011)) for small & tiny

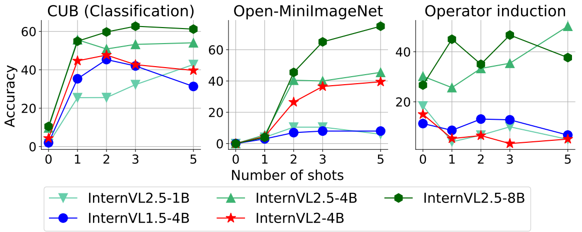


Figure 2: *ICL evaluation for different InternVL sizes and versions. Left to right: tasks are ordered with increasing difficulty. Within the same version (i.e., version 2.5), small VLMs exhibit better ICL capability than tiny VLMs, though tiny VLMs (i.e., size 4B) also improve across versions.*

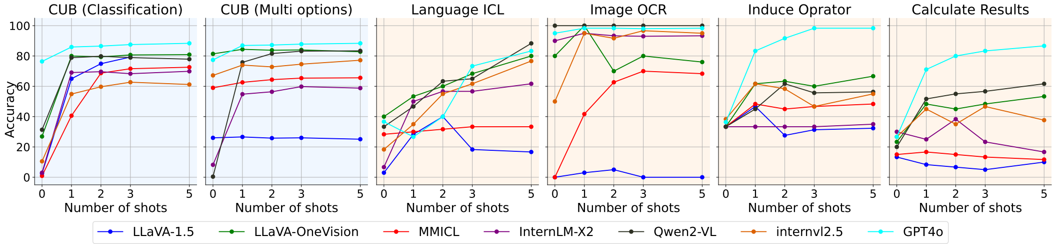


Figure 3: *ICL evaluation to verify models' inherent capabilities (best viewed in color). From left to right: CUB (Classification): image label prediction. CUB (Multi options): (A,B,.. prediction). Language ICL: converted text-only ICL. Image OCR: text recognition from the image. Induce operator: inducing the operator represented by the question mark. Calculate results: final evaluation.*

VLMs to validate our choice of model size. We compare models of the same family to exclude other potential factors and refer to Appendix A.2 for other families. Results are shown in Fig. 2, with tasks ordered from left to right with increasing difficulty. Tiny VLMs struggle to benefit from the demonstrations and to satisfy Eq. 1, especially when the task is more challenging. Meanwhile, we also observe steady improvements in the ICL capability of tiny VLMs across versions (e.g., 4B models), suggesting that ICL may become feasible at this scale with stronger models. In contrast, consistent ICL capability emerges when the model size exceeds 4B. We observe the same property in other model families in Appendix A.2. Therefore, in the rest of the paper we focus on small VLMs, as they exhibit a good balance between efficiency and potential ICL capability.

Take-away message #1: What scale?

In-context reasoning in general scales with size. Currently, small VLMs (4B<N≤12B) achieve a strong balance between efficiency and ICL capability.

Vision Language ICL (VL ICL) Failure Modes: A Case Study. We hypothesize that task learning via context can be hierarchically decomposed, with each subtask relying on specific foundational capabilities. For example, in-context classification depends on vision-text alignment, which small VLMs often acquire during pretraining—explaining their success on CUB classification (Fig. 3). However, when reframed as a multi-option VQA, performance drops sharply for LLaVA-1.5 (Fig. 3), likely due to its limited text reasoning ability needed to map image, options, and labels. Examples are provided in Appendix A.3. To identify the sub-capabilities required for VL ICL and to trace its failure modes, we analyze a representative reasoning task from the VL-ICL benchmark: operator induction. We decompose this task into discrete subtasks, each of which must be successfully combined to solve the full problem. We further compare performance on this task when only textual input is provided, highlighting the differences between Vision-ICL and its pure language counterpart.

The operator induction task involves inferring the mathematical operator represented by a question mark in the demonstrations and applying it to a query. For example, given the demonstration *5 ? 8 = 40*, we infer that *?* represents multiplication, and thus the correct answer to the query *7 ? 3* would be 21. Here, we include both in-context tuned VLMs and small powerful models[1]. To better understand the capabilities and limitations of Vision-ICL, we decompose the task into three subtasks that represent the necessary steps for solving the full problem. First, *Image OCR* involves reading the expression from the image, which tests the model's visual perception ability. Second, *Induce Operator* focuses on identifying the operator without performing the final calculation. Third, *Calculate Results* requires computing the final answer based on the predicted operator. Additionally, we convert all inputs to text and require the VLM to reason using only textual input *(Language ICL)*, in order to assess the reasoning capabilities of the underlying LLM after vision-language pretraining. Illustrative examples of these subtasks are provided in Appendix A.6.

With the models we tested, GPT-4o has the largest size and the best performance in almost every step, which serves as an upper-bound reference for small VLMs. In contrast, the behavior of small VLMs significantly varies. The failure at the final step can be traced back to earlier steps: LLaVA-1.5 fails at image OCR, making it impossible to perform the task. Although InternLM-XComposer2 (InternLM-X2) succeeds in the first step, it fails at inducing the operator. Interestingly, the most recent models, such as LLaVA-OneVision, InternVL2.5 and Qwen2-VL exhibit clearly better ICL capability and succeed to improve overall task performance from demonstrations. We believe this superiority

[1] Ranking from MMBench

stems from their better overall capability. This is especially clear if we compare LLaVA-1.5 with LLaVA-OneVision, two generations of the same model family. Surprisingly, in-context tuned models (InternLM-X2 and MMICL) fail at inducing the operators from demonstrations. A final interesting observation is that while LLaVA-OneVision, InternVL2.5 and Qwen2-VL rival GPT-4o in pure language ICL, a significant performance gap is exhibited when performing the same task in vision language form. Nevertheless, our findings reveal that recent and modern small VLMs demonstrate promising ICL capabilities, which can be harnessed to boost their performance online.

Take-away message #2: VL-ICL challenge

Multimodal in-context reasoning is more complex, relying on the (successful) combination of core capabilities such as perception, modality alignment, and language reasoning.

ICL potential for low-resource adaptation. The inference-only nature of ICL makes it substantially more computationally efficient for injecting new knowledge compared to direct parameter update methods such as fine-tuning (FT). To illustrate this, we compare their performance and compute budgets in Fig. 4. Here, the compute budget is normalized by the total cost of extensive offline fine-tuning under the assumption of ample resources. In practice, additional compute for fine-tuning corresponds to more training epochs, whereas additional compute for ICL corresponds to longer contexts with more demonstrations. Our results show that fine-tuning remains effective for maximizing performance when compute is unconstrained, since ICL does not always scale with longer contexts (Li et al., 2024c). However, under strict compute budgets, ICL delivers markedly greater performance gains than fine-tuning. Furthermore, because ICL requires no explicit model training, it avoids catastrophic forgetting (French, 1999)—the degradation of general capabilities of the model when learning new tasks. Taken together, these results highlight ICL as a more flexible and robust paradigm for low-resource scenarios.

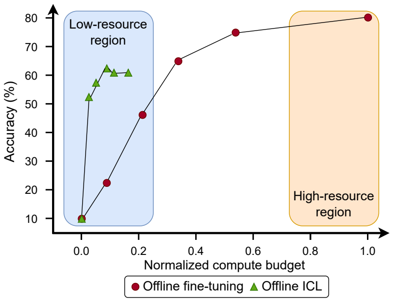


Figure 4: *Comparison of offline fine-tuning v.s. ICL under a compute budget. ICL is a more appealing choice in a low-source environment as it brings significant performance improvement at low computational cost.*

Take-away message #3: Is ICL advantageous for online adaptation?

Compared to fine-tuning, ICL enables rapid performance improvements with minimal overhead, making it well-suited for online adaptation in low-resource settings.

Having shown that recent small VLMs can significantly benefit from in-context examples offering a clear advantage over direct parameter updates in low-resource settings, we now introduce our method for online adaptation using sparse teacher demonstrations.

4 ICD: In-Context Distillation

We consider a real-world deployment setting where test samples are drawn online from the target distribution $\mathcal{D}$ of a deployment environment. In low-resource scenarios, we assume a small student VLM S deployed on edge devices, together with access to a powerful teacher model T, a large capable VLM hosted in the cloud. Unlike conventional large-small model knowledge distillation that relies on further training of the small model to improve its performance (Hinton et al., 2015; Hsieh et al., 2023), we propose to transfer the knowledge from the teacher to the student via ICL. We require no efforts to annotate target tasks in advance, rather demonstrations must be constructed directly from the online test stream. To this end, we gather a dynamic demonstration pool $\tilde{\mathcal{P}}$ by querying the teacher model for annotations on the test samples. Formally, $\tilde{\mathcal{P}} = \{(\hat{I}_i, \hat{Q}_i, A'_i)\}_{i=1}^{P}$, where P is the pool size and $A'_i = T(\hat{x}_i)$ denotes the teacher's response. A discussion on the annotation type is provided in Appendix A.5. In this way, the teacher's knowledge is first embedded into the demonstration pool. At inference, the student retrieves a subset of relevant demonstrations $\mathcal{G} \subseteq \tilde{\mathcal{P}}$ as context to enhance the input, enabling the teacher's knowledge to be distilled into the student's predictions without explicit model updates. This collaborative framework of teacher and student depicts the core structure of our in-context distillation (ICD). Inspired by Eq. 1, we formulate the corresponding optimization problem[2] for our ICD as:

$$\min_{\mathcal{G}} \; \mathbb{E}_{\hat{x}\sim\mathcal{D}}\left[\ell(S(\hat{x} \mid \mathcal{G}), \hat{y})\right] - \mathbb{E}_{\hat{x}\sim\mathcal{D}}\left[\ell(S(\hat{x}), \hat{y})\right]. \tag{2}$$

Our objective is to construct the best demonstration set $\mathcal{G}$ that maximizes the utility of the context when performing ICL. The advantage of this formulation is multi-fold. First, the use of the teacher model to provide demonstrations alleviates the need for human demonstrations, as they can be expensive and time-consuming, as we show in Tab. 3. Moreover, the dynamic knowledge injection via ICL is training-free, which avoids the computational cost of parameter update and the potential risk of forgetting. Lastly, the asynchronous inference of the student and teacher models helps reduce the latency in online scenarios: the student model does not wait for the teacher model's response in real time, as the teacher's role is only to populate the demonstration pool for later use. To the best of our knowledge, this is the first training-free knowledge distillation framework for VLMs.

In the following, we illustrate our modular design to address the following challenges: (1) the quality of the teacher's annotation; (2) the effectiveness of demonstration selection; and (3) the frequency (and thus the cost) of querying the teacher model.

4.1 Improving Annotations with Test-Time Scaling

As a teacher model can still be subject to mistakes and hallucinations, to reduce the noise in our ICD pipeline, we leverage test-time scaling (TTS) techniques to further refine and reduce the noise on the teacher annotations (Muennighoff et al., 2025; Snell et al., 2024). This refinement is beneficial for the student model since ICL for small VLMs is sensitive to the noise in the demonstrations, as shown in Li et al. (2024b). We adopt a simple TTS strategy, namely BestofN (Brown et al., 2024), to improve the reliability of the annotation. Specifically, we perform multiple inferences with the same input and only accept consistent outputs as valid demonstrations.

4.2 Improving Demonstration Selection with Cross-Modal Matching

Common practice for selecting demonstrations uses encoders to extract image and text features for all samples, and select demonstrations based on their uni-modal similarity (image to image and query to annotation) to the query (Li et al., 2024b; Baldassini et al., 2024). We observe that while image-based filtering is effective, text based filtering is less reliable due to the generic nature of the query in many cases, e.g., *What species is this bird?* from the CUB dataset (Wah et al., 2011).

To address this, we introduce a cross-modal matching strategy that leverages both image and text (SelectDemo in Algo. 2). Given a multimodal encoder E, let $F_i, F_t \in \mathbb{R}^{p\times L}$ be the image and text features from the pool $\tilde{\mathcal{P}}$, with p samples and L-dimensional embeddings. Each demonstration candidate is encoded as $f_i = E(I)$, $f_t = E(Q, A')$, and the query is encoded as $\hat{f}_i = E(\hat{I})$, $\hat{f}_t = E(\hat{Q})$. We perform a three-step selection in the following order: image-text $\mathcal{R}_{it}(F_t, \hat{f}_i, K_{it})$, image-image $\mathcal{R}_{ii}(F_i, \hat{f}_i, K_{ii})$, and text-text $\mathcal{R}_{tt}(F_t, \hat{f}_t, K_{tt})$, where $\mathcal{R}(\cdot, K)$ denotes top-K filtering on the input features based on similarity. While a generic query $\hat{Q}$ may limit the utility of F_t in the text-text step $\mathcal{R}_{tt}$, the cross-modal step $\mathcal{R}_{it}$ serves as an effective pre-selection mechanism that compensates for this and leverages the textual information more robustly. Further discussion of the selection mechanism is in Appendix A.15.

Algorithm 1 Online In-Context Distillation

Input: Dataset $\mathcal{D}$, Student S, Teacher T, Pool $\mathcal{P}$, Multimodal Encoder E, Threshold δ
Output: Prediction $\hat{A}$ for each samples x in $\mathcal{D}$

```
for each $x = (I, Q) \in \mathcal{D}$ do
    $\hat{A}, u \leftarrow S(x)$
    ▷ Uncertainty check, see Sec.4.3
    if $u < \delta$ then
        return $\hat{A}$
    else
        ▷ Demonstration selection, see Sec.4.2
        $\mathcal{G} \leftarrow \textsc{SelectDemo}(x, \mathcal{P}, E)$
        $\hat{A}', u' \leftarrow S(x \mid \mathcal{G})$
        if $u' < u$ then
            $\hat{A} \leftarrow \hat{A}'$
        end if
        if $u' \geq \delta$ then
            ▷ Refine with TTS, see Sec.4.1
            $A' \leftarrow \texttt{TTS}(T(x))$
            $\mathcal{P} \leftarrow \mathcal{P} \cup \{(x, A')\}$
        end if
        return $\hat{A}$
    end if
end for
```

[2] Note that $\hat{y}$ is not provided but to illustrate the objective.

4.3 Uncertainty-Aware ICL for Online Evaluation

Querying teacher models can still be expensive and time-consuming, especially for the powerful closed-source model. Thus, we extend ICD with an uncertainty-aware mechanism to reduce the annotation rate. For each incoming sample, we estimate the predictive uncertainty u of the student model S with the average entropy of the generation process. We discuss and ablate our design choice in Appendix A.13. Specifically, let $\{p_1, p_2, \dots, p_J\}$ denote the token-wise probability distributions over the vocabulary $\mathcal{V}$ for a generated output sequence of length J, where each $p_j = \mathrm{softmax}(a_j)$ is computed from the model's output logits a_j at decoding step j. The model's uncertainty can then be estimated as the average entropy over the output sequence:

$$u = \frac{1}{J} \sum_{j=1}^{J} \mathcal{H}(p_j) = -\frac{1}{J} \sum_{j=1}^{J} \sum_{v \in \mathcal{V}} p_j(v) \log p_j(v). \tag{3}$$

A prediction $\hat{A}$ is accepted as-is if u is less than δ, a threshold calibrated on a small validation set annotated by T. Otherwise, we invoke ICL using demonstrations retrieved from $\tilde{\mathcal{P}}$ to obtain a new student prediction $\hat{A}$ and a new uncertainty estimation u'. If the uncertainty is reduced $u' < u$, the student outputs $\hat{A}$ (given the higher likelihood of a correct response). In the case of $u' \geq \delta$ the query is deemed difficult and sent to the cloud for annotation.

Our ICD is illustrated in Fig. 1 (left), and detailed in Algo. 2 and Appendix A.9. We prioritize components that are both simple and effective. This not only improves the computational efficiency but also enhances the accessibility and deployability of our framework for low-resource scenarios. In summary, we present a modular framework with the goal of instant improvement of the student performance at minimal compute/annotation budget combining modules for uncertainty estimation, teacher test-time scaling and cross-modal demonstration selection.

5 Experiments

Datasets. We consider classification-based tasks such as GTSRB (Stallkamp et al., 2012) for traffic sign classification, WikiArt (Tan et al., 2019) for painting style classification, and CUB (Wah et al., 2011) for fine-grained bird classification. Furthermore, we consider the more challenging generative tasks. For instance, we test on Flickr30k (Young et al., 2014) for image captioning, PMC-VQA (Zhang et al., 2023a) for biomedical VQA and VQA-AD (Atakishiyev et al., 2023) for synthetic autonomous driving scenes. In Appendix A.11, we discuss our choice of datasets as opposed to more generic ones such as TextVQA (Singh et al., 2019) and OK-VQA (Marino et al., 2019). Details of prompts used for each dataset are in Appendix A.7.

Models. We choose two representative models that have been shown effective in in-context reasoning from our feasibility tests in Sec. 3, i.e., LLaVA-OneVision and InternVL2.5. Complete tables with more models can be found in Appendix A.10. Unless otherwise stated, we consider the small VLMs (4B<N≤12B) as students, and GPT-4o (Achiam et al., 2023) as the teacher.

Baselines. For self-boosting methods applied to the student model, we include chain-of-thought (*CoT*) prompting (Wei et al., 2022b) and *BestofN* (Brown et al., 2024) as representative test-time scaling techniques. To assess the utility of the teacher annotations, we consider *self-labeling*, similar to the self-generation in (Chen et al., 2023a) for LLMs. In *self-labeling* the pool of demonstration pool is updated with the student model's predictions. We refer to our method as ICD and we consider the *offline ICD* variant where the pool is pre-collected and fully annotated in advance. A detailed discussion of the impact of different sizes of the offline pool is provided in Appendix A.12. We also compare with other VL ICL methods in Sec. 5.2.

Evaluation metrics. We evaluate 3-shot ICL by default (full tables with different shots in Appendix A.10). Accuracy on the test set is used for classification and VQA tasks. For image captioning, we reported the BLEU-4 (Papineni et al., 2002) scores. We report the teacher's annotation rate $T(x)\%$ as the percentage of requested annotations given the size of the test stream. More details on implementation and inference computational costs are in Appendix A.4.

5.1 Main results

In Table 1, we compare different methods on various tasks. Self-boosting methods (i.e., CoT and BestofN) struggle to achieve consistent and significant improvements over 0-shot performance.

Table 1: *Online evaluation results on various datasets. Our ICD method consistently outperforms zero-shot baselines and prior approaches. "LLaVA" refers to LLaVA-OneVision, and "InternVL" refers to InternVL2.5. T(x) denotes the frequency of querying the teacher model. The best results of each model are highlighted in bold. Results of GPT-4o and offline ICD are included as reference.*

	GTSRB		**CUB**		**WikiArt**		**Flickr30k**		**PMC-VQA**		**VQA-AD**		**Average**	
	LLaVA	InternVL	LLaVA	InternVL	LLaVA	InternVL	LLaVA	InternVL	LLaVA	InternVL	LLaVA	InternVL	LLaVA	InternVL
0-shot	42.6	47.2	26.9	10.5	43.1	34.1	26.7	10.7	47.5	34.5	37.0	19.0	37.3	26.0
CoT	32.1 -10.5	41.0 -7.2	18.9 -8.0	12.8 +2.3	36.5 -6.6	35.9 +1.8	20.5 -6.2	13.5 +2.8	48.2 +0.7	30.1 -4.4	28.0 -9.0	24.0 +5.0	30.7 -6.6	26.2 +0.2
BestofN	44.2 +1.6	50.4 +3.2	29.8 +2.9	14.2 +3.7	43.6 +0.5	34.4 +0.3	26.0 -0.7	10.2 -0.5	48.7 +1.2	34.7 +0.2	42.0 +5.0	23.0 +4.0	39.1 +1.8	27.8 +1.8
Self-labeling	41.6 -1.0	47.7 +0.5	27.2 +0.3	11.8 +1.3	40.4 -2.7	32.5 -1.6	24.7 -2.0	11.2 +0.5	48.0 +0.5	35.1 +0.6	41.0 +4.0	21.0 +2.0	37.2 -0.1	26.6 +0.6
ICD	**70.8** +28.2	**61.9** +14.7	**60.3** +33.4	**56.5** +46.0	**49.0** +5.9	**41.3** +7.2	**27.3** +0.6	**13.7** +3.0	**51.1** +3.6	**38.1** +3.6	**47.0** +10.0	**33.0** +14.0	**50.9** +13.6	**40.8** +14.8
↪ *T(x)*	*4.4*	*6.8*	*4.2*	*6.8*	*20.4*	*11.1*	*21.3*	*18.3*	*39.3*	*31.1*	*18.2*	*14.5*	*17.9*	*14.7*
Offline ICD	72.6	66.1	73.1	62.9	53.3	44.1	26.9	14.1	48.5	38.9	50.0	37.0	54.0	43.8
↪ *T(x)*	*100.0*	*100.0*	*100.0*	*100.0*	*100.0*	*100.0*	*100.0*	*100.0*	*100.0*	*100.0*	*100.0*	*100.0*	*100.0*	*100.0*
GPT-4o	68.1		76.4		56.3		24.6		55.8		41.0		53.7	

While such refinements reduce uncertainty and randomness in student predictions, they do not fundamentally enhance the model's capability. Similar limitations arise with self-labeling: despite access to additional data at inference time, it contributes little new information in unseen domains. These observations highlight the necessity of leveraging a strong teacher model as a source of external knowledge to improve student performance. Importantly, our ICD uses the teacher model in a restrained manner through uncertainty thresholding, while achieving close performance to a fully annotated offline ICD. For instance, ICD boosts student performance from 42.6% to 70.8%, compared to zero-shot or TTS-based baselines with as little as 4.4% annotation rate, more importantly outperforming GPT-4o performance (68.1%).

Different tasks present varying levels of difficulty in this scenario. Classification tasks show the good potential of ICL when necessary capabilities are fulfilled, i.e. image-text alignments, as we analyze in Sec. 3.2. Comparing our proposed framework ICD to the model's zero-shot, we observe a consistent boost in performance, which closes the gap between the small VLMs and the powerful teacher, e.g. we enhance LLaVA-OneVision and InternVL2.5 from lower than 30% to around 60% on the CUB dataset, which is significantly closer to 76.4% of GPT-4o. This performance boost enables the model to be practically beneficial in these tasks with no change in the model's parameters. Interestingly, while the teacher model (GPT-4o) does not achieve a better BLEU-4 score than the student LLaVA-OneVision in the image captioning task (24.6 v.s. 26.7 on Flickr30k), it still helps the model to improve via ICL compared to self-labeling. We speculate this is because GPT-4o tends to generate more detailed captions. To validate this, we calculate the string length of each model's caption. While LLaVA-OneVision's average length is 44.2 characters, that of GPT-4o is 62.6, which is 40% longer. These two cases demonstrate that the teacher with a more consistent labeling strategy, especially with TTS, and more detailed answers would provide additional information and reduce the uncertainty of the student. **On average, our ICD brings 14.8% and 13.6% points of performance gain for InternVL2.5 and LLaVA-OneVision across all tasks**, respectively.

5.2 Additional Analysis

Computational analysis. We emphasize the efficiency of our method in achieving strong performance with minimal computational cost, compared with learning-based methods. To estimate cost, we use widely adopted Python packages—`CodeCarbon` and `EcoLogits`—to track carbon emissions during training and inference. These measures are simple to compute and serve as reliable proxies for compute overhead. We compare our ICD approach with LoRA (Hu et al., 2022) fine-tuning in an online learning setting designed for low-resource scenarios. Specifically, we report cumulative carbon emissions and evaluation accuracy after every 64 new annotations obtained from the teacher model, which together form a mini-batch for online learning. Details of online learning can be found in Appendix A.16. Fig. 5 shows that ICD provides an instant performance

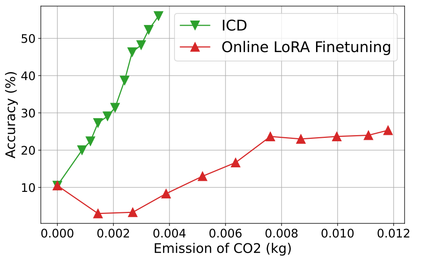


Figure 5: *Accumulative accuracy vs. compute cost in an online scenario. Our ICD excels at this tradeoff.*

boost without the burden of unstable online model training. Moreover, ICD avoids the usual trade-off between compute and performance, delivering the highest performance at minimal cost.

ICL selection mechanism. Using LLaVA-OneVision as the base model, we compare with several existing ICL selection strategies: VICL (Zhou et al., 2024b), PICa (Yang et al., 2022), RICES (Alayrac et al., 2022; Awadalla et al., 2023), SQ, SQ-I (Li et al., 2024b) and MMICES (Chen et al., 2025a). Implementation details of these methods can be found in Appendix A.4. Results are presented in Tab. 2 under the offline ICD setting to guarantee enough demonstrations for selection analysis. Our cross-modal selection consistently achieves the best performance across datasets.

Table 2: *Comparison with different ICL methods with the teacher model's annotation in offline setting. Our cross-modal selection achieves the best performance across datasets.*

Method		**CUB**	**GTSRB**	**VQA-AD**
0-shot		26.9	42.6	37.0
ICL Method	**ICL details**			
ICD	Our cross-modal selection	**73.1**	**72.6**	**50.0**
VICL	Image to caption	65.2	65.2	43.0
PICa	Image to caption & tags	63.5	61.4	40.0
RICES	Image-image selection	72.6	69.3	47.0
SQ	Text-text selection	26.7	39.8	47.0
SQ-I	$\mathcal{R}_{tt} + \mathcal{R}_{ii}$	68.7	70.0	44.0
MMICES	$\mathcal{R}_{ii} + \mathcal{R}_{tt}$	72.9	70.8	48.0

Teacher model generality. Fig. 6 reports the performance gain with different teacher models to highlight the flexibility of our framework. Overall, ICD succeeds in improving the student performance and brings it closer to the corresponding teacher performance, a consistent significant improvement over the student 0-shot performance. Moreover, open-source medium VLMs, which can be locally deployed, serve as an efficient alternative to closed-source models. For instance, InternVL3-38B achieves a comparable performance boost to Gemini-pro-2.5 in both VQA-AD and WikiArt. Notably, our training-free ICD provides significant improvement across different teacher models, showing a flexibility in framework design.

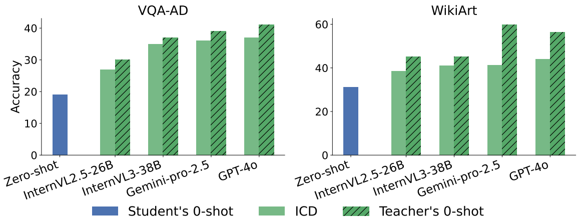


Figure 6: *ICD with different teacher models. We report teacher's 0-shot performance for reference. ICD brings the performance of the student model closer to that of the teacher.*

Practical costs. We include a rough estimation of cost for a WikiArt-scale dataset of 80k images in Tab 3, details are in Appendix A.8. While human annotation is thorough and invaluable, it can be time-consuming. In contrast, automatic annotation can generally be completed within one day. The cost of model annotation is also comparably low. As a reference, annotating an artistic style dataset (Ruta et al., 2022) of 135k images costs in total 160k $. Overall, our ICD strikes a good balance between performance and practical costs.

Table 3: *Practical costs with different annotation strategies. ICD balances well between cost and performance.*

Baseline	**Source of Annotation**	**Cost**	**Time**	**WikiArt**	
				0-shot	3-shot
InternVL-2.5	Human	>4000 $	40 days	34.1	44.7
	Self	< 1 $	4 hours		32.5
	GPT-4o	< 40 $	20 hours		44.1
	InternVL-26B	< 1 $	8 hours		38.6

6 Conclusion

This work explores the potential of enabling small VLMs to perform effectively under low compute budgets. We propose a simple yet effective in-context distillation (ICD) framework to enhance small VLMs, eliminating the need for human annotation and model retraining. Our approach enhances the large teacher model's annotation with test-time scaling, and provides the small student model with a dynamically populated demonstration pool for in-context learning, incorporating cross-modal demonstration selection and uncertainty-aware querying. We also performed an in-depth analysis of ICL, revealing key factors for ICL effectiveness in small VLMs. We validate our ICD approach via a wide range of experiments, in a realistic and challenging online evaluation, showing strong performance gains at low annotation and compute costs. To the best of our knowledge, we are the first to explore an effective annotation- and training-free framework for large to small VLMs distillation. Limitations of our proposed method are discussed in Appendix A.14.

Ethics Statement. The authors confirm that they have read and commit to adhering to the ICLR Code of Ethics.

Reproducibility statement. We provide all necessary implementation details to ensure reproducibility, including the choice of models, the text prompts used for each dataset and task, and detailed descriptions of the techniques employed in this work in the main paper and Appendix. The source code would be released upon acceptance.

Large Language Model Assistance. Large language models were used to check and correct grammatical aspects. The authors have thoroughly reviewed and edited all content and take full responsibility for the final published work.

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.

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? investigations with linear models. *arXiv preprint arXiv:2211.15661*, 2022.

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.

Shahin Atakishiyev, Mohammad Salameh, Housam Babiker, and Randy Goebel. Explaining autonomous driving actions with visual question answering. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1207–1214, 2023.

Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023.

Folco Bertini Baldassini, Mustafa Shukor, Matthieu Cord, Laure Soulier, and Benjamin Piwowarski. What makes multimodal in-context learning work? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1539–1550, 2024.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024.

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

Shuo Chen, Zhen Han, Bailan He, Jianzhe Liu, Mark Buckley, Yao Qin, Philip Torr, Volker Tresp, and Jindong Gu. Can multimodal large language models truly perform multimodal in-context learning?, 2024. URL `https://arxiv.org/abs/2311.18021`.

Shuo Chen, Zhen Han, Bailan He, Jianzhe Liu, Mark Buckley, Yao Qin, Philip Torr, Volker Tresp, and Jindong Gu. Can multimodal large language models truly perform multimodal in-context learning? In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 6000–6010. IEEE, 2025a.

Shuo Chen, Jianzhe Liu, Zhen Han, Yan Xia, Daniel Cremers, Philip Torr, Volker Tresp, and Jindong Gu. True multimodal in-context learning needs attention to the visual context. In *Second Conference on Language Modeling*, 2025b. URL `https://openreview.net/forum?id=n4JdyBGu6T`.

Wei-Lin Chen, Cheng-Kuang Wu, Yun-Nung Chen, and Hsin-Hsi Chen. Self-icl: Zero-shot in-context learning with self-generated demonstrations. *arXiv preprint arXiv:2305.15035*, 2023a.

Yi-Syuan Chen, Yun-Zhu Song, Cheng Yu Yeo, Bei Liu, Jianlong Fu, and Hong-Han Shuai. Sinc: Self-supervised in-context learning for vision-language tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15430–15442, October 2023b.

Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, and Furu Wei. Why can gpt learn in-context? language models implicitly perform gradient descent as meta-optimizers. *arXiv preprint arXiv:2212.10559*, 2022.

Bosheng Ding, Chengwei Qin, Linlin Liu, Yew Ken Chia, Shafiq Joty, Boyang Li, and Lidong Bing. Is gpt-3 a good data annotator? *arXiv preprint arXiv:2212.10450*, 2022.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, et al. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

Benjamin D Douglas, Patrick J Ewell, and Markus Brauer. Data quality in online human-subjects research: Comparisons between mturk, prolific, cloudresearch, qualtrics, and sona. *Plos one*, 18 (3):e0279720, 2023.

Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3 (4):128–135, 1999.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. Chatgpt outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120, 2023.

Juyeon Heo, Miao Xiong, Christina Heinze-Deml, and Jaya Narain. Do llms estimate uncertainty well in instruction-following? *arXiv preprint arXiv:2410.14582*, 2024.

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

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*, 2023.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Yushi Hu, Otilia Stretcu, Chun-Ta Lu, Krishnamurthy Viswanathan, Kenji Hata, Enming Luo, Ranjay Krishna, and Ariel Fuxman. Visual program distillation: Distilling tools and programmatic reasoning into vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9590–9601, June 2024.

Diana Kornbrot. Point biserial correlation. *Wiley StatsRef: Statistics Reference Online*, 2014.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022.

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.

Li Li, Jiawei Peng, Huiyi Chen, Chongyang Gao, and Xu Yang. How to configure good in-context sequence for visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26710–26720, June 2024b.

Liunian Harold Li, Jack Hessel, Youngjae Yu, Xiang Ren, Kai-Wei Chang, and Yejin Choi. Symbolic chain-of-thought distillation: Small models can also "think" step-by-step. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Toronto, Canada, July 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.150. URL `https://aclanthology.org/2023.acl-long.150/`.

Tianle Li, Ge Zhang, Quy Duc Do, Xiang Yue, and Wenhu Chen. Long-context llms struggle with long in-context learning. *arXiv preprint arXiv:2404.02060*, 2024c.

Xuanlin Li, Yunhao Fang, Minghua Liu, Zhan Ling, Zhuowen Tu, and Hao Su. Distilling large vision-language model with out-of-distribution generalizability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2492–2503, October 2023b.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*, 2021.

Yinpeng Liu, Jiawei Liu, Xiang Shi, Qikai Cheng, Yong Huang, and Wei Lu. Let's learn step by step: Enhancing in-context learning ability with curriculum learning. *arXiv preprint arXiv:2402.10738*, 2024.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*, 2021.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pp. 3195–3204, 2019.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*, 2025.

Jane Pan. What in-context learning "learns" in-context: Disentangling task recognition and task learning. Master's thesis, Princeton University, 2023.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Hanoona Rasheed, Muhammad Uzair Khattak, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Fine-tuned clip models are efficient video learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6545–6554, 2023.

Dan Ruta, Andrew Gilbert, Pranav Aggarwal, Naveen Marri, Ajinkya Kale, Jo Briggs, Chris Speed, Hailin Jin, Baldo Faieta, Alex Filipkowski, et al. Stylebabel: Artistic style tagging and captioning. In *European Conference on Computer Vision*, pp. 219–236. Springer, 2022.

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

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks*, pp. –, 2012. ISSN 0893-6080. doi: 10.1016/j.neunet.2012.02.016. URL `http://www.sciencedirect.com/science/article/pii/S0893608012000457`.

Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiying Yu, Yueze Wang, Yongming Rao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative multimodal models are in-context learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14398–14409, 2024.

Wei Ren Tan, Chee Seng Chan, Hernan Aguirre, and Kiyoshi Tanaka. Improved artgan for conditional synthesis of natural image and artwork. *IEEE Transactions on Image Processing*, 28(1):394–409, 2019. doi: 10.1109/TIP.2018.2866698. URL `https://doi.org/10.1109/TIP.2018.2866698`.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Tiannan Wang, Wangchunshu Zhou, Yan Zeng, and Xinsong Zhang. Efficientvlm: Fast and accurate vision-language models via knowledge distillation and modal-adaptive pruning, 2022. URL `https://arxiv.org/abs/2210.07795`.

Xinlong Wang, Wen Wang, Yue Cao, Chunhua Shen, and Tiejun Huang. Images speak in images: A generalist painter for in-context visual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6830–6839, 2023a.

Xinru Wang, Hannah Kim, Sajjadur Rahman, Kushan Mitra, and Zhengjie Miao. Human-llm collaborative annotation through effective verification of llm labels. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, pp. 1–21, 2024.

Xinyi Wang, Wanrong Zhu, and William Yang Wang. Large language models are implicitly topic models: Explaining and finding good demonstrations for in-context learning. *arXiv preprint arXiv:2301.11916*, 1:15, 2023b.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022a.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022b.

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*, 2024.

Yuchen Yang, Houqiang Li, Yanfeng Wang, and Yu Wang. Improving the reliability of large language models by leveraging uncertainty-aware in-context learning. *arXiv preprint arXiv:2310.04782*, 2023.

Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. An empirical study of gpt-3 for few-shot knowledge-based vqa. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 3081–3089, 2022.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the association for computational linguistics*, 2:67–78, 2014.

Xiaoman Zhang, Chaoyi Wu, Ziheng Zhao, Weixiong Lin, Ya Zhang, Yanfeng Wang, and Weidi Xie. Pmc-vqa: Visual instruction tuning for medical visual question answering. *arXiv preprint arXiv:2305.10415*, 2023a.

Yuanhan Zhang, Kaiyang Zhou, and Ziwei Liu. What makes good examples for visual in-context learning? In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 17773–17794. Curran Associates, Inc., 2023b. URL `https://proceedings.neurips.cc/paper_files/paper/2023/file/398ae57ed4fda79d0781c65c926d667b-Paper-Conference.pdf`.

Haozhe Zhao, Zefan Cai, Shuzheng Si, Xiaojian Ma, Kaikai An, Liang Chen, Zixuan Liu, Sheng Wang, Wenjuan Han, and Baobao Chang. Mmicl: Empowering vision-language model with multi-modal in-context learning. *arXiv preprint arXiv:2309.07915*, 2023.

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, 2024a. URL `https://arxiv.org/abs/2402.14289`.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

Yucheng Zhou, Xiang Li, Qianning Wang, and Jianbing Shen. Visual in-context learning for large vision-language models, 2024b. URL `https://arxiv.org/abs/2402.11574`.

Yongshuo Zong, Ondrej Bohdal, and Timothy Hospedales. Vl-icl bench: The devil in the details of multimodal in-context learning. *arXiv preprint arXiv:2403.13164*, 2024.

A Appendix

A.1 Models Cards

In this section, we list the models that we tested in this work in Tab. 4. Unless specified, we refer to the small VLMs (4B<N≤12B) when calling a model family. For instance, when we use the term LLaVA-OneVision, it refers to LLaVA-OneVision-7B.

Table 4: List of different models tested in our work.

Model Name	**Model Size**	**Category**	**Link**
LLaVA-OneVision	0.5B	Tiny	link
InternVL2.5	1B	Tiny	link
InternVL2.5	2B	Tiny	link
InternVL2.5	4B	Tiny	link
Qwen2-VL	2B	Tiny	link
InternVL2.5	8B	Small	link
LLaVA-1.5	7B	Small	link
LLaVA-OneVision	7B	Small	link
Qwen2-VL	7B	Small	link
InternLM-XComposer2	7B	Small	link
MMICL	12B	Small	link
InternVL2.5	26B	Medium	link
GPT-4o	Unknown	Large	link

A.2 Analysis of the Model Size

In Fig. 7, we present the complete comparison with three different model families: InternVL2.5, LLaVA-OneVision, and Qwen2-VL. We observe the same trend: small models benefit from ICL while tiny VLMs struggle to improve with demonstrations.

A.3 Visualization of CUB Experiments

In this section, we present illustrative examples for the two variants of the VQA task based on the CUB dataset: `CUB (Classification)` and `CUB (Multi Options).`

- **CUB (Classification)**

  The prompts follow the standard conversion of classification tasks to VQA tasks, as shown in Sec. A.7. An illustrative example is in Fig. 8
- **CUB (Multi Options)**

  For multi-option classification, we just take the ground truth class label and randomly sample 3 other options to form a 4-option question. An illustrative example is in Fig. 9.

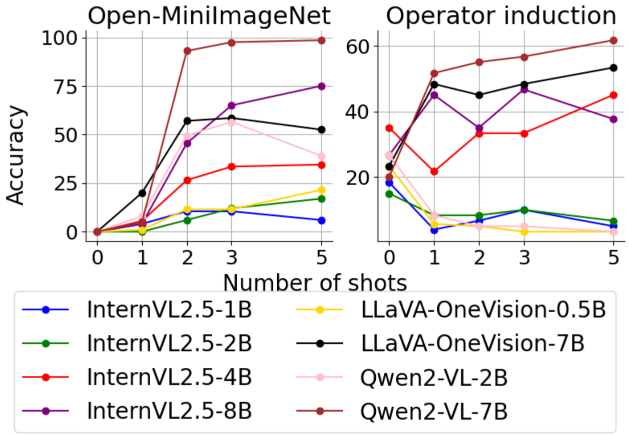
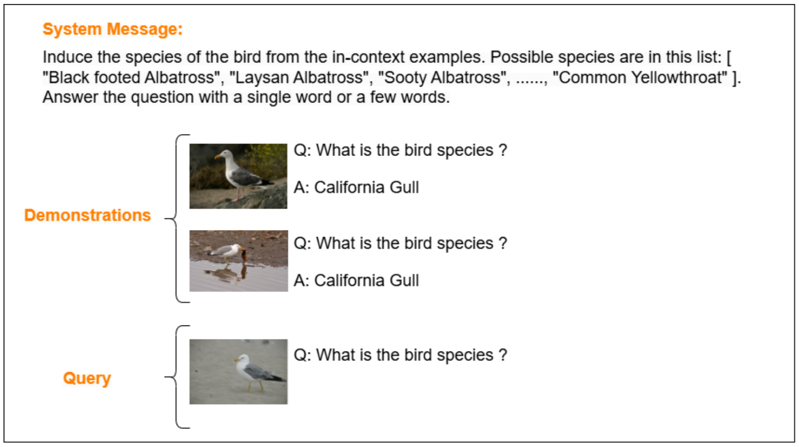


Figure 7: ICL evaluation with respect to the model size. From left to right, the tasks are more challenging to perform ICL. Small VLMs benefit better from emerging ICL capability than tiny VLMs.



System Message:

Induce the species of the bird from the in-context examples. Possible species are in this list: ["Black footed Albatross", "Laysan Albatross", "Sooty Albatross", ..., "Common Yellowthroat"]. Answer the question with a single word or a few words.

Demonstrations

Q: What is the bird species ?

A: California Gull

Q: What is the bird species ?

A: California Gull

Query

Q: What is the bird species ?

Figure 8: Example of ICL CUB (Classification) task. The query sample is prepended with 2 demonstrations in this example.

A.4 Implementation and Computation Details

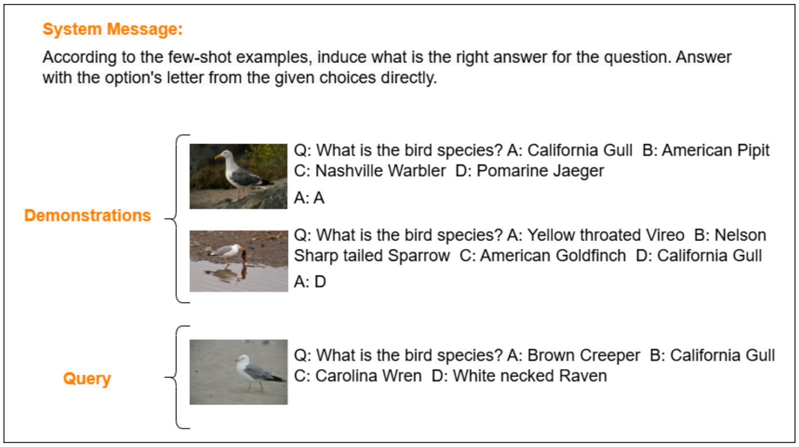
In this section, we present the details of implementation and the discussion of the computation cost of our proposed method.

A.4.1 Implementation

Data. We use the training set of each dataset as the source of the unlabeled dataset, and also to compare the quality of teacher annotations with human annotations.

Models. We use the open-source models (as listed in Appendix A.1) that are available and can be downloaded from the Hugging Face repository. Each model is loaded and set to the `eval` mode for inference. Unless specified, there are no model training or parameter updates on any models.

Inference The student models are tested with zero-shot or in-context learning with the query samples (and demonstrations), which are formatted into an input prompt as described in Appendix A.7. As for selecting demonstrations, we used $K_{it} = 50\% \times |\hat{\mathcal{P}}|$ where $|\hat{\mathcal{P}}|$ is the number of samples in the pool. Moreover, $K_{ii} = 10$ to select the top 10 similar samples with image similarities and $K_{tt} = 5$ to select the samples as demonstration.



System Message:

According to the few-shot examples, induce what is the right answer for the question. Answer with the option's letter from the given choices directly.

Q: What is the bird species? A: California Gull B: American Pipit C: Nashville Warbler D: Pomarine Jaeger

A: A

Q: What is the bird species? A: Yellow throated Vireo B: Nelson Sharp tailed Sparrow C: American Goldfinch D: California Gull

A: D

Q: What is the bird species? A: Brown Creeper B: California Gull C: Carolina Wren D: White necked Raven

Figure 9: Example of ICL CUB (Multi Options) task. The query sample is prepended with 2 demonstrations in this example.

Annotation The teacher model annotates the provided sample with the same prompts presented in Appendix A.7. To further refine the annotation, we apply test-time scaling techniques to reduce the noise and uncertainty in the prediction. We forward 3 times the same input to GPT-4o with its default *temperature* $= 1$ and keep only the consistent predictions. Specifically, for short text such as multi-option VQA or classification, we can simply compare the consistency of the outputs using an exact match. For captioning tasks, we first calculate the BLEU-2 score between each pair of the three predictions. If they are all high in general (> 0.5 in practice), we regard this annotation as consistent.

Online Pipeline The threshold of the uncertainty measure plays an important role in our selective annotation and ICL. In practice, we found that the threshold can be transferred among different tasks. Specifically, for CUB, WikiArt, GTSRB, PMC-VQA, and VQA-AD, the threshold is set to be 0.4. Instead, for Flickr30k, the threshold is set to be 1.6.

Baseline ICL methods. To ensure a fair comparison, all ICL methods use the same multi-modal encoder (i.e., SigLIP: `google/siglip-so400m-patch14-384`) rather than the standard CLIP model in PICa, SQ and SQ-I. This more powerful encoder leads to better demonstration selection. For instance, SQ-I [25] achieves 68.7% with SigLIP vs. 56.2% with CLIP on CUB.

Code We developed our method, analysis, and online pipeline based on the implementation of VL-ICLZong et al. (2024) benchmark.

A.4.2 Computation Cost

In this section, we discuss the computation cost of our proposed framework in two aspects. First, we discuss the computation at inference time that is related to ICL. Moreover, we discuss the broader ecological impact of our proposed method with the computation we can save for an environmentally friendly AI.

Table 5: Average inference time for each sample.

Mode	0-shot	1-shot	2-shot	3-shot	5-shot
Time (s)	0.375	1.325	1.610	1.895	2.490

Inference-Time Computation We report average inference time under consistent configurations as a proxy for computational overhead. Zero-shot evaluation involves a single forward pass, whereas ICL introduces two additional costs. First, for each query, features must be extracted and used to retrieve relevant demonstrations—this involves a forward pass through the multi-modal encoder and one or more matrix multiplications to calculate similarity and identify top candidates (see Sec. 4). This retrieval step is performed only once per query. Second, the inclusion of demonstrations increases the total number of input tokens, adding overhead during inference. As shown in Tab. 5, the jump from 0-shot to 1-shot yields the largest increase in computation time, while the cost scales approximately linearly from 1-shot to 5-shot. All experiments are conducted on a single NVIDIA RTX A6000 GPU with 48 GB of memory.

Computational Efficiency and Ecological Impact Our method is entirely training-free, eliminating the need for computationally intensive fine-tuning, which significantly reduces energy consumption and environmental impact. In contrast to full model retraining, our approach leverages inference-time adaptation through ICL, avoiding repeated gradient-based optimization. Furthermore, we do not query the large teacher model for every test sample. Instead, annotation is performed selectively, conditioned on the student model's uncertainty. This conditional querying strategy substantially reduces the number of teacher invocations, as we show in Tab. 1 of the main paper, resulting in additional computational savings and improved sustainability.

A.5 Teacher's Annotation

While annotating the data with labels is a common practice, we can also ask the teacher to generate other annotations in other forms, as shown in Tab. 6. Additional descriptions bring minor improvement, as they might be complementary to small details that a small VLM cannot perceive. Furthermore, distilling the chain of thought (CoT) in addition to the label provides an additional boost. We conjecture that it triggers the reasoning ability of the small models and reduces the risk of shortcut learning. We leave the exploration of more sophisticated teacher-annotation paradigms for future work.

Table 6: Comparison of different annotation types from the teacher model.

Baseline	**Annotation**	**CUB**				
		Zero shot	1 shot	2 shot	3 shot	5 shot
LLaVA-OneVision	Label	26.9	72.4	73.1	73.1	72.9
	+ Description		72.6	73.2	73.3	72.8
	+ CoT		72.7	74.0	73.7	73.0

A.6 Visualization of Operator Induction Task

In this section, we visualize the original Operator Induction task as in Fig. 10. This is also the Calculate Results task in Sec. 3.2, which requires the model to output the final results. Other sub-tasks are presented as follows.

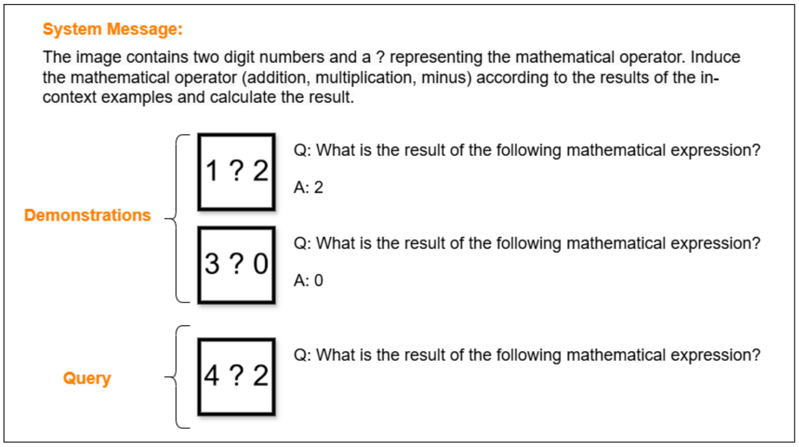
- Text Reasoning: the model is required to infer the operator and calculate the results with purely text input, as shown in Fig. 11
- Image OCR: the model is required to do text recognition to read the expression from the image, as shown in Fig. 12
- Induce Operator: the model is required to infer the operator without calculating the results. Legal answers are provided to help the model better reason, as shown in Fig. 13

A.7 Prompts for Each Dataset

In this section, we list the prompts we used for each dataset. We mostly follow the implementation of VL-ICLZong et al. (2024) while slightly modifying some texts to improve the overall performance. The general form of the prompt is as follows:

```
<System message>


# Concatenation of demonstrations
<Demonstration Image>
<Demonstration Question>
<Demonstration Answer>
```



System Message:

The image contains two digit numbers and a ? representing the mathematical operator. Induce the mathematical operator (addition, multiplication, minus) according to the results of the in-context examples and calculate the result.

Demonstrations

Q: What is the result of the following mathematical expression?

A: 2

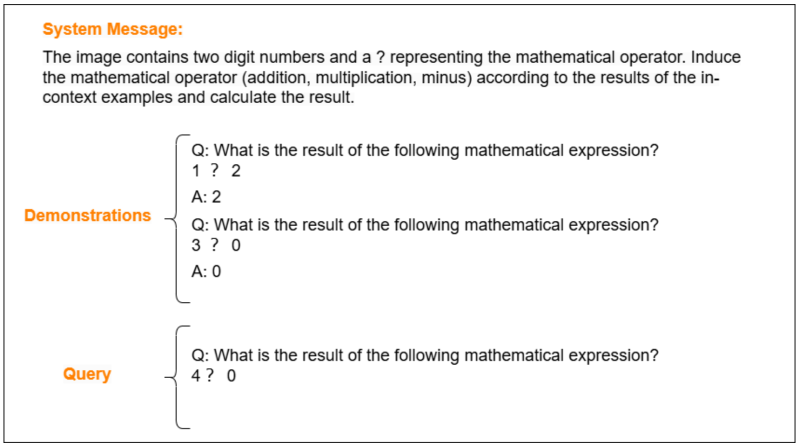
Q: What is the result of the following mathematical expression?

A: 0

Query

Q: What is the result of the following mathematical expression?

Figure 10: Example of ICL for operator induction task. The query sample is prepended with 2 demonstrations in this example.



System Message:

The image contains two digit numbers and a ? representing the mathematical operator. Induce the mathematical operator (addition, multiplication, minus) according to the results of the in-context examples and calculate the result.

Demonstrations

Q: What is the result of the following mathematical expression?
1 ? 2
A: 2

Q: What is the result of the following mathematical expression?
3 ? 0
A: 0

Query

Q: What is the result of the following mathematical expression?
4 ? 0

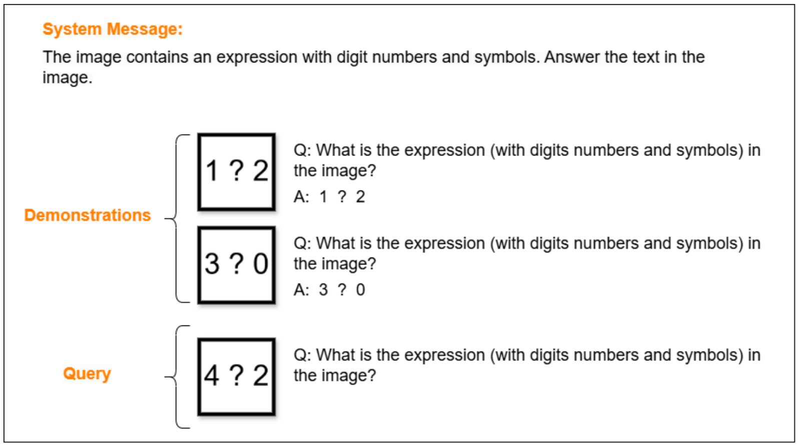
Figure 11: Example of ICL for operator induction **subtask**: Text Reasoning. The query sample is prepended with 2 demonstrations in this example.

```
<Query Image>
<Query Question>
```

The structure might vary slightly for the requirements of each model. We present the standardized question and system message we provide to the models for each task.

A.7.1 Question for Each Task

- **GTSRB Stallkamp et al. (2012)**
  Question: What is the traffic sign?
- **WikiArt Tan et al. (2019)**
  Question: What is the painting style?
- **CUB Wah et al. (2011)**
  Question: What is the bird species ?
- **PMC-VQA Zhang et al. (2023a)**
  Question (with options): What is observed in the control nuclei from P. australis roots in panel (d)? A: Chromatin condensation B: Fragmentation of nuclei C: Lignin labeling D: None of the above.



System Message:

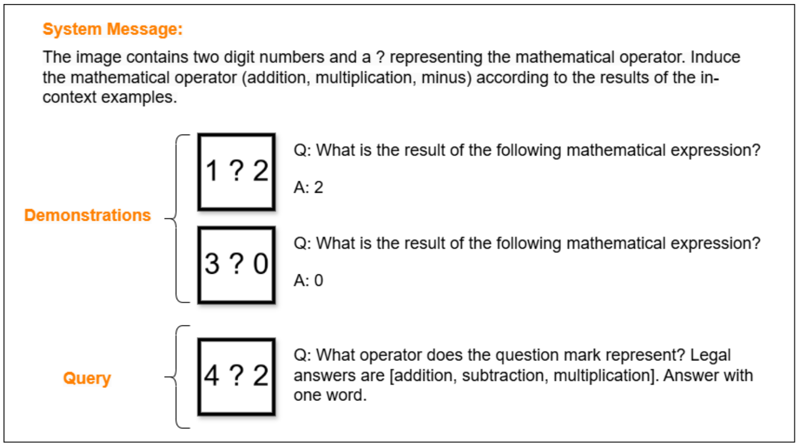
The image contains an expression with digit numbers and symbols. Answer the text in the image.

Q: What is the expression (with digits numbers and symbols) in the image?
A: 1 ? 2

Q: What is the expression (with digits numbers and symbols) in the image?
A: 3 ? 0

Q: What is the expression (with digits numbers and symbols) in the image?

Figure 12: Example of ICL for operator induction **subtask**: Image OCR. The query sample is prepended with 2 demonstrations in this example.



System Message:

The image contains two digit numbers and a ? representing the mathematical operator. Induce the mathematical operator (addition, multiplication, minus) according to the results of the in-context examples.

Q: What is the result of the following mathematical expression?
A: 2

Q: What is the result of the following mathematical expression?
A: 0

Q: What operator does the question mark represent? Legal answers are [addition, subtraction, multiplication]. Answer with one word.

Figure 13: Example of ICL for operator induction **subtask**: Induce Operator. The query sample is prepended with 2 demonstrations in this example.

- **VQA-AD Atakishiyev et al. (2023)**
  Question: What will be the next action of the car?
- **Flickr30k Young et al. (2014)**
  Describe this image in a single sentence.

A.7.2 SYSTEM MESSAGE

- **GTSRB Stallkamp et al. (2012)**
  ```
  Induce the traffic sign from the in-context examples.
  Possible answers are in this list: ["Speed limit (20
  km/h)", "Speed limit (30 km/h)", "Speed limit (50 km/h)",
  "Speed limit (60 km/h)", "Speed limit (70 km/h)", "Speed
  limit (80 km/h)", "End of speed limit (80 km/h)", "Speed
  limit (100 km/h)", "Speed limit (120 km/h)", "No passing",
  "No passing for vehicles over 3.5 tonnes", "Right-of-way at
  the next intersection", "Priority road", "Yield", "Stop",
  "No vehicles", "Vehicles over 3.5 tonnes prohibited",
  "No entry", "General caution", "Dangerous curve to the
  left", "Dangerous curve to the right", "Double curve",
  "Bumpy road", "Slippery road", "Road narrows on the right",
  ```

"Road work", "Traffic signals", "Pedestrians", "Children crossing", "Bicycles crossing", "Beware of ice/snow", "Wild animals crossing", "End of all speed and passing limits", "Turn right ahead", "Turn left ahead", "Ahead only", "Go straight or right", "Go straight or left", "Keep right", "Keep left", "Roundabout mandatory", "End of no passing", "End of no passing for vehicles over 3.5 tonnes"]. Answer the question with a single word or a few words.

- **WikiArt Tan et al. (2019)**

  Induce the painting style from the in-context examples. Possible styles are in this list: [’Expressionism’, ’Pop Art’, ’High Renaissance’, ’Symbolism’, ’Minimalism’, ’Pointillism’, ’Action painting’, ’Abstract Expressionism’, ’Naive Art Primitivism’, ’New Realism’, ’Impressionism’, ’Post Impressionism’, ’Mannerism Late Renaissance’, ’Analytical Cubism’, ’Northern Renaissance’, ’Contemporary Realism’, ’Romanticism’, ’Baroque’, ’Ukiyo e’, ’Early Renaissance’, ’Realism’, ’Fauvism’, ’Art Nouveau Modern’, ’Color Field Painting’, ’Rococo’, ’Cubism’, ’Synthetic Cubism’]. Answer the question with a single word or a few words.

- **CUB Wah et al. (2011)**

  Induce the species of the bird from the in-context examples. Possible species are in this list: ["Black footed Albatross", "Laysan Albatross", "Sooty Albatross", "Groove billed Ani", "Crested Auklet", "Least Auklet", "Parakeet Auklet", "Rhinoceros Auklet", "Brewer Blackbird", "Red winged Blackbird", "Rusty Blackbird", "Yellow headed Blackbird", "Bobolink", "Indigo Bunting", "Lazuli Bunting", "Painted Bunting", "Cardinal", "Spotted Catbird", "Gray Catbird", "Yellow breasted Chat", "Eastern Towhee", "Chuck will Widow", "Brandt Cormorant", "Red faced Cormorant", "Pelagic Cormorant", "Bronzed Cowbird", "Shiny Cowbird", "Brown Creeper", "American Crow", "Fish Crow", "Black billed Cuckoo", "Mangrove Cuckoo", "Yellow billed Cuckoo", "Gray crowned Rosy Finch", "Purple Finch", "Northern Flicker", "Acadian Flycatcher", "Great Crested Flycatcher", "Least Flycatcher", "Olive sided Flycatcher", "Scissor tailed Flycatcher", "Vermilion Flycatcher", "Yellow bellied Flycatcher", "Frigatebird", "Northern Fulmar", "Gadwall", "American Goldfinch", "European Goldfinch", "Boat tailed Grackle", "Eared Grebe", "Horned Grebe", "Pied billed Grebe", "Western Grebe", "Blue Grosbeak", "Evening Grosbeak", "Pine Grosbeak", "Rose breasted Grosbeak", "Pigeon Guillemot", "California Gull", "Glaucous winged Gull", "Heermann Gull", "Herring Gull", "Ivory Gull", "Ring billed Gull", "Slaty backed Gull", "Western Gull", "Anna Hummingbird", "Ruby throated Hummingbird", "Rufous Hummingbird", "Green Violetear", "Long tailed Jaeger", "Pomarine Jaeger", "Blue Jay", "Florida Jay", "Green Jay", "Dark eyed Junco", "Tropical Kingbird", "Gray Kingbird", "Belted Kingfisher", "Green Kingfisher", "Pied Kingfisher", "Ringed Kingfisher", "White breasted Kingfisher", "Red legged Kittiwake", "Horned Lark", "Pacific Loon", "Mallard", "Western Meadowlark", "Hooded Merganser", "Red breasted Merganser", "Mockingbird", "Nighthawk", "Clark Nutcracker", "White breasted Nuthatch", "Baltimore Oriole", "Hooded Oriole", "Orchard Oriole", "Scott Oriole", "Ovenbird", "Brown Pelican", "White Pelican", "Western Wood Pewee", "Sayornis", "American Pipit", "Whip poor Will", "Horned Puffin", "Common Raven", "White necked Raven", "American Redstart", "Geococcyx", "Loggerhead Shrike", "Great Grey Shrike", "Baird Sparrow", "Black throated Sparrow", "Brewhars Sparrow", "Chipping Sparrow", "Clay colored Sparrow", "House Sparrow", "Field Sparrow", "Fox Sparrow", "Grasshopper Sparrow", "Harris Sparrow", "Henslow Sparrow", "Le Conte Sparrow", "Lincoln Sparrow", "Nelson Sharp tailed Sparrow", "Savannah Sparrow", "Seaside Sparrow", "Song Sparrow", "Tree Sparrow", "Vesper Sparrow", "White crowned Sparrow", "White throated Sparrow", "Cape Glossy Starling", "Bank Swallow", "Barn Swallow", "Cliff Swallow", "Tree Swallow", "Scarlet Tanager", "Summer Tanager", "Artic Tern", "Black Tern", "Caspian Tern", "Common Tern", "Elegant Tern", "Forsters Tern", "Least Tern", "Green tailed Towhee", "Brown Thrasher", "Sage Thrasher", "Black capped Vireo", "Blue headed Vireo", "Philadelphia Vireo", "Red eyed Vireo", "Warbling Vireo", "White eyed Vireo", "Yellow throated Vireo", "Bay breasted Warbler", "Black and white Warbler", "Black throated Blue Warbler", "Blue winged Warbler", "Canada Warbler", "Cape May Warbler", "Cerulean Warbler", "Chestnut sided Warbler", "Golden winged Warbler", "Hooded Warbler", "Kentucky Warbler", "Magnolia Warbler", "Mourning Warbler", "Myrtle Warbler", "Nashville Warbler", "Orange crowned Warbler", "Palm Warbler", "Pine Warbler", "Prairie Warbler", "Prothonotary Warbler", "Swainson Warbler", "Tennessee Warbler", "Wilson Warbler", "Worm eating Warbler", "Yellow Warbler", "Northern Waterthrush", "Louisiana Waterthrush", "Bohemian Waxwing", "Cedar Waxwing", "American Three toed Woodpecker", "Pileated Woodpecker", "Red bellied Woodpecker", "Red cockaded Woodpecker", "Red headed Woodpecker", "Downy Woodpecker", "Bewicks Wren", "Cactus Wren", "Carolina Wren", "House Wren", "Marsh Wren", "Rock Wren", "Winter Wren", "Common Yellowthroat"]. Answer the question with a single word or a few words.

- **PMC-VQA Zhang et al. (2023a)**

  According to the few-shot examples, induce what is the right answer for the question. Answer with the option's letter from the given choices directly.

- **VQA-AD Atakishiyev et al. (2023)**

  According to the few-shot examples, induce what the right answer is for the question. Answer with the option's letter from the given choices directly.

- **Flickr30k Young et al. (2014)**

  Induce the concept from the in-context examples. Answer the question with a single sentence that best captions the image.

A.8 Cost and Time Estimation for Data Annotation

We analyze the cost and time required for annotating 80k images using three different strategies: (1) OpenAI's GPT-4o API, (2) locally hosted InternVL2.5 models (InternVL2.5-7B and InternVL2.5-26B), and (3) human annotation.

A.8.1 Cost and Time Estimation for GPT-4o API

GPT-4o API pricing is based on token usage, where input tokens cost 2.50 $ per million tokens and output tokens cost 10.00 $ per million tokens (by February 2025). Assuming a text for input prompt of 50 tokens for annotation instructions and an input of 100 tokens per image with `High` resolution, and the output prompt for a few words as annotation with 10 tokens.

The corresponding cost is calculated as:

$$C_{input} = \frac{(100+50)*80000}{1,000,000} \times 2.50 = 30\ \$ \tag{4}$$

$$C_{output} = \frac{10*80,000}{1,000,000} \times 10.00 = 8.00\ \$ \tag{5}$$

$$C_{total} = C_{input} + C_{output} = 38\ \$ \tag{6}$$

In practice, the process and communication time for one image with GPT-4o-API is around 1 second:

$$T_{total} = 80,000 \text{ sec} \approx 20 \text{ hours} \tag{7}$$

A.8.2 Cost and Time Estimation for InternVL2.5 models on RTX A6000

For a locally hosted InternVL2.5-8B model on an RTX A6000 GPU, the average inference time per sample is empirically estimated as 0.2 seconds.

For 80,000 images, the total processing time is:

$$T_{total} = 80,000 \times 0.2 \approx 4 \text{ hours} \tag{8}$$

The power consumption of an RTX A6000 system is estimated at 0.4 kW, leading to the following energy costs at 0.15 $ per kWh:

$$C_{total} = 0.4 \times 4 \times 0.15 = 0.24\ \$ \tag{9}$$

Similarly, for a medium-size VLM InternVL2.5-26B, the processing time for each image is empirically estimated as 0.4 seconds. Therefore, the time and cost would be a factor of 2 of that of InternVL2.5-8B. Thus,

$$C_{total} = 0.48\ \$, T_{total} = 8 \text{ hours} \tag{10}$$

A.8.3 Cost and Time Estimation for Human Annotation

The average annotation time per image is assumed to be 15 seconds. Thus, the total annotation time for one annotator is:

$$T^{human}_{total} = 80,000 \times 15 = 1,200,000 \text{ sec} = 333 \text{ hours} \tag{11}$$

Considering full-time jobs with 8 working hours per day, the annotation process needs more than 40 days for one annotator.

The cost of human annotation varies by region, we take 13 $ per hour as a rough estimation. The total cost for human annotation is thus:

$$C_{human} = 333 \times 13 = 4,000\ \$ \tag{12}$$

A.9 Algorithm

In this section, we present the full and detailed algorithm of our online ICD pipeline (see Algorithm. 2) with the definition of the two functions SelectDemo (see Algorithm. 3) and TTS (see Algorithm. 4).

Algorithm 2 Online In-Context Distillation

Input: Dataset $\mathcal{D}$, Student S, Teacher T, Pool $\mathcal{P}$, Multimodal Encoder E, Threshold δ
Output: Prediction $\hat{A}$ for each samples x in $\mathcal{D}$

```
for each $x = (I, Q) \in \mathcal{D}$ do
    Predict with Zero-shot and estimate uncertainty $\hat{A}, u \leftarrow S(x)$
    ▷ Uncertainty check, see Sec. 4.4
    if $u < \delta$ then
        return $\hat{A}$
    else
        ▷ Demonstration selection, see Sec. 4.3
        Select the demonstration set $\mathcal{G} \leftarrow \textsc{SelectDemo}(x, \mathcal{P}, E)$
        Predict with ICL and estimate uncertainty $\hat{A}', u' \leftarrow S(x \mid \mathcal{G})$
        if $u' < u$ then
            Update the prediction $\hat{A} \leftarrow \hat{A}'$
        end if
        if $u' \geq \delta$ then
            ▷ Refine with TTS, see Sec. 4.2
            Use teacher model to annotate and refine $A' \leftarrow \text{TTS}(T(x))$
            Concatenate the annotated sample to the pool $\mathcal{P} \leftarrow \mathcal{P} \cup \{(x, A')\}$
        end if
        return $\hat{A}$
    end if
end for
```

Algorithm 3 $\textsc{SelectDemo}(x, \mathcal{P}, E)$

Input: Sample x, Pool $\mathcal{P}$, Encoder E, Selection Number K_{ii}, K_{tt}, K_{it}
Output: Selected demonstration samples

```
▷ Extract features
$(F_i, F_t) \leftarrow E(\mathcal{P})$
$(\hat{f}_i, \hat{f}_t) \leftarrow E(\hat{x})$
▷ Rank by cross-modal similarity
Selecting top K from the Index $\leftarrow$ TopK$(CosineSimilarity(F_t, \hat{f}_i), K_{it})$
Filter features: $F_i, F_t \leftarrow F_i[\text{Index}], F_t[\text{Index}]$
▷ Rank by image-image similarity
Index $\leftarrow$ TopK$(CosineSimilarity(F_i, \hat{f}_i), K_{ii})$
Filter features: $F_i, F_t \leftarrow F_i[\text{Index}], F_t[\text{Index}]$
▷ Rank by text-text similarity
Index $\leftarrow$ TopK$(CosineSimilarity(F_t, \hat{f}_t), K_{tt})$
Filter features: $F_i, F_t \leftarrow F_i[\text{Index}], F_t[\text{Index}]$
return $\mathcal{P}[\text{Index}]$
```

Algorithm 4 $\text{TTS}(T, x)$

Input: Teacher model T, Data sample x, Iteration K
Output: Annotation $\hat{A}$

```
$\mathcal{A} \leftarrow \emptyset$
▷ Best-of-N with multiple inferences
for $k = 1$ to $K$ do
    $A_k \leftarrow T(x)$
    $\mathcal{A} \leftarrow \mathcal{A} \cup \{A_k\}$
end for
▷ Keep only consistent prediction
if Consistent prediction: $\forall i \neq j, A_i, A_j \in \mathcal{A}, A_i = A_j$ then
    return $A_0$
end if
```

Table 7: Evaluation of ICL classification tasks. Our ICD consistently improves over zero-shot performance. Baseline methods are tested with zero-shot and up to 5 demonstrations for ICL. The performance of fine-tuned models and GPT-4o is included as a reference. The best results (excluding oracle) of each model are highlighted in bold. Oracle labels represent the upper bound with human annotations.

Baseline	Method	GTSRB					WikiArt					CUB				
		Zero shot	1 shot	2 shot	3 shot	5 shot	Zero shot	1 shot	2 shot	3 shot	5 shot	Zero shot	1 shot	2 shot	3 shot	5 shot
GPT-4o	-	68.1	-				56.3	-				76.4	-			
InternVL2.5	Fine-tuning	47.2	With oracle labels: 86.7 With teacher labels: 79.4				34.1	With oracle labels: 60.7 With teacher labels: 58.7				10.5	With oracle labels: 84.8 With teacher labels: 83.5			
MMICL	Oracle	24.1	50.8	77.4	79.3	79.0	32.8	34.3	42.3	44.2	36.2	0.9	40.6	68.7	71.6	72.6
	Self-labeling		23.4	27.2	27.5	25.1		30.5	32.3	32.9	28.7		7.0	7.5	7.6	7.8
	ICD		46.5	65.6	**68.2**	67.3		33.4	41.9	**43.0**	34.3		38.0	62.4	**65.4**	64.5
LLaVA-1.5	Oracle	2.6	85.2	85.4	85.3	84.3	16.4	49.4	47.4	48.5	47.3	2.5	65.0	74.9	79.3	-
	Self-labeling		2.0	2.3	2.2	2.1		17.3	17.0	18.3	17.8		3.0	2.6	2.7	-
	ICD		70.6	72.0	72.5	**72.7**		49.5	**49.9**	49.7	47.2		57.8	67.6	**70.6**	-
InternVL2.5	Oracle	47.2	57.3	69.6	78.8	80.1	34.1	42.4	41.3	44.7	43.0	10.5	54.9	59.7	62.7	61.2
	Self-labeling		44.6	46.1	47.7	48.3		33.2	32.6	32.5	30.4		10.5	10.7	11.8	11.3
	ICD		64.9	63.5	66.1	**66.6**		41.3	43.7	**44.1**	42.7		54.6	58.3	**62.9**	57.2
LLaVA-OneVision	Oracle	42.6	85.5	85.9	85.6	86.1	43.1	54.2	54.2	55.8	54.7	26.9	80.0	79.4	80.7	80.9
	Self-labeling		42.2	41.9	41.6	41.8		40.3	40.0	40.4	38.9		27.6	27.1	27.2	26.5
	ICD		72.0	72.5	72.6	**74.3**		52.0	51.8	**53.3**	52.7		72.4	**73.1**	**73.1**	72.9

Table 8: Evaluation of ICL generative tasks. Our ICD effectively enhances recent small VLMs. Baseline methods are tested with zero-shot and up to 5 demonstrations for ICL. The performance of fine-tuned models and GPT-4o is included as a reference. The best results (excluding oracle) of each model are highlighted in bold. Oracle labels represent the upper bound with human annotations.

Baseline	Method	Flickr30k					PMC-VQA					VQA-AD				
		Zero shot	1 shot	2 shot	3 shot	5 shot	Zero shot	1 shot	2 shot	3 shot	5 shot	Zero shot	1 shot	2 shot	3 shot	5 shot
GPT-4o	-	24.6	-				55.8	-				41.0	-			
InternVL2.5	Fine-tuning	10.7	With oracle labels: 16.9 With teacher labels: 17.3				34.5	With oracle labels: 54.1 With teacher labels: 52.8				36.0	With oracle labels: 60.0 With teacher labels: 54.0			
MMICL	Oracle	24.3	28.6	29.0	28.5	27.7	34.7	35.9	36.0	34.3	34..4	22.0	20.0	28.0	29.0	32.0
	Self-labeling		23.5	22.4	21.5	19.0		34.2	33.2	32.5	32.3		23.0	27.0	27.0	31.0
	ICD		**28.9**	28.8	27.7	27.3		34.6	**35.4**	31.7	33.2		24.0	32.0	31.0	**35.0**
LLaVA-1.5	Oracle	25.1	20.9	23.0	23.1	23.6	19.1	21.8	20.3	18.8	20.8	32.0	17.0	18.0	19.0	20.0
	Self-labeling		18.5	20.4	21.4	21.0		30.0	33.3	**34.9**	31.8		18.0	19.0	17.0	24.0
	ICD		21.3	22.4	22.2	**22.5**		21.6	20.3	18.6	21.2		**28.0**	24.0	**28.0**	19.0
InternVL2.5	Oracle	10.7	12.7	12.7	12.1	11.7	34.5	42.1	39.1	39.2	38.2	19.0	38.0	40.0	43.0	37.0
	Self-labeling		11.5	11.8	11.2	11.6		37.5	36.4	35.1	34.9		10.0	19.0	21.0	21.0
	ICD		13.6	13.7	**14.1**	13.5		**40.5**	39.9	38.9	35.7		31.0	35.0	**37.0**	33.0
LLaVA-OneVision	Oracle	26.7	23.1	26.5	27.7	28.1	47.5	48.2	49.1	48.3	47.5	37.7	47.0	51.0	58.0	56.0
	Self-labeling		20.7	24.0	24.7	26.0		45.6	48.2	48.0	47.2		41.0	39.0	41.0	43.0
	ICD		22.8	26.1	26.9	**28.2**		48.0	**48.7**	48.5	47.7		49.0	49.0	**50.0**	48.0

A.10 MULTI-SHOT ICL EXPERIMENTS

In this section, we present the experiments with the offline-gathered pool, where the teacher annotates the entire unlabeled set. With the labeled pool, we perform ICL with different numbers of demonstrations to show the benefit of providing more demonstrations and to compare the performance with oracle labels. The results are presented in Tab. 7 and Tab. 8. Our experiments confirm that 3-shot ICL is the most effective choice in general and with more demonstrations (i.e., 5 shots), the performance instead degrades. We attribute this phenomenon to the extended input length with more demonstrations and the potential noise in the demonstrations.

A.11 DISCUSSION OF DATASETS

A.11.1 DESCRIPTION OF DATASETS FOR EVALUATION

Following our analysis on Sec. 3, we consider classification-based tasks to show a good potential of ICL with necessary capabilities fulfilled, i.e. image-text alignments. Furthermore, we consider the more challenging generative tasks, such as VQA and image captioning, under the condition that they designate specific domain knowledge. For classification tasks, GTSRB Stallkamp et al. (2012)

contains over 50,000 traffic sign images across 43 classes for recognition tasks. WikiArt Tan et al. (2019) includes paintings from 27 artistic styles with more than 80k images. CUB Wah et al. (2011) comprises 11,788 bird images from 200 species for fine-grained classification. For generative tasks, PMC-VQA Zhang et al. (2023a) is a biomedical VQA dataset that contains a total of 227k VQA pairs of 149k images. VQA-AD Atakishiyev et al. (2023) focuses on synthetic autonomous driving scenes, covering 5 driving actions with 250 frames for training and 100 frames for testing. Flickr30k Young et al. (2014) features 31,000 images with human-annotated captions for image-text modeling.

A.11.2 Evaluation on Generic Datasets

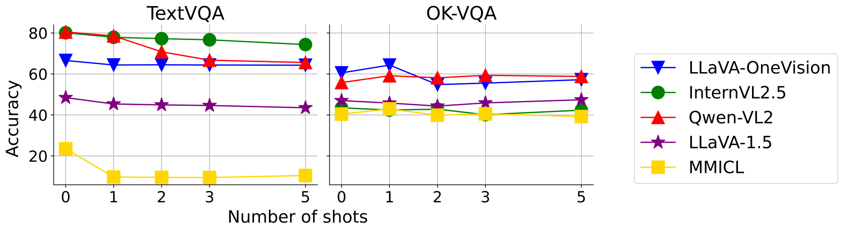


Figure 14: ICL experiments on generic datasets. Most models cannot improve with demonstrations.

Based on our analysis in Sec. 3, we conclude that ICL favors specific tasks over generic tasks. To further justify our finding, we show the ICL experiments on two representative generic VQA datasets: TextVQA Singh et al. (2019) and OK-VQA Marino et al. (2019). TextVQA comprises 45,336 questions with 28,408 images, requiring models to read and reason about text within images. In contrast, OK-VQA includes over 14,000 questions with 10 diverse categories, necessitating external knowledge or commonsense reasoning to provide correct answers. In these experiments, we use the oracle label for the demonstration. The results are shown in Fig. 14. We observe that all models tested struggle to improve with demonstrations, which supports our remark in Sec. 3. As the dataset is generic and diverse, it is more challenging to find semantically meaningful demonstrations that would help the model better understand the task. We leave further investigation of ICL on these challenging datasets as future work.

A.12 Different Size of Unlabeled Set for ICD

In the main experiments, we consider annotating all the training samples to ensure the diversity and relevance of the demonstrations. In practice, there is no need to annotate the entire training set, and one can consider applying our framework whenever a few samples are available from a new domain or task. As shown in Fig. 15, the final performance remains close when we reduce the annotation size from the full set to 25% of the data. However, there is a clear gap between 5% and the full set, which confirms the necessity to have diverse demonstrations for ICL. Moreover, we show in the figure the performance of our online ICD, where the percentage of the gathered pool is automatically collected with the uncertainty-aware selection. We also report the online ICD results in the same figure, with respect to the reduced annotation rate $T(x)$. It is interesting to see that our online ICD achieves significantly higher accuracy with fewer data annotated, which confirms the effectiveness of an active and selective annotation strategy.

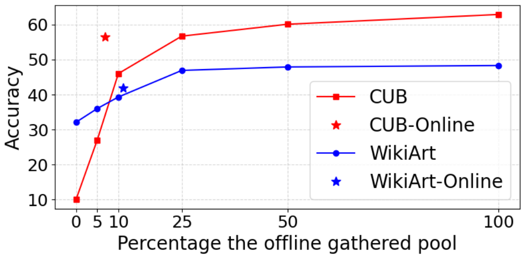


Figure 15: Comparison of different annotation scales in our framework.

A.13 Justification of the Uncertainty Measure

In this section, we justify our choice of uncertainty measure, i.e., average entropy of the generated sequence (see Sec. 4.4 and Eq. 3). First, we show the negative correlation between the uncertainty score and the prediction accuracy. Moreover, we compare different uncertainty estimation approaches.

A.13.1 Correlation between Uncertainty and Prediction Accuracy

We conduct a statistical analysis to quantify the correlation between these two terms. Specifically, we estimate the Point Biserial Correlation Kornbrot (2014), which is dedicated to measuring the correlation between a continuous variable, i.e., uncertainty score, and a binary variable, i.e., prediction accuracy. The correlation score ranges from -1 to 1, where a value closer to -1 means a stronger negative correlation between the two variables. We first perform zero-shot predictions on the test data and estimate the uncertainty as described in Sec. 4.4, and we repeat the experiments on CUB and WikiArt datasets with the LLaVA-OneVision model. We obtain -0.400 with a p-value $4.82 \times e^{-96}$ on CUB and -0.359 with a p-value of $9.73 \times e^{-38}$ on WikiArt, which indicates a statistically significant negative relationship between uncertainty and prediction accuracy. This suggests that the model tends to be more accurate when it is confident, and less accurate when it expresses higher uncertainty. This test justifies the effectiveness of this simple approach as an indicator of the prediction reliability.

A.13.2 Comparison of Different Entropy-Based Uncertainty

We further justify our choice by comparing different strategies for calculating the entropy. In our Eq. 3, we estimate the uncertainty as the average entropy across all indices. We denote our strategy as *All sequence* as it considers the entire sequence. However, there are other alternatives. For instance, one can consider the probability distribution of only the first token or the last token (i.e., the famous <EOS> token) instead of all sequence to calculate the entropy. We show the comparison in Tab. 9 with Point Biserial Correlation. We observe that the probability distribution of the <EOS> token is not always statistically significant. Instead, considering only the first token or the entire sequence leads to significant correlation between the uncertainty and prediction accuracy. These results confirm our design choice for the uncertainty estimation.

Table 9: Comparison of different strategies to estimate entropy-based uncertainty. Our chosen method is among the most significant ones.

Dataset	Method	Correlation	P-Value
CUB	EOS	-0.013	0.537
	First token	-0.409	$3.02 \times e^{-96}$
	All sequence	-0.400	$4.82 \times e^{-96}$
WikiArt	EOS	-0.120	$3.09 \times e^{-5}$
	First token	-0.358	$1.67 \times e^{-37}$
	All sequence	-0.359	$9.73 \times e^{-38}$

A.13.3 Comparison of Different Uncertainty Measures

We also considered other uncertainty measures in our experiments. According to recent studies Yang et al. (2023); Heo et al. (2024), off-the-shelf methods for uncertainty estimation can be categorized into 3 groups: **entropy-based**, **verbalized**, and **multiple-inference-based**. We first compare entropy-based and verbalized uncertainty, as both of them can be obtained with a single inference. This is an important advantage for our online ICD as it requires the student model to give an instant response in the online evaluation. In this regard, uncertainty estimation with multiple inferences that lead to latency and computational overhead is not ideal for our scenario. The comparison is in Tab. 10. The system prompt for the model for verbalized uncertainty is *Estimate your uncertainty with a float value between 0 and 1, with 0 as no uncertainty and 1 as the maximal uncertainty.* We found that, unlike entropy-based uncertainty, verbalized uncertainty fails to reflect the reliability of the prediction.

Table 10: Comparison of different strategies to estimate uncertainty. Our chosen method is statistically significant while the other one is not.

Dataset	Method	Correlation	P-Value
CUB	Verbalized	0.13	0.02
	Entropy	-0.400	$4.82 \times e^{-96}$
WikiArt	Verbalized	-0.065	0.259
	Entropy	-0.359	$9.73 \times e^{-38}$

Moreover, we compare with multi-inference-based uncertainty estimation. Specifically, we convert multi-inference uncertainty into a binary code, where consistent outputs are denoted as 0 for no uncertainty, and inconsistent outputs are denoted as 1. Similarly, we measure the Point Biserial Correlation between the entropy-based uncertainty and the uncertainty code from multiple inferences. We obtain 0.482 with a p-value of $1.99 \times e^{-30}$ on WikiArt and 0.426 with a p-value of $1.86 \times e^{-23}$ on CUB. This justifies that these two uncertainty measures are to some extent equivalent to each other. However, entropy-based uncertainty prevents the need for multiple inferences, which is thus a reasonable choice for our uncertainty estimation.

A.14 Limitations and Broader Impacts

Our ICD framework assumes the model possesses sufficient ICL capability. When this assumption does not hold, the model may not benefit from our framework. While our goal is to promote the use of small VLMs in practical applications—where smaller models offer better efficiency and accessibility—we are currently constrained by the capabilities of existing models. In particular, our framework does not extend to tiny VLMs with fewer than 4B parameters, which lack adequate ICL ability. We anticipate that ongoing progress in VLM research will yield more capable small models, enabling our approach to generalize to even smaller scales and more realistic deployment scenarios. Moreover, while we focus on a training-free paradigm where we rely on test-time computation to enhance the performance, the framework can be extended to incorporate in-context training for a model as a post-pretraining step to improve the performance at inference time.

Regarding the broader impact of our method, by avoiding full model fine-tuning and reducing reliance on large teacher models, our ICD lowers the energy footprint typically associated with adapting VLMs to new tasks. This makes it more accessible and sustainable for users with limited computational resources. Additionally, by facilitating the practical deployment of compact models, our approach may support broader adoption of multimodal AI in low-resource settings, such as education, environmental monitoring, or assistive technologies. At the same time, we acknowledge that our method builds on existing pretrained models and may inherit their limitations, including potential biases. Responsible use and continued evaluation in downstream applications remain important considerations.

A.15 Justification of our Selection Mechanism

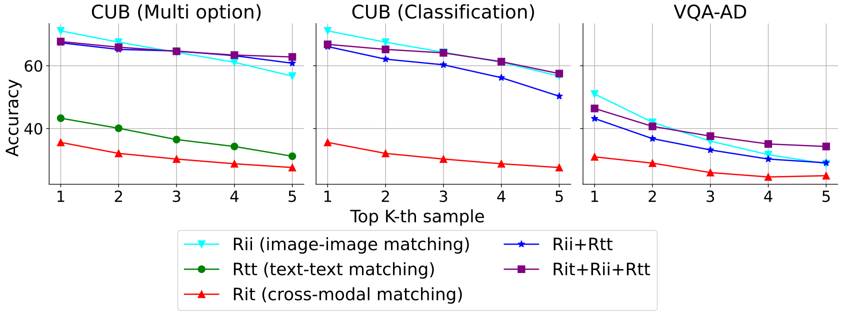


Figure 16: Comparison of the demonstration accuracy. Our cross-modal matching helps ensure a reasonable demonstration selection when the query text is generic. The text-text selection accuracy is omitted for clarity of the figure on CUB (Classification) and VQA-AD because of random selection.

We further justify our selection mechanism. We define the demonstration accuracy for a classification task as the proportion of demonstrations that belong to the same class as the query sample. We report this value as it is a direct measure of the quality of the demonstrations and the quality of the selection, and it is positively correlated with a better ICL performance. Moreover, to better illustrate the issue of generic query text, we utilize the CUB dataset with two question formats. In the standard classification task, the question is generic, i.e., *What species is this bird?*. In contrast, once converting the CUB dataset for multi-option VQA, the question contains additional information from the options, e.g., *What species is this bird? A. California Gull B. American Pipit*. In the former, since the question's embedding contains no information about the class, the text-text matching is ineffective. Nevertheless, in the latter, the options provide hints for the candidate classes and can be used for text-text matching. VQA-AD is included as it also contains a generic query question.

Therefore, we apply different strategies that we present in Sec. 4 to select demonstrations and report the demonstration accuracy in Fig. 16. Image-image matching R_{ii}, text-text matching R_{tt}, and cross-modal matching R_{it} are denoted as Rii, Rtt, Rit, respectively, in the figure. Note that we omit the text-text matching in the CUB (Classification) and VQA-AD setting to improve the visibility, as the matching with a generic query question is a random selection. First, we found that image-image matching is the most effective selection mechanism in most cases, especially for a task that is based on image classification. Moreover, questions with options enable effective text-text selection. Next, the cross-modal matching is not a strong selection mechanism as it relies on cross-modal matching capability of the multi-modal encoder. This explains why we only use it as a pre-selection mechanism to filter out potential noise. However, the cross-modal matching significantly enhances the selection when the text-text matching is not effective, as it helps eliminate irrelevant samples and reduces the noise. Lastly, we observe that the pre-selection of our cross-modal matching enhances the selection accuracy of the fourth or fifth demonstration, as it is useful in eliminating the noise from the candidate samples. We further show the comparison on VQA-AD, which contains more complex scenes in each image than classification tasks, leading to a degraded reliability on image-image only selection. In this case, the improvement of our proposed cross-modal matching is more noticeable. This justifies our design choice and highlights the importance of having an effective selection mechanism.

A.16 LoRA fine-tuning details

We compare our ICD with LoRA-based fine-tuning. In principle, parameter-efficient tuning methods are significantly less compute-intense than full-parameter tuning. However, we still observe an advantage of our ICD over this method, showing its best performance with minimal compute budget.

For LoRA fine-tuning, we follow the implementation of the official codebase of InternVL2.5. Specifically, we use a rank of $r = 16$. The LoRA adapter is inserted to the language model of the VLMs. In our resource constraint setting, to avoid a substaintial computational overhead, we apply an online learning paradigm with LoRA. Specifically, we wait until the teacher model annotate 64 samples to form a mini-batch. With this mini-batch, we conduct one gradient descent. After each step, we evaluate the test data with the updated LoRA adapter and the frozen backbone to calculate the accumulated accuracy. The associated computational cost consists of the training cost and the inference cost.