

LLaVA-SCo: Teach Vision Language Models to Self-Correct

Zixuan Liu
 Tulane University
 zliu41@tulane.edu

Guangkai Jiang
 guangkaijiang@gmail.com

Siavash Khajavi
 Aalto University
 siavash.khajavi@aalto.fi

Abstract

Large Language Models (LLMs) have demonstrated remarkable progress in self-correction, as seen in models like DeepSeek-R1. However, the current Vision-Language Models (VLMs) often struggle with self-correction in complex question-answering tasks. Furthermore, reinforcement learning (RL)-based approaches, while effective, incur substantial computational costs. In this work, we propose LLaVA-SCo, a novel VLM designed for efficient self-correction without relying on RL. Instead, we introduce a self-correction stage following the sequential reasoning steps of LLaVA-CoT, refining model outputs through a two-turn self-correction mechanism trained with supervised fine-tuning. To support this, we construct a large-scale dataset enriched with refined reasoning annotations to enhance correction capabilities. Experiments on multimodal reasoning benchmarks demonstrate that LLaVA-SCo outperforms its base model by 2.7%, exhibiting improvements in reasoning capabilities. Additionally, evaluations using GPT-4o indicate that responses after self-correction exhibit clearer structure, improved comprehension, and are preferred by GPT-4o. Furthermore, self-correction performance metrics validate that LLaVA-SCo effectively refines its reasoning, achieving consistent accuracy gains while minimizing reversal errors, confirming its ability to systematically improve responses through self-correction. The code is available at <https://github.com/ZixuanLiu4869/LLaVA-SCo>.

1. Introduction

Recent advancements in Large Language Models (LLMs), exemplified by DeepSeek-R1 [6], have demonstrated remarkable performance in reasoning tasks across domains such as mathematics, coding, and logical inference [17, 26, 27, 32]. A particularly intriguing capability of these models is their ability to exhibit an "aha moment"—a form of self-reflection where the model revisits and reassesses its previous steps while exploring alternative problem-solving approaches [6]. These emergent behaviors are not ex-

plicitly programmed but arise from reinforcement learning techniques, such as Group Relative Policy Optimization (GRPO) [26]. While vision is a crucial modality for enhancing AI's cognitive abilities and real-world understanding, current Vision-Language Models (VLMs) lack a comparable mechanism for self-correction [2]. Enabling VLMs to replicate this "aha moment" is essential for improving their reasoning capabilities and reliability in complex multimodal tasks.

One approach to aligning VLMs with this self-reflective behavior is through reinforcement learning (RL)-based methods, such as GRPO, which have shown promising results [26]. However, RL pipelines are inherently more complex than supervised learning, requiring multiple model instances, iterative sampling from VLM policies, and high computational costs [3, 24]. Recent research suggests that structured, stage-wise reasoning via Supervised Fine-Tuning (SFT) can offer an efficient alternative, helping VLMs segment their reasoning process into clear, distinct phases [29]. For example, the LLaVA-CoT [29] model explicitly divides reasoning into four stages—Summary, Caption, Reasoning, and Conclusion—each serving a distinct role in the reasoning workflow.

Inspired by this structured reasoning paradigm, we introduce LLaVA-Self-Correction (LLaVA-SCo), a novel Self-Correction Vision-Language Model that enhances reasoning accuracy without the need for computationally expensive RL techniques. Our key innovation is the explicit introduction of a self-correction stage after the sequential reasoning stages of LLaVA-CoT. Instead of relying solely on an initial reasoning pass, our approach encourages the model to engage in self-evolution, systematically reevaluating its prior response, identifying errors, and refining its reasoning strategy.

In this self-correction stage, we explicitly teach the model to introspect and revise its outputs using a dedicated tag (e.g., `<SELF-CORRECTION>...</SELF-CORRECTION>`). This mechanism guides the model to dynamically refine its responses, mitigating the common issue in autoregressive models where an early mistake cascades into a flawed final answer [29]. Previous

research has shown that strong LLMs can often correct incomplete or incorrect mathematical proofs when provided with partial solutions [26, 30], suggesting that leveraging previous responses should, in principle, enable VLMs to refine their reasoning as well.

To enable this self-correction capability, we train LLaVA-SCo on two-turn supervised fine-tuning. In the first turn, the model generates its initial answer, serving as a baseline attempt. In the second turn, the model revisits and refines its response, guided by an explicit self-correction instruction tag. To construct a robust training dataset, we obtain self-corrected responses from GPT-4o [20], prioritizing instances where the second response successfully improves upon the first attempt and where the model demonstrates explicit reasoning corrections rather than superficial rewording. By training on these self-correction traces, we teach the model not only to reason in structured stages but also to dynamically refine its own outputs in a self-improving feedback loop.

We evaluate LLaVA-SCo on several multimodal reasoning benchmarks and observe that the self-correction stage significantly enhances the model’s ability to solve complex reasoning tasks compared to the base model. Furthermore, evaluations using GPT-4o indicate that self-corrected responses exhibit clearer logical structures and are preferred over the answers from the original LLaVA-CoT model. Moreover, the evaluation of self-correction performance demonstrates that our model effectively performs self-correction, achieving consistent accuracy improvements across benchmarks.

Our main contribution is LLaVA-SCo, a Vision-Language Model designed to perform self-correction through supervised fine-tuning on two-turn self-correction traces. Experimental results demonstrate improved accuracy compared to baseline methods, highlighting the effectiveness of our approach in refining reasoning capabilities and response quality. By integrating structured reasoning with self-correction, we take a step towards more reliable and interpretable multimodal AI systems.

2. Related Work

Vision-Language Models. With the rapid evolution of Large Language Models (LLMs), there has been growing interest in incorporating visual information into these models to enhance their ability to interact with and understand multimodal content [5, 14, 33]. Recent progress in Vision-Language Models (VLMs) has led to powerful models such as GPT-4o [20], LLaVA-1.5 [15], and LLaVA-CoT [29], which have demonstrated remarkable performance across various multimodal tasks, including visual question answering and image classification. Current VLMs typically harness the advanced reasoning capabilities of LLMs to interpret visual information. These models leverage vast

image-text datasets to establish connections between the vision encoder and the LLM, allowing LLMs to effectively process and understand visual content [14, 29, 33]. Further advancements in this field have focused on enhancing multimodal datasets and developing innovative architectural frameworks to strengthen the integration between vision and language representations [4, 15, 28]. These advancements contribute to enhancing the model’s ability to seamlessly integrate and reason across both text and visual data, pushing the boundaries of VLM capabilities.

Intrinsic Self-Correction. Self-correction in LLMs and VLMs focuses on enabling models to rectify errors in their generated content, such as factual inaccuracies or illogical reasoning [7, 22]. While strong models can improve their outputs through external feedback [21], achieving effective self-correction without external guidance—a process known as intrinsic self-correction or self-correction through in-context learning—remains significantly more challenging [9, 10]. Existing efforts to enhance intrinsic self-correction largely fall into two categories: prompting-based approaches and fine-tuning-based methods.

Prompting techniques attempt to guide models to self-correct using natural language instructions. For instance, [12] demonstrated that LLMs could perform self-correction through simple instruction-based prompting, while [19] introduced an iterative feedback loop, where an LLM evaluates its own outputs and refines them accordingly. These approaches have the advantage of not requiring additional supervised training data, fine-tuning, or external supervision. However, recent studies indicate that naive prompting alone is often insufficient for robust self-correction and may even degrade performance in certain cases [9, 23, 31]. Consequently, prompting alone has not yet led to major breakthroughs in intrinsic self-correction.

In contrast, fine-tuning approaches enhance a model’s ability to self-correct by training on explicit revision demonstrations, either collected from human annotators [25] or stronger models [23]. Various fine-tuning strategies have been explored for both LLMs and VLMs. For example, [13] proposed an iterative RL-based approach, where LLMs are fine-tuned using a reward function that evaluates the quality of their outputs across multiple rounds of self-correction. This method enables self-correction without requiring larger external models or human supervision. Similarly, [7] applied Direct Preference Optimization (DPO) [24] to enhance VLMs’ ability to self-improve. They constructed a large-scale preference dataset, incorporating both successful and failed self-correction attempts, and fine-tuned the VLMs based on these preference signals. While DPO-based self-correction effectively aligns model behavior with preferred responses, it lacks an explicit self-correction stage. Instead of instructing the model to directly identify and fix its mistakes, DPO optimizes responses

based on preference ranking, making corrections implicit rather than structured. This can lead to less interpretable reasoning improvements and a reliance on preference-based adjustments rather than an explicit self-revision process. In contrast, our method directly trains the model to recognize its errors and revise its reasoning step-by-step, mimicking the cognitive process of human self-correction. Our approach ensures that VLMs learn a more transparent and interpretable correction strategy, rather than solely relying on implicit preference adjustments.

3. Method

Our approach, LLaVA-SCo, introduces a two-turn supervised fine-tuning framework that incorporates a dedicated self-correction stage, enhancing the reasoning capabilities of Vision-Language Models (VLMs). Unlike conventional training methods that solely optimize for high-reward responses to given prompts, our method encourages models to develop an intrinsic self-correction mechanism that remains effective at inference time. Figure 1 illustrates our framework compared to LLaVA-CoT.

3.1. Problem Setup

Our objective is to develop a simple yet effective training approach that enables VLMs to refine their own predictions using only self-generated data. We operate within the intrinsic self-correction framework [9, 10], where models attempt to rectify their initial responses without relying on external feedback.

Formally, given a dataset $D = \{(x_i, v_i, y_i^1)\}_{i=1}^N$ consisting of problem descriptions x_i paired with corresponding images v_i , an initial response y_i^1 generated by a VLM policy π_θ . Our goal is to train the model to improve its response, ensuring a more accurate and coherent solution to the problem x . This formulation is similar to prior work [13], but crucially, we do not assume access to an external oracle for correctness verification during training. Instead, our model learns from revision demonstrations in a manner similar to [23, 25]. Furthermore, at test time, the model must independently identify potential errors in its responses and correct them, making this approach particularly suitable for complex reasoning tasks, such as mathematical problem-solving. Unlike LLaVA-CoT, we do not rely on stage-level beam search at inference time for most of our experiments, though it remains an optional enhancement for future work.

3.2. Structured Self-Correction for Enhanced Reasoning

LLaVA-CoT [29] demonstrated that breaking down reasoning into structured stages can improve VLMs' ability to perform systematic reasoning. Inspired by this approach, we introduce an explicit self-correction stage within the reasoning process, moving beyond simple prompting meth-

ods [13]. Instead of treating self-correction as a loosely guided refinement, our method explicitly integrates it into the model's structured reasoning process, ensuring a more rigorous and interpretable correction mechanism. The generation process is structured into the following stages:

- **First Turn.** In this initial phase, the VLM generates a response using a structured reasoning process similar to LLaVA-CoT [29], employing predefined special tags: `<SUMMARY>`, `</SUMMARY>`, `<CAPTION>`, `</CAPTION>`, `<REASONING>`, `</REASONING>`, `<CONCLUSION>`, `</CONCLUSION>`. This step forms the foundation for subsequent self-correction.
- **Self-Correction Stage.** Building upon the initial attempt, the model reevaluates its prior response, systematically checking for potential errors and refining its answer where necessary. Unlike direct prompting methods, which merely instruct the model to generate a revised response without an explicit intermediate step, our approach **integrates self-correction into the structured reasoning pipeline**. We introduce the `<SELF-CORRECTION></SELF-CORRECTION>` tags to clearly define this stage, ensuring the correction process is an explicitly learned integral part of the reasoning rather than an externally applied instruction. This allows the model to develop an internal mechanism for detecting and addressing errors, rather than relying on user intervention or additional prompts.
- **Second Turn.** Following the self-correction stage, the model synthesizes a refined response. The final output is derived from the conclusion stage of the final refined response, while prior reasoning steps remain as internal states guiding the response generation. To further enhance accuracy, a weighted voting mechanism can be employed, where the second attempt is given greater importance in determining the final response.

A key distinction of our approach is that self-correction is not treated as an ad-hoc refinement but rather as a structured step within the model's reasoning process. Traditional prompting-based self-correction methods merely instruct the model to revise its response, but this revision occurs in an uncontrolled manner, often resulting in minimal or surface-level improvements. In contrast, our method explicitly teaches the model to recognize and rectify its mistakes by incorporating a dedicated correction phase into the reasoning workflow. Furthermore, since VLMs generate responses token-by-token, errors in the initial attempt can propagate and influence the overall reasoning trajectory [13]. Our self-correction stage interrupts this erroneous trajectory, teaching the model to reconsider its approach before producing the final output. This structured refinement process leads to more reliable improvements and enhances the model's ability to generalize its correction strategies.

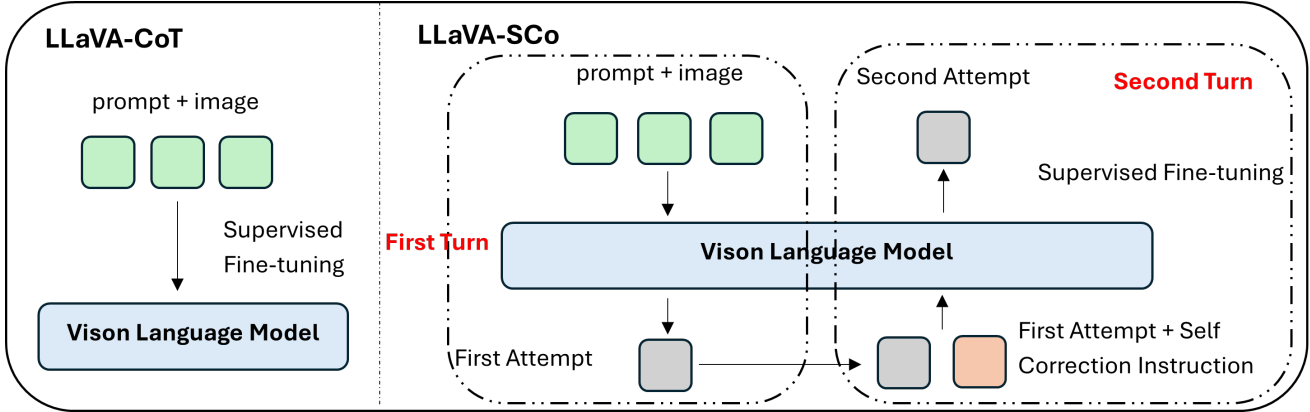


Figure 1. **LLaVA-SCo compared to LLaVA-CoT.** Our approach extends the LLaVA-CoT training framework by incorporating an additional turn for self-correction. In the second turn, the model outputs an explicit instruction tag prompting it to revise its initial response. This enhancement allows LLaVA-SCo not only to generate high-quality responses in the first attempt but also to refine and improve them through self-correction in the second turn.

3.3. Data Preparation and Model Training

To construct revision demonstration traces for training, we utilize the LLaVA-CoT-100k dataset [29], which contains nearly 100k visual question answer (VQA) pairs. We first generate the first-attempt responses by querying the LLaVA-CoT model, which has been trained on this dataset using supervised fine-tuning. Additionally, we incorporate the original answers provided in the LLaVA-CoT-100k dataset as another source of the first-attempt responses.

Our goal is to generate high-quality revision demonstrations that enable the model to refine its reasoning in subsequent attempts. Since no existing multimodal model can directly perform self-correction reasoning, we leverage GPT-4o [4] to generate revision demonstrations. Specifically, GPT-4o evaluates the first-attempt responses, identifies errors, and produces improved corrections. Notice that we want to train the model to improve from the first attempt to the second, while also ensuring it can generate high-quality initial responses that require no correction but exhibit enhanced reasoning in the second attempt. To do this, we categorize the collected data into two main cases: (1) instances where the first attempt contains errors, but the second attempt successfully corrects them, and (2) instances where both attempts generate correct responses, with the second attempt demonstrating further refinement and improved reasoning as evaluated by GPT-4o. This approach is designed to mitigate behavior collapse observed in previous work using supervised fine-tuning [13], ensuring that the model effectively learns the desired self-correction behavior.

Once the revision demonstrations are collected, we concatenate the two attempts with the explicitly defined self-correction stage in between, as described in the previous section. This forms a complete reasoning process where self-correction is explicitly integrated into the model’s

workflow. Additional details on the data generation process are provided in Appendix 7.

The compiled dataset can be directly used for Supervised Fine-Tuning (SFT) on any existing model to incorporate self-correction capabilities. In this work, we use LLaVA-CoT as the base model and apply LoRA parameter-efficient fine-tuning [8] using our constructed dataset. The training is conducted on a single node equipped with an NVIDIA A100 GPU. Further details on training parameters are provided in Appendix 8.

4. Experiments

In this section, we conduct a comprehensive evaluation of LLaVA-SCo by comparing it with the original LLaVA-CoT model across 4 commonly used multimodal benchmarks. Our goal is to measure the impact of our approach during training and assess its effectiveness in enhancing reasoning and self-correction capabilities. Through these experiments, we seek to answer the following key research questions:

1. Does LLaVA-SCo outperform the base model in terms of reasoning accuracy and overall performance?
2. Can our model effectively perform self-correction, leading to improved responses in the second attempt?

To address these questions, we conduct experiments on multiple benchmarks, comparing the accuracy of LLaVA-SCo and LLaVA-CoT. Moreover, we evaluate performance differences by calculating the win rate between the outputs of LLaVA-SCo and LLaVA-CoT. We also measure the self-correction performance between the first and second responses according to the metrics in [13]. These comparisons allow us to quantify both overall model improvements and the model’s ability to refine its responses through self-correction.

4.1. Experimental Setup

To assess the performance of LLaVA-SCo, we conducted experiments on four well-established and challenging multimodal benchmarks: MMStar [5], MMBench [16], MathVista [18], and AI2D [11]. MMStar and MMBench are designed to evaluate general visual question-answering capabilities, while MathVista and AI2D specifically assess a model’s ability to perform mathematical and scientific reasoning. For MathVista, we use the testmini subset. All models are evaluated using the official inference scripts provided in the Llama-Cookbook with hyperparameters as default. Performance is measured based on the accuracy of the generated responses compared to ground truth, as well as the pairwise win rate between models, with GPT-4o serving as the evaluator. Details regarding the evaluation prompts are provided in Appendix 7. To evaluate the self-correction performance of our model, we introduce the following metrics, inspired by [13]:

1. **Accuracy@Initial (Acc@T1)**: The model’s accuracy on its first response attempt.
2. **Accuracy@Revised (Acc@T2)**: The model’s accuracy after generating the second attempt following the self-correction process.
3. **Net Improvement ($\Delta_{T1 \rightarrow T2}$)**: The overall increase in accuracy between the first and second attempts.
4. **Successful Corrections ($\Upsilon_{T1 \rightarrow T2}^{F \rightarrow T}$)**: The proportion of cases where the model’s first response was incorrect but was successfully corrected in the second attempt.
5. **Reversal Errors ($\Upsilon_{T1 \rightarrow T2}^{T \rightarrow F}$)**: The proportion of cases where the model’s first response was correct but became incorrect after the self-correction step.

4.2. Results

Table 1 presents a comparative evaluation of LLaVA-SCo against LLaVA-CoT across four multimodal benchmarks. Our proposed model, LLaVA-SCo, consistently outperforms LLaVA-CoT in three of the four benchmarks, demonstrating its effectiveness in enhancing reasoning. LLaVA-SCo achieves a slight improvement over LLaVA-CoT on MMStar, with accuracy increases from 57.6 to 58.0, suggesting minor refinements in general visual understanding. The most significant performance gain is observed in MMBench, where **LLaVA-SCo outperforms LLaVA-CoT by 9.4 percentage points (84.4 vs. 75.0)**, highlighting the improved multimodal question-answering capabilities of our model. On MathVista, LLaVA-SCo surpasses LLaVA-CoT by 1.7 percentage points, indicating a positive impact on mathematical reasoning tasks. In contrast, LLaVA-SCo performs slightly lower than LLaVA-CoT on AI2D, suggesting that the self-correction stage might not significantly benefit scientific diagram-based reasoning or that may need additional fine-tuning. Overall, LLaVA-SCo achieves a higher average performance (69.23 vs. 66.52) across all bench-

marks, demonstrating its improved reasoning capabilities.

Benchmark	LLaVA-CoT	LLaVA-SCo (Ours)
MMStar	57.6	58.0
MMBench	75.0	84.4
MathVista	54.8	56.5
AI2D	78.7	78.0
Average	66.5	69.2

Table 1. Performance comparison on multimodal benchmarks.

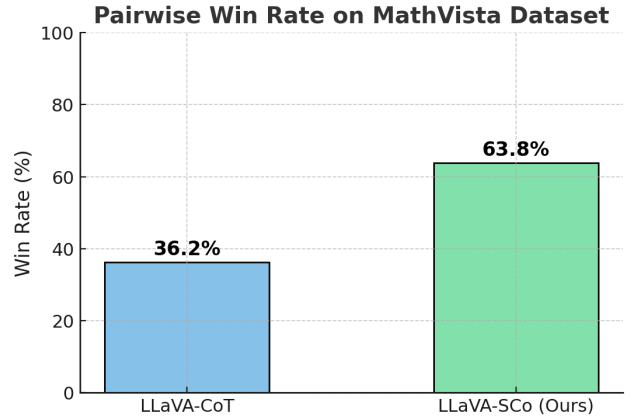


Figure 2. Pairwise win rate comparison between LLaVA-CoT and LLaVA-SCo on the MathVista dataset.

Figure 2 presents the pairwise win rate comparison between LLaVA-CoT and LLaVA-SCo on the MathVista dataset. Our model, LLaVA-SCo, achieves a **win rate of 63.8%**, while LLaVA-CoT attains 36.2%, demonstrating a significant improvement. This result highlights the effectiveness of our self-correction mechanism, which enables LLaVA-SCo to refine its reasoning and produce more accurate responses. The large margin in win rate suggests that our model consistently generates superior responses compared to the base model, particularly in complex multimodal reasoning tasks. These findings further validate our training methodology, reinforcing the importance of structured self-correction in enhancing vision-language models. In Table 2, we provide an example illustrating how LLaVA-SCo outperforms LLaVA-CoT in a real multimodal reasoning task, where LLaVA-CoT fails to provide the correct response while LLaVA-SCo successfully arrives at the correct answer. This comparison demonstrates how our model leverages self-correction to enhance reasoning accuracy. Additional examples of comparisons between the two models can be found in the Appendix 9, where we include cases where both models provide correct responses but LLaVA-SCo demonstrates superior reasoning, as well

as cases where both models are incorrect but LLaVA-SCo exhibits better logical consistency.

Table 3 presents the self-correction performance of LLaVA-SCo across four multimodal benchmarks. The table compares accuracy before and after self-correction (Acc@T1 vs. Acc@T2) and evaluates other key performance metrics. LLaVA-SCo consistently improves accuracy ($\Delta_{T1 \rightarrow T2}$) across all benchmarks, with the highest gain in MMStar (2.8%). MathVista and AI2D also show notable improvements, highlighting the effectiveness of self-correction in reasoning tasks. The successful correction rate ($\Upsilon_{T1 \rightarrow T2}^{F \rightarrow T}$) is highest in MMStar (3.9%), followed by MathVista (2.9%) and AI2D (2.5%), indicating strong correction capabilities. The reversal error rate ($\Upsilon_{T1 \rightarrow T2}^{T \rightarrow F}$) remains low across all benchmarks, with a maximum of 1.3% in MathVista, suggesting that LLaVA-SCo effectively refines responses without introducing significant new errors. These results validate the model’s ability to improve reasoning while maintaining reliability.

Figure 3 shows two example traces of self-correction performed by our LLaVA-SCo model. More examples can be found in Appendix 10. In the first example, during the first attempt, LLaVA-SCo miscalculates the remaining objects, incorrectly concluding that six objects are left. During the self-correction stage, the model reevaluates its reasoning process and identifies the green rubber objects that were mistakenly counted. After correcting this oversight, LLaVA-SCo produces the correct final answer of seven objects, aligning with the ground truth. In the second example, the model is tasked with determining whether Medium Periwinkle is the smoothest color in a scatter plot. Initially, LLaVA-SCo misinterprets the smoothness criteria, concluding that Medium Periwinkle appears smooth due to its consistent and evenly spaced data points. However, during self-correction, the model conducts a deeper analysis and identifies that, while Medium Periwinkle appears smooth, it is not the smoothest compared to other colors such as Violet and Rebecca Purple, which exhibit even fewer fluctuations. This leads the model to correctly conclude that Medium Periwinkle cannot be labeled as the smoothest, aligning with the ground truth. These examples highlight how LLaVA-SCo effectively leverages its structured self-correction mechanism to identify and rectify errors. The model demonstrates a deeper understanding of visual and numerical reasoning by refining its conclusions based on a more thorough analysis of the input data.

5. Discussion and Limitations

In this work, LLaVA-SCo performs the self-correction stage after completing the full reasoning process in the first attempt, which consists of the summary, caption, reasoning, and conclusion stages. A more comprehensive study of performing the self-correction stage after different reasoning

stages remains as future work. Moreover, as illustrated in Figure 3, we observe that the **summary and caption stages remain largely unchanged**, with most corrections occurring during the **reasoning stage**. This suggests that reevaluating the summary and caption may be redundant, and future work should explore refining only the reasoning stage to improve efficiency during inference.

Although the proposed two-turn supervised fine-tuning method improves accuracy across multiple benchmarks, its overall performance gains remain modest. The slight performance decrease on AI2D shows limitations of our self-correction techniques in certain fields. To further enhance self-correction capabilities, future research should consider incorporating reinforcement learning (RL). Our approach provides a foundation by equipping the model with intrinsic self-correction abilities, and subsequent RL-based fine-tuning could further optimize these capabilities—analogueous to Reinforcement Learning from Human Feedback (RLHF).

One potential direction is to optimize both reasoning attempts jointly using an RL objective, formulated as:

$$\max_{\theta} \mathbf{E}_{x,v,y_1,y_2 \sim \pi_{\theta}(\cdot|x,v)} \left[\sum_{i=1}^2 r(x,v,y_i) - \beta \mathbf{D}_{KL}(\pi_{\theta} || \pi_{ref}) \right] \quad (1)$$

where r serves as an oracle for evaluating response correctness, $\mathbf{D}_{KL}$ represents the KL divergence, and β is a hyperparameter controlling the balance between optimization and preventing excessive deviation from the original model.

6. Conclusion

In this paper, we introduce **LLaVA-SCo**, a novel vision-language model designed to perform self-correction. By explicitly incorporating a dedicated self-correction stage within the reasoning process, our model enhances its ability to identify and rectify errors, ultimately improving the overall reasoning quality. To achieve this, we construct a large dataset containing high-quality revision demonstration traces and train the model using a lightweight and efficient two-turn supervised learning approach.

Experimental results demonstrate that LLaVA-SCo effectively exhibits self-correction capabilities, leading to improved accuracy across multiple multimodal reasoning benchmarks. Additionally, evaluation results indicate that its responses are consistently preferred by GPT-4o.

Our work establishes a foundation for future research on **intrinsic self-correction in Vision-Language Models (VLMs)**. Potential directions for expansion include leveraging reinforcement learning techniques to further enhance self-correction capabilities, as discussed in Section 5. We believe that developing more robust self-correction mechanisms will contribute to the advancement of VLMs, fostering more reliable and interpretable multimodal AI systems.

Example 1: LLaVA-CoT Incorrect, LLaVA-SCo Correct

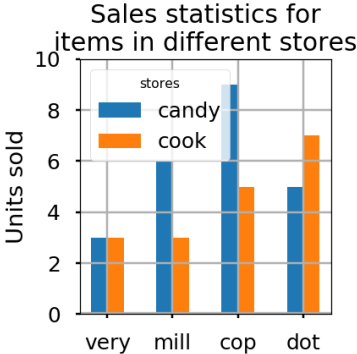
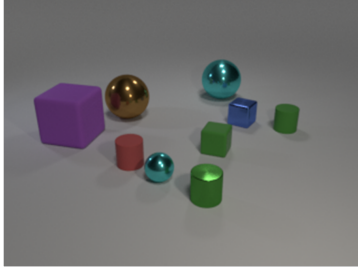
Prompt	How many items sold less than 5 units in at least one store?
Image	
Ground Truth	2
LLaVA-CoT Response	<REASONING>I will examine the sales figures for each item in the chart. For "candy," the sales are as follows: "very" (3 units), "mill" (6 units), "cop" (9 units), and "dot" (5 units). For "cook," the sales are: "very" (3 units), "mill" (3 units), "cop" (5 units), and "dot" (7 units). I will count the items that have sales less than 5 units in at least one store. "Candy" has sales less than 5 units in "very" (3 units). "Cook" has sales less than 5 units in "very" (3 units) and "mill" (3 units). Therefore, a total of three items have sales less than 5 units in at least one store. </REASONING><CONCLUSION>3</CONCLUSION>
LLaVA-SCo Response	<REASONING>I will analyze the bar chart for each item in the stores listed. For "candy," the sales are as follows: "very" (3 units), "mill" (6 units), "cop" (9 units), and "dot" (5 units). For "cook," the sales are: "very" (3 units), "mill" (3 units), "cop" (5 units), and "dot" (7 units). I will count the items that sold less than 5 units in at least one store. - "Candy" sold less than 5 units in "very" (3 units). - "Cook" sold less than 5 units in "very" (3 units), "mill" (3 units), and "cop" (5 units). Since "candy" sold less than 5 units in "very" and "cook" sold less than 5 units in "very," "mill," and "cop," both items have sales less than 5 units in at least one store. </REASONING><CONCLUSION>2</CONCLUSION>
GPT Evaluation	Response 2 is preferred because it provides the correct answer ("2") in alignment with the correct answer provided in the question. It systematically analyzes the bar chart and is clear in identifying that both items ("candy" and "cook") sold less than 5 units in at least one store, without making any calculation errors. In contrast, Response 1 incorrectly concludes "3," failing the correctness criterion despite a detailed explanation. Response 2 also has clear reasoning, making it more structured and understandable.

Table 2. Comparison of LLaVA-CoT and LLaVA-SCo evaluated by GPT-4o. The Summary and Caption stages are omitted, focusing on key reasoning differences.

Benchmark	Acc@T1	Acc@T2	$\Delta_{T1 \rightarrow T2}$	$\Upsilon_{T1 \rightarrow T2}^{F \rightarrow T}$	$\Upsilon_{T1 \rightarrow T2}^{T \rightarrow F}$
MMStar	55.2	58.0	2.8	3.9	1.1
MMBench	83.2	84.4	1.2	1.7	0.5
MathVista	54.9	56.5	1.6	2.9	1.3
AI2D	76.5	78.0	1.5	2.5	1.0

Table 3. Self-Correction Performance Metrics Across Benchmarks of LLaVa-SCo.



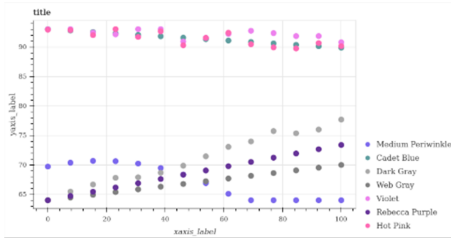
Question: Subtract all big matte balls. Subtract all green rubber objects. How many objects are left?

Ground Truth: 7.

LLaVA-SCo turn 1 response (**incorrect**): <SUMMARY> To solve the problem, I will identify and subtract the specified objects from the image. I will then count the remaining objects to provide the final answer. </SUMMARY>\n\n<CAPTION> The image contains various 3D objects, including spheres, cubes, and cylinders. The objects are of different colors and materials, such as matte and shiny surfaces. </CAPTION>\n\n<REASONING> First, I will identify all the big matte balls in the image. There is one large matte purple cube, but no big matte balls. Next, I will identify all green rubber objects. There are two green cylinders, which are rubber-like in appearance. I will subtract these two green rubber objects from the total count. After subtracting these objects, I will count the remaining objects. </REASONING>\n\n<CONCLUSION> 6 </CONCLUSION>

<SELF-CORRECTION> There might be an error in the response. I will reevaluate the response, correct the error, if any, and improve the solution. </SELF-CORRECTION>

LLaVA-SCo turn 2 response (**correct**): <SUMMARY> To solve the problem, I will identify and subtract the specified objects from the image. I will then count the remaining objects to provide the final answer. </SUMMARY>\n\n<CAPTION> The image contains various 3D objects, including spheres, cubes, and cylinders. The objects are of different colors and materials, such as matte and shiny surfaces. </CAPTION>\n\n<REASONING> First, I will identify all the big matte balls in the image. There is no big matte ball present in the image. Next, I will identify all green rubber objects. There are two green cylinders, which are rubber-like in appearance. I will subtract these two green rubber objects from the total count. After subtracting these objects, I will count the remaining objects. **The image contains a total of 9 objects: 1 large purple cube, 1 shiny gold sphere, 1 shiny blue sphere, 1 shiny teal sphere, 1 small red cylinder, 1 small teal cylinder, 1 small green cylinder, 1 small green cube, and 1 small blue cube. Subtracting the two green rubber objects leaves 7 objects.** </REASONING>\n\n<CONCLUSION> 7 </CONCLUSION>



Question: Is Medium Periwinkle the smoothest?

Ground Truth: No.

LLaVA-SCo turn 1 response (**incorrect**): <SUMMARY> To determine if Medium Periwinkle is the smoothest, I will analyze the graph's data points for Medium Periwinkle and assess their smoothness compared to other colors. </SUMMARY>\n\n<CAPTION> The image is a scatter plot with different colors representing various data sets. Each color corresponds to a specific label, such as Medium Periwinkle, Cadet Blue, and others. The x-axis ranges from 0 to 100, and the y-axis ranges from 65 to 95. The data points for Medium Periwinkle are shown in blue. </CAPTION>\n\n<REASONING> To evaluate the smoothness of Medium Periwinkle, I will examine the distribution of its data points. The smoothness can be assessed by looking at the consistency and lack of irregularities in the data points' arrangement. Medium Periwinkle's data points appear to be relatively consistent and evenly spaced, indicating a smooth trend. Comparing this to other colors, Medium Periwinkle seems to have fewer irregularities and a more uniform pattern, suggesting it is indeed the smoothest. </REASONING>

<SELF-CORRECTION> There might be an error in the response. I will reevaluate the response, correct the error, if any, and improve the solution. </SELF-CORRECTION>

LLaVA-SCo turn 2 response (**correct**): <SUMMARY> To determine if Medium Periwinkle is the smoothest, I will analyze the graph's data points for Medium Periwinkle and assess their smoothness compared to other colors. </SUMMARY>\n\n<CAPTION> The image is a scatter plot with different colors representing various data sets. Each color corresponds to a specific label, such as Medium Periwinkle, Cadet Blue, and others. The x-axis ranges from 0 to 100, and the y-axis ranges from 65 to 95. The data points for Medium Periwinkle are shown in blue. </CAPTION>\n\n<REASONING> To evaluate the smoothness of Medium Periwinkle, I will examine the distribution of its data points. The smoothness can be assessed by looking at the consistency and lack of irregularities in the data points' arrangement. Medium Periwinkle's data points appear to be relatively consistent and evenly spaced, indicating a smooth trend. **However, upon closer inspection, Medium Periwinkle's data points exhibit minor fluctuations and irregularities, particularly around the middle of the x-axis range. While it is smoother than some other colors, such as Violet or Rebecca Purple, it is not entirely smooth. Therefore, Medium Periwinkle cannot be definitively labeled as the smoothest.** </REASONING>\n\n<CONCLUSION> No </CONCLUSION>

Figure 3. The figure presents two example traces of self-correction performed by our LLaVA-SCo model. In both cases, LLaVA-SCo successfully identifies and corrects the errors from its first attempt. Notably, the corrections primarily occur in the REASONING stage, subsequently influencing the CONCLUSION stage, while the SUMMARY and CAPTION stages remain largely unchanged or undergo only minor modifications.

References

- [1] Meta AI. Llama cookbook. <https://github.com/meta-llama/llama-cookbook>, 2024. 1
- [2] Muhammad Awais, Muzammal Naseer, Salman Khan, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Foundation models defining a new era in vision: a survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 1
- [3] Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, J       Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*, 2023. 1
- [4] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. In *European Conference on Computer Vision*, pages 370–387. Springer, 2024. 2, 4
- [5] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*, 2024. 2, 5
- [6] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shitong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojuan Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yudian Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. 1
- [7] Jiayi He, Hehai Lin, Qingyun Wang, Yi Fung, and Heng Ji. Self-correction is more than refinement: A learning framework for visual and language reasoning tasks. *arXiv preprint arXiv:2410.04055*, 2024. 2
- [8] 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. 4
- [9] Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. *arXiv preprint arXiv:2310.01798*, 2023. 2, 3
- [10] Ryo Kamoi, Yusen Zhang, Nan Zhang, Jiawei Han, and Rui Zhang. When can llms actually correct their own mistakes? a critical survey of self-correction of llms. *Transactions of the Association for Computational Linguistics*, 12:1417–1440, 2024. 2, 3
- [11] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 235–251. Springer, 2016. 5
- [12] Geunwoo Kim, Pierre Baldi, and Stephen McAleer. Language models can solve computer tasks. *Advances in Neural Information Processing Systems*, 36:39648–39677, 2023. 2
- [13] Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, et al. Training language models to self-correct via reinforcement learning, 2024. URL <https://arxiv.org/abs/2409.12917>. 2, 3, 4, 5
- [14] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 2
- [15] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 2
- [16] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer, 2024. 5

- [17] Anton Lozhkov, Raymond Li, Loubna Ben Allal, Federico Cassano, Joel Lamy-Poirier, Nouamane Tazi, Ao Tang, Dmytro Pykhtar, Jiawei Liu, Yuxiang Wei, et al. Starcoder 2 and the stack v2: The next generation. *arXiv preprint arXiv:2402.19173*, 2024. [1](#)
- [18] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023. [5](#)
- [19] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594, 2023. [2](#)
- [20] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. [2](#)
- [21] Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies. *arXiv preprint arXiv:2308.03188*, 2023. [2](#)
- [22] Mansi Phute, Alec Helbling, Matthew Hull, ShengYun Peng, Sebastian Szyller, Cory Cornelius, and Duen Horng Chau. Llm self defense: By self examination, llms know they are being tricked. *arXiv preprint arXiv:2308.07308*, 2023. [2](#)
- [23] Yuxiao Qu, Tianjun Zhang, Naman Garg, and Aviral Kumar. Recursive introspection: Teaching language model agents how to self-improve. *Advances in Neural Information Processing Systems*, 37:55249–55285, 2025. [2](#), [3](#)
- [24] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023. [1](#), [2](#)
- [25] William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. Self-critiquing models for assisting human evaluators. *arXiv preprint arXiv:2206.05802*, 2022. [2](#), [3](#)
- [26] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. [1](#), [2](#)
- [27] Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024. [1](#)
- [28] Junke Wang, Lingchen Meng, Zejia Weng, Bo He, Zuxuan Wu, and Yu-Gang Jiang. To see is to believe: Prompting gpt-4v for better visual instruction tuning. *arXiv preprint arXiv:2311.07574*, 2023. [2](#)
- [29] Guowei Xu, Peng Jin, Li Hao, Yibing Song, Lichao Sun, and Li Yuan. Llava-o1: Let vision language models reason step-by-step. *arXiv preprint arXiv:2411.10440*, 2024. [1](#), [2](#), [3](#), [4](#)
- [30] Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. Do large language models latently perform multi-hop reasoning? *arXiv preprint arXiv:2402.16837*, 2024. [2](#)
- [31] Huaixiu Steven Zheng, Swaroop Mishra, Hugh Zhang, Xinyun Chen, Minmin Chen, Azade Nova, Le Hou, Heng-Tze Cheng, Quoc V Le, Ed H Chi, et al. Natural plan: Benchmarking llms on natural language planning. *arXiv preprint arXiv:2406.04520*, 2024. [2](#)
- [32] Tianyang Zhong, Zhengliang Liu, Yi Pan, Yutong Zhang, Yifan Zhou, Shizhe Liang, Zihao Wu, Yanjun Lyu, Peng Shu, Xiaowei Yu, et al. Evaluation of openai o1: Opportunities and challenges of agi. *arXiv preprint arXiv:2409.18486*, 2024. [1](#)
- [33] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. [2](#)