Synthetic Stimuli, Real Gains: Rethinking VLM Fine-Tuning Through Fully Controlled Data Generation

Anonymous ACL submission

Abstract

Performance gains of Vision Language Models (VLMs) obtained by fine-tuning are generally based on ad hoc data collection and annotation of real-world scenes. Despite the improvements, this process is often prone to biases, errors, and distribution imbalance, resulting in overfitting and imbalanced performance. Although a few studies have explored synthetic data generation, they typically lack control over data distribution and annotation quality. In this work, we re-evaluate the potential of model fine-tuning by exploring a fully controlled data generation and annotation pipeline, obtaining bias-free data with balanced distribution and clean annotations. Using the spatial reasoning task of identifying the absolute position of an object as a use case, we fine-tune state-of-the-art VLMs and conduct exhaustive evaluations on both synthetic and real-world benchmarks, including transferability to real-world scenes. Our experiments reveal two key findings: 1) fine-tuning on balanced data yields uniform performance across the visual scene and mitigates common biases with as few as 130 samples; and 2) fine-tuning on synthetic stimuli improves performance by 13% on real-world data (COCO), outperforming models fine-tuned on the full COCO train set.[1]

1 Introduction

Vision-Language Models (VLMs) have demonstrated competitive performance across a variety of downstream reasoning tasks, including visual question answering (Goyal et al., 2017; Chen et al., 2024a; Deitke et al., 2025), spatial reasoning (Krishna et al., 2017; Thrush et al., 2022; Yuksekgonul et al., 2023), counting (Acharya et al., 2019; Paiss et al., 2023), and visual scenes understanding (Fu et al., 2023; Cheng et al., 2024). To improve performance on these tasks, the prevailing approach is to collect task-specific annotated datasets from real-world scenarios, fine-tune the model on these data, and evaluate it on benchmarks built from similar distributions (Fu et al., 2025b; Yue et al., 2024). This pipeline has become the de facto paradigm for adapting and assessing VLMs in downstream tasks. However, despite satisfactory benchmark performance, VLMs still exhibit severe limitations in understanding the structure and semantics of visual scenes (Kamath et al., 2023; Rudman et al., 2025; Rizzoli et al., 2025). Therefore, the improvement does not necessarily reflect enhanced generalization, as it may be driven by random or spurious correlations (Roth et al., 2023; Esfandiarpoor et al., 2024).

A close inspection of the data used to improve (fine-tune) and evaluate (benchmark) VLMs' performance reveals annotation errors, distribution imbalance, and strong scene biases (Acharya et al., 2019; Kirillov et al., 2023; Schuhmann et al., 2021). As a result, both fine-tuning and evaluation reinforce each other's limitations, giving the illusion of improvement while masking fundamental weaknesses in visual reasoning. Models fine-tuned on collected data often learn to associate task success with spurious cues, such as object co-occurrence or central positioning rather than generalization; meanwhile, as the benchmarks are constructed from the same biased distributions, evaluation rewards the models for reproducing dataset-specific shortcuts instead of robust understanding (Chen et al., 2024b; Rahmanzadehgervi et al., 2024).

The current limitations in VLM understanding may result in catastrophic errors, especially in real-world deployment, where conditions differ from training. For instance, a model might learn to detect pedestrians only when they appear near the image center, and fail when they occur elsewhere. This highlights the need for a training and evaluation process that promotes task competence regardless of variability in irrelevant aspects, such as object

[1] We release all materials: link removed for double-blind review process

color, shape, or position.

Recent studies have attempted to move beyond performance metrics, probing VLMs' ability to reason about visual properties and relations (Peng et al., 2024; Rudman et al., 2025; Chen et al., 2025). These efforts highlight that benchmark results often conceal poor structural understanding and sensitivity to confounders. However, these studies remain limited by partial coverage and remaining biases in their data, preventing a systematic analysis of how VLMs acquire and generalize spatial knowledge. This highlights the need for systematic, controlled, and exhaustive datasets that enable the isolation of reasoning from spurious correlations. In this work, we study the role of fully controlled data generation and annotation for fine-tuning VLMs, using the spatial reasoning task of identifying the absolute position of an object (Peng et al., 2024; Rizzoli et al., 2025) as a use case. We frame our study around two central research questions.

RQ1 (Assessment): Can controlled synthetic fine-tuning improve VLMs? Current training pipelines often expose models to dataset biases, annotation errors, and distribution imbalance. We construct an exhaustive and balanced dataset to isolate performance from spurious cues and identify models' limitations. For this purpose, we comprehensively synthesize object attributes such as color, shape, size, and position. Focusing on the absolute position task as a use case, we fine-tune state-of-the-art VLMs and evaluate their ability to generalize across object configurations, measuring whether controlled training conditions enhance their spatial reasoning capabilities.

RQ2 (Transfer): Do improvements learned from controlled synthetic data transfer to real-world scenes? While synthetic data enables controlled, exhaustive, and error-free coverage, models are required to perform reliably on real-world images. To assess transferability, we evaluate VLMs fine-tuned on the synthetic dataset in an unmatched setting. We construct a real-world dataset for the same downstream task, starting from COCO (Lin et al., 2014). We further assess whether fine-tuning on synthetic data provides benefits over fine-tuning directly on real-world data by comparing transfer performance in the unmatched setting with a matched setting where models are fine-tuned and evaluated on real-world data. This setup allows us to assess whether the acquired spatial reasoning skills of identifying the absolute position of an object extend beyond synthetic stimuli and enhance reliability in real-world scenarios.

Together, these research questions guide our investigation into how controlled synthetic data can enhance fine-tuning outcomes and transferability of VLMs. Our experiments show that fine-tuning on controlled synthetic data improves model performance and transfers effectively to real-world settings. Notably, improvements are most pronounced where models were previously biased. Interestingly, fine-tuning on the entire COCO training set degrades performance, suggesting that more data is not always beneficial. Moreover, while fine-tuning on a balanced subset of COCO training data (matched setting) also improves overall performance, it introduces biases such as failing to learn specific positions (e.g., center), and does not consistently achieve the robustness of our synthetic approach.

2 Related Work

Scene Understanding Improving the performance of VLMs via fine-tuning on task-specific data has been applied across diverse domains, including mathematical reasoning (Zhang et al., 2025a; Shi et al., 2024; Gao et al., 2025; Zhang et al., 2025b), visual relationship understanding (Zhu et al., 2024), scene graph construction (Park et al., 2025), spatial reasoning (Ogezi and Shi, 2025; Ning et al., 2025), visual reasoning (Cheng et al., 2024), and shape recognition (Rudman et al., 2025). However, most studies inherit the issues of real-world data, while synthetic approaches often lack control over distribution and rely on annotations from generative models prone to hallucination.

Synthetic Data Generation Recent studies have resorted to synthetic data to cope with issues related to real-world data. Johnson et al. (2017) aimed at avoiding annotation errors via deterministic scene generation. SPEC (Peng et al., 2024) uses a diffusion-based approach to generate objects and background for the absolute position task. Nevertheless, their approach suffers from hallucinations and inconsistencies (Aithal et al., 2024; Kim et al., 2025). Similar issues are present in approaches that synthesize data for fine-tuning (Li and Li, 2025; Park et al., 2025) or for evaluation (Sbrolli et al., 2026) via generative models. Wang et al. (2025) generate synthetic scene and QA annotations, reducing labeling errors, but not addressing label imbalance. Other studies generate scenes consisting of geometric shapes (Rahmanzadehgervi

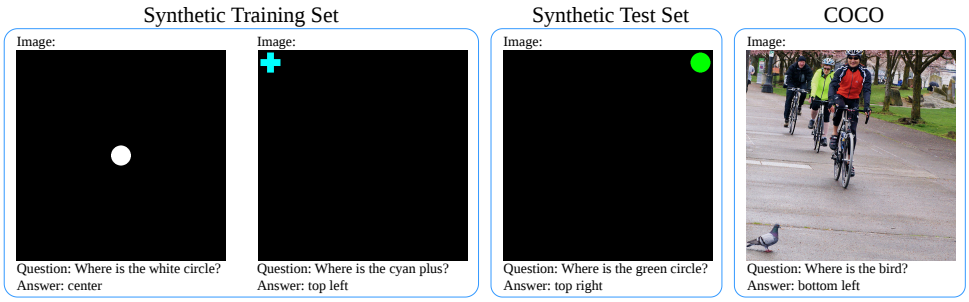


Figure 1: **Data samples.** Synthetic data consists of exhaustive sets of image-question pairs about a single object on a black background. Objects in the Synthetic Test Set are of color-shape combinations unseen in the Synthetic Training Set. The COCO Sets are obtained from the original COCO dataset by asking questions only for objects being the only instance of their category in a given image to avoid ambiguous questions.

et al., 2024; Rudman et al., 2025; Rizzoli et al., 2025), enabling systematic evaluation by isolating task-relevant factors and marginalizing irrelevant properties. In a related study, Kamath et al. (2023) proposed a dataset of real-world images obtained by physically constructing scenes with controlled perturbations, which, while interesting, imposes limited scalability due to setup cost and time.

3 Approach

We investigate how a fully controlled data generation pipeline can improve fine-tuning outcomes via the *Absolute Position* task, formulated as Visual Question Answering (VQA) over a 3×3 grid. To disentangle task performance from dataset artifacts, we construct controlled synthetic datasets with exhaustive and balanced coverage using CIVET (Rizzoli et al., 2025). We then assess transferability by evaluating the performance of the fine-tuned model on COCO (unmatched), and compare against fine-tuning on real-world data from the same distribution (matched).

3.1 Task: Absolute Position

This task requires identifying in which of nine equally sized regions of an image a target object is located. Each image is divided into a 3×3 grid representing nine possible locations: *top left*, *top center*, *top right*, *center left*, *center*, *center right*, *bottom left*, *bottom center*, and *bottom right*. For each image, we generate a closed-ended VQA sample asking for the location of a specific object, e.g., *"Where is the red square?"*. The nine grid locations are presented as answer options, and their order is randomized to prevent positional bias. This task setup follows recent work on spatial reasoning in VLMs (Peng et al., 2024; Rizzoli et al., 2025).

3.2 Dataset Construction

Using the CIVET framework (Rizzoli et al., 2025), we generate all synthetic data ensuring exhaustive coverage, balanced distributions, and the absence of annotation errors or sampling bias. We use synthetic data to isolate performance from confounding factors, while real-world data enable us to test transferability in an unmatched setting. Therefore, we complement these synthetic datasets with a version of the same task built from the COCO dataset (Lin et al., 2014). Data examples are illustrated in Figure 1, and additional details on the data construction are reported in §A.

Synthetic Test Set We first build an exhaustive synthetic evaluation dataset to measure spatial reasoning. Each image contains a single object on a uniform black background. We systematically vary four object attributes (i.e., *color*, *shape*, *size*, and *position*), generate an image for each combination, and obtain a balanced dataset of 3,888[2] VQA samples following the formulation in Section 3.1.

Synthetic Training Set To study whether controlled synthetic data can improve VLMs' spatial reasoning, we construct a training dataset with the same structure as the evaluation data but distinct color-shape combinations to avoid overlap (i.e., white <*shape*> or <*color*> plus). The resulting dataset comprises 1,620[3] image-question pairs, balanced across all positions. We keep 80% (1296) of the dataset for training and 20% (324) for validation. This configuration encourages the model to learn spatial reasoning independently of specific object shape or color cues, enabling an error-free analysis of controlled fine-tuning effects in both

[2]Computed as 6 colors $\times$ 4 shapes $\times$ 2 sizes $\times$ 81 fine-grained positions (i.e., 9×9 cells)

[3]Computed as (6 colored plusses $+$ 4 white shapes) $\times$ 2 sizes $\times$ 81 positions

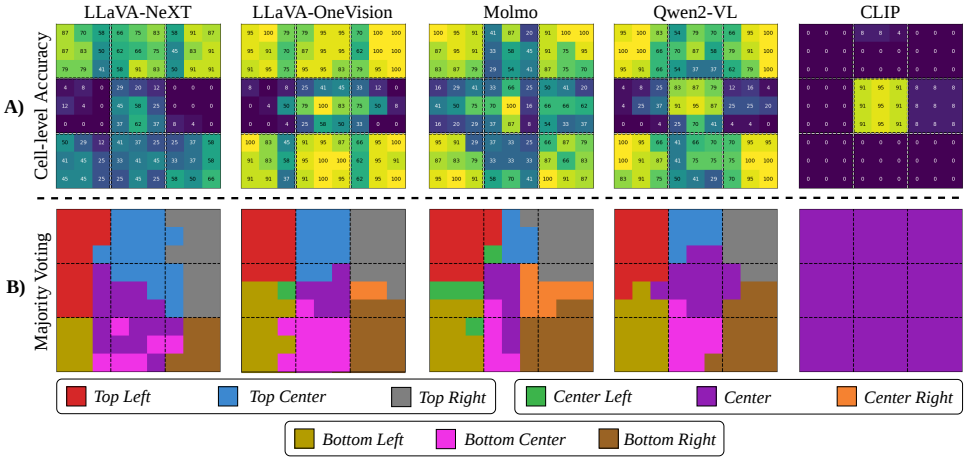


Figure 2: **A) Cell-Level Accuracy** and **B) Spatial Responses Map of VLM.** A) Accuracy on the *Absolute Position* task averaged over object variations across a fine-grained grid of 9×9 cells shows pronounced positional biases before fine-tuning. The models perform best in upper regions, with consistently low accuracy in center-left and center-right cells; CLIP exhibits an extreme central bias, failing elsewhere. B) Majority-vote answers reveal how models internally distort spatial structure.

synthetic and unmatched real-world settings.

Real-World Dataset To assess transferability to real-world data, we construct training and test datasets starting respectively from the train and validation[4] splits of COCO. For each image, we generate VQA samples that query the position of categories with a unique instance to avoid unambiguous questions (e.g., *"Where is the person?"*). We obtain a training set of 161,086 image-question pairs, a validation set of 40,272 pairs, and a test set of 8,548 pairs. While remaining consistent with the synthetic setup, the dataset captures real-world variability (e.g., multiple objects, diverse layouts, and non-square aspect ratios) and enables systematic analysis of how absolute position is learned in synthetic settings and transfers to real-world scenes.

3.3 Vision-Language Models

We evaluate five VLMs representative of the main dual-encoder and encoder-decoder architectures, allowing us to study balanced synthetic fine-tuning across design families. CLIP (Radford et al., 2021) is a dual-encoder model that learns aligned image and text representations through contrastive training. We include CLIP as a baseline and because its vision encoder serves as the foundation for several subsequent encoder-decoder VLMs. LLaVA-NeXT 7B (Liu et al., 2024) builds on CLIP by projecting visual features into the embedding space of a Large Language Model (LLM) through a learned projection layer. Molmo 7B (Deitke et al., 2025) follows a similar design but fine-tunes the entire architecture. The more recent Qwen2-VL 7B (Wang et al., 2024) and LLaVA-OneVision 8B (An et al., 2025) directly process images of varying resolutions without cropping or resizing.

[4]Test annotation is not provided.

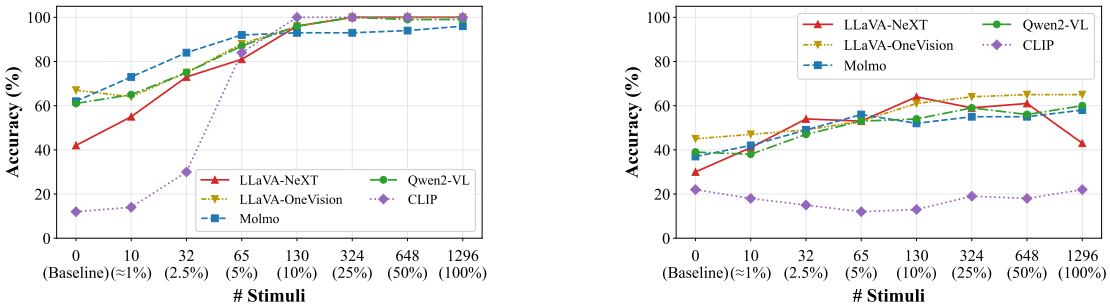
4 Experiments

RQ1: Can Controlled Synthetic Fine-Tuning Improve VLMs?

To investigate the effect of controlled synthetic fine-tuning, we begin by evaluating the spatial reasoning behavior of base models in the absolute position task. We then evaluate how fine-tuning on balanced synthetic data reshapes and improves these behaviors.

A. Cell-Level Accuracy To evaluate the spatial biases of base models before fine-tuning, we analyze both their fine-grained positional accuracy and spatial structure (Figure 2). For cell-level accuracy (Figure 2A), we subdivide the 3×3 region grid into a finer 9×9 cell layout and compute the mean accuracy over all object variations within each cell.

Results: The results (Figure 2) indicate that all models exhibit strong spatial biases prior to fine-tuning. LLaVA-OneVision , Molmo, and Qwen2-VL perform best in the upper and lower regions, while LLaVA-NeXT achieves high accuracy only in the upper regions, and CLIP achieves high accuracy only in the central region and fails elsewhere. A consistent weakness emerges in the center-left and center-right regions, and near region boundaries, where all VLMs struggle. While LLaVA-



(a) Synthetic Test Set

(b) COCO Test Set

Figure 3: **Effect of synthetic dataset scale.** Accuracy on the *Absolute Position* task as a function of the number of synthetic training stimuli. **(a)** Effect on the Synthetic Test Set. **(b)** Effect on the COCO Test Set.

OneVision, Molmo, and Qwen2-VL show better coverage, no model achieves uniform spatial performance.

B. Spatial Responses Map of VLMs We further analyze the models' answers regarding the absolute position through majority-vote (Figure 2B). For each cell, we aggregate answers across all object variations, and color-code the cells according to the most frequent answer position. This visualization exposes how models' answers "remap" the spatial grid before fine-tuning.

Results: Figure 2B shows that llavas over-represents the upper half, with many central cells misclassified as upper positions, and the bottom center collapsed into the center. While more symmetric, LLaVA-OneVision and Qwen2-VL exhibit strong vertical compression, with top and bottom regions substituting the central-left and central-right regions. Molmo produces a more coherent layout but compresses the central regions both horizontally and vertically. CLIP degenerates into answering only with center, confirming its extreme central bias observed in the cell-level accuracy. Together, these two analyses reveal that VLMs encode strong spatial bias towards top regions, and fail to represent intermediate regions. This highlights the necessity of fine-tuning on controlled synthetic data to eliminate such biases and foster accurate spatial representations.

C. Fine-Tuning We investigate whether fine-tuning on balanced synthetic data can enhance the spatial reasoning capabilities of VLMs in the absolute position task. Fine-tuning details are reported in §B. We fine-tune the models on the balanced synthetic dataset and compute the mean accuracy across five runs (detailed results and standard deviation are reported in §C.1).

Results: While the base models achieve at best 67% accuracy, fine-tuning consistently improves spatial reasoning across all models, achieving near-perfect accuracy ($\geq$ 96%) and minimal variance across runs. Overall, these findings validate our first research question (RQ1), i.e. fine-tuning on balanced synthetic data substantially enhances performance on the absolute position task while maintaining robustness across training runs.

D. Scaling Synthetic Data We evaluate how progressively increasing the size of the synthetic training set affects model performance.

Results: Across all models, accuracy increases rapidly with a small number of training stimuli (Figure 3a). Most models reach near-optimal accuracy with only 10% of the full set, after which performance plateaus, suggesting diminishing returns from additional data. Molmo exhibits the fastest gains, achieving strong performance even with limited data, while LLaVA-NeXT , LLaVA-OneVision, and Qwen2-VL improve more gradually but ultimately converge at a similar level. CLIP shows a different pattern, with minimal improvement at small scales followed by a sharp increase once sufficient samples are available, reflecting a greater dependence on data volume. Overall, the results demonstrate that fine-tuning on a small, balanced subset of synthetic data is sufficient to achieve robust performance, highlighting the sample efficiency of fine-tuning on controlled synthetic data.

RQ2: Do Improvements Learned from Controlled Synthetic Data Transfer to Real-World Scenes?

After observing improved performance by fine-tuning on controlled synthetic data, we investigate whether these improvements transfer to real-world data by evaluating models on COCO Absolute Posi-

Model	Training Set (#Samples)	Test Set Accuracy (%)	
		Synthetic	COCO
LLaVA-NeXT	Synthetic (1.3k)	100 (↑ 58)	43 (↑ 13)
	COCO *Complete* (161k)	0 (↓ 42)	0 (↓ 30)
LLaVA-OneVision	Synthetic (1.3k)	100 (↑ 33)	65 (↑ 20)
	COCO *Complete* (161k)	11 (↓ 56)	26 (↓ 19)
Molmo	Synthetic (1.3k)	96 (↑ 34)	58 (↑ 21)
	COCO *Complete* (161k)	4 (↓ 58)	6 (↓ 31)
Qwen2-VL	Synthetic (1.3k)	99 (↑ 38)	60 (↑ 21)
	COCO *Complete* (161k)	9 (↓ 52)	20 (↓ 19)
CLIP	Synthetic (1.3k)	100 (↑ 88)	22 (=)
	COCO *Complete* (161k)	11 (↓ 1)	36 (↑ 14)

Table 1: **Cross-domain transfer to real-world data**. Accuracy (%) on the *Absolute Position* task for models fine-tuned on balanced Synthetic (1.3k), and COCO Complete (161k) test sets. Results are averaged over 5 runs. Arrows (↑/↓) indicate absolute increase and decrease in accuracy with respect to the *Base Model*, while (=) denotes no change.

Model	Test Set Accuracy (%)	
	Synthetic	COCO
LLaVA-NeXT	71 (↑ 29)	67 (↑ 37)
LLaVA-OneVision	77 (↑ 10)	64 (↑ 19)
Molmo	80 (↑ 18)	45 (↑ 8)
Qwen2-VL	80 (↑ 19)	61 (↑ 22)
CLIP	13 (↑ 1)	36 (↑ 14)

Table 2: **Fine-Tuning on balanced real-world data**. Accuracy (%) on the *Absolute Position* task for models fine-tuned on a COCO Subset (1.3k), balanced in terms of object category and position. Results are averaged over 5 runs. Arrows (↑) indicate absolute increase in accuracy with respect to the *Base Model*.

tion. To probe transferability, we consider two complementary evaluation conditions: i) *unmatched setting*, where models are fine-tuned on synthetic data and evaluated on COCO; and ii) *a matched setting*, where models are fine-tuned and evaluated on COCO data. This comparison enables us to disentangle whether the benefits of exhaustive, bias-free synthetic training extend to uncontrolled real-world distributions.

A. Cross-Domain Transfer We evaluate the VLMs on the COCO Absolute Position test set to measure how effectively spatial reasoning learned from synthetic data transfers to real-world images. Each model is fine-tuned on the synthetic training dataset and subsequently tested both on the synthetic and COCO benchmarks. To compare the performance with the matched setting, we additionally fine-tune models on the complete COCO training set (∼161k samples).

Results Table 1 summarizes model accuracy across matched (synthetic) and unmatched (COCO) evaluation settings. Fine-tuning on the balanced synthetic dataset markedly improves performance on the absolute position task across all encoder-decoder models, not only on the matched synthetic test but also when transferring to real-world data. LLaVA-OneVision, Molmo, and Qwen2-VL each show gains of +20% points or more on COCO, achieving around 60% accuracy after synthetic fine-tuning. This indicates that models trained on controlled stimuli acquire transferable reasoning rather than overfitting to synthetic patterns. Nevertheless, CLIP fails to benefit from fine-tuning on synthetic data, suggesting a limitation of dual-encoder models. In contrast, models fine-tuned on the full COCO training set (Table 1) exhibit strong degradation, with some models' performance dropping to near-zero accuracy, despite keeping the training procedure and hyperparameters the same as the synthetic fine-tuning. LLaVA-NeXT generates empty strings as output, while Molmo keeps generating tokens from the set {*center, right, left*}, resulting in invalid answers. Instead, LLaVA-OneVision and Qwen2-VL generate answers that are mostly valid, but often incorrect, reaching 50% accuracy only in the center region, which is the most frequent in the data. This suggests that naive fine-tuning on large-scale real-world data can inject noise and bias that hinder the learning of consistent spatial structure.

B. Balanced Real-World Fine-Tuning To test whether data scale and imbalance hinder learning rather than the real-world setting, we construct a subset of COCO equivalent in size to our synthetic training set (i.e., 1,296 samples), balanced in object categories and positional distribution.

Results: Interestingly, fine-tuning models on the balanced COCO Subset improves results and outperforms fine-tuning on the full COCO training set (Table 2). Notably, fine-tuning models on our controlled Synthetic training set (unmatched) or the balanced COCO Subset (matched) yields similar results when tested on COCO, but with the COCO Subset leading to more unstable performance, with much higher gains for LLaVA-NeXT and considerably lower for Molmo. Overall, these results demonstrate that quality, balance, and control in training data outweigh sheer quantity.

C. Data Scale and Transfer Efficiency To understand how data quantity influences the performance to real-world settings, we progressively increase

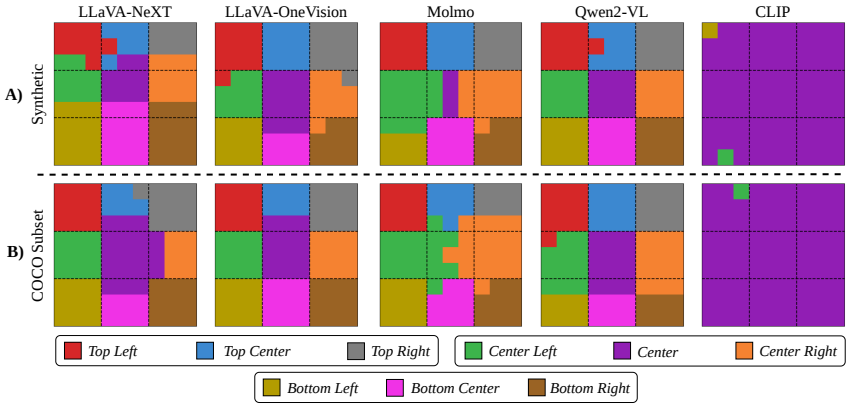


Figure 4: **Model answers on the COCO Absolute Position test set** after fine-tuning on different data sources (by majority voting); A) Models fine-tuned on synthetic data; and, B) Models fine-tuned on COCO Subset.

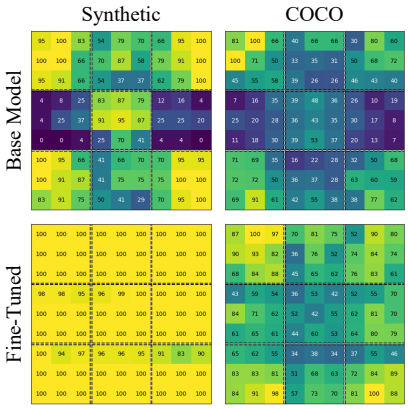


Figure 5: **Cell-level accuracy of Qwen2-VL 7B before and after fine-tuning** on synthetic data, on both the synthetic and COCO Absolute Position test sets.

the number of synthetic training samples and evaluate model accuracy on the COCO test set.

Results: Across VLM models, performance improves sharply even with a small fraction of the synthetic dataset, demonstrating the sample efficiency of balanced synthetic fine-tuning (Figure 3b). With 10% of the full synthetic data (130 samples), LLaVA-NeXT achieves its maximum transfer accuracy, and LLaVA-OneVision, Molmo, and Qwen2-VL obtain most of their transfer improvement, after which performance plateaus. In contrast to encoder-decoder VLMs, CLIP remains largely insensitive to training size, with accuracy fluctuating around 20%, suggesting that dual-encoder architectures do not effectively transfer from fine-tuning on synthetic data. Overall, these results highlight that balanced synthetic data achieves strong transfer with few samples, and that careful control and balance are far more beneficial than scale alone.

D. Cell-Level Accuracy To better understand how fine-tuning affects positional reasoning, we analyze cell-level accuracy and model answer patterns before and after fine-tuning.

Results: Overall, these analyses demonstrate that fine-tuning on controlled synthetic data not only enhances positional accuracy but also refines spatial answers into coherent layouts. Figure 5 illustrates cell-level accuracy of Qwen2-VL, evaluated on both the synthetic and COCO test sets (other models are presented in §C.3). Before fine-tuning, the model exhibits strong spatial biases, performing best in the upper and lower regions while struggling in the center-left and center-right. Fine-tuning on synthetic data improves performance as the accuracy becomes nearly uniform across all 9×9 cells, with the largest gains mostly where the base model performed worst. Crucially, these improvements transfer to COCO, indicating that the model improved spatial reasoning capability for the absolute position rather than memorizing synthetic patterns. Figure 4 further visualizes position answers on COCO after fine-tuning on different data sources. Models fine-tuned on synthetic data (Figure 4A) generally produce consistent and well-structured spatial partitions, whereas fine-tuning on the balanced COCO subset (Figure 4B) can yield noisier and less regular layouts. Notably, in Molmo the *center* region is effectively overwritten after COCO fine-tuning, indicating that real-world data may be more challenging to learn from, even when balanced.

5 Ablation and Representation Analyses

We further investigate the factors that influence the robustness and interpretability of VLMs after con-

trolled fine-tuning.

A. Scene Complexity & Distractors As real-world scenes in COCO contain, on average, seven objects, we augment our synthetic dataset with distractor objects (details in §A). This allows us to systematically evaluate how increasing training scene complexity affects spatial reasoning over absolute position and transfer to real-world data. We fine-tune each VLM on synthetic datasets containing one, three, or five distractors and evaluate them on the Synthetic and COCO Absolute Position test sets (additional results are reported in §C.4).

Results: Table 3 shows that moderate visual clutter improves transfer to COCO for encoder-decoder VLMs. LLaVA-NeXT and Molmo benefit the most from adding three distractors, gaining +12% and +3% points, respectively on COCO. However, excessive clutter (five distractors) leads to diminishing or negative returns, suggesting that overly complex synthetic scenes can hinder learning and transfer. Qwen2-VL exhibits stable performance up to three distractors but slight degradation beyond that, indicating a similar saturation effect. LLaVA-OneVision shows improvement with distractors, but with reduced gains with respect to the clean set. CLIP remains largely unaffected, consistent with the limited transferability we observed. Overall, these findings indicate that introducing moderate scene complexity during fine-tuning enhances robustness and transfer to real-world data.

B. Layer-wise Representation Analysis To better understand how fine-tuning reshapes the internal representations of VLMs, we perform a layer-wise performance analysis (Fu et al., 2025a; Alghisi et al., 2025) before and after fine-tuning on our synthetic training set. For each layer of the LLM component, we extract the hidden representation corresponding to the final question token and train a linear SVM probe (3-fold cross-validation) to predict the absolute position label. This analysis allows us to localize where spatial reasoning emerges in the model hierarchy and how fine-tuning alters the encoding of spatial information (plots for each model are reported in §C.5).

Results: On synthetic data, accuracy rapidly increases for all models in early layers (5∼10) and saturates in the upper-middle layers. In contrast, on COCO the same trend appears with a slower rise (saturation after layer 15), reflecting the increased visual and linguistic complexity of real-world scenes. Together, these results indicate that fine-tuning on controlled synthetic data strengthens the internal representation of VLMs and that the learned representation for absolute position largely transfers to real-world settings, albeit with reduced confidence and stability.

Model	Training Set	Test Set Accuracy (%)	
		Synthetic	COCO
LLaVA-NeXT	Synthetsubsetic (1.3k)	100 (↑ 58)	42 (↑ 12)
	+3 Distractors	100 (↑ 58)	54 (↑ 24)
	+5 Distractors	81 (↑ 39)	48 (↑ 18)
LLaVA-OneVision	Synthetic (1.3k)	100 (↑ 33)	65 (↑ 20)
	+3 Distractors	100 (↑ 33)	60 (↑ 15)
	+5 Distractors	100 (↑ 33)	60 (↑ 15)
Molmo	Synthetic (1.3k)	96 (↑ 34)	57 (↑ 18)
	+3 Distractors	95 (↑ 33)	60 (↑ 21)
	+5 Distractors	97 (↑ 35)	65 (↑ 26)
Qwen2-VL	Synthetic (1.3k)	99 (↑ 38)	58 (↑ 20)
	+3 Distractors	93 (↑ 32)	58 (↑ 20)
	+5 Distractors	90 (↑ 29)	54 (↑ 16)
CLIP	Synthetic (1.3k)	100 (↑ 88)	22 (=)
	+3 Distractors	11 (↓ 1)	22 (=)
	+5 Distractors	11 (↓ 1)	28 (↑ 6)

Table 3: **Effect of fine-tuning with distractors** on the *Absolute Position* task. Results show the average accuracy (%) across five runs. Arrows (↑/↓) indicate absolute increase and decrease in accuracy with respect to the *Base Model*, while (=) denotes no change.

6 Conclusion

We studied a controlled approach to fine-tune Vision-Language Models, showing that controlled data can improve the fine-tuning outcomes and transfer to real-world scenes on the spatial reasoning task of absolute position. By systematically varying visual attributes and scene complexity, we isolated how models acquire and generalize spatial knowledge, revealing that the quality and balance of data matter more than scale. Our analyses further demonstrated that controlled fine-tuning reshapes model representations in interpretable ways and promotes robustness across models and complex scenes.

Beyond the specific task of spatial reasoning, our findings suggest that synthetic data, when exhaustively designed and bias-free, can serve as a reliable tool for diagnosing, training, and benchmarking encoder-decoder models. Future work should investigate how controlled stimuli can be extended to other reasoning dimensions, such as relational, causal, and temporal understanding, and how such targeted fine-tuning might complement large-scale pretraining. Bridging synthetic precision with real-world richness offers a path towards VLMs that not only perform well but also reason reliably and transparently across visual domains.

Limitations

Our controlled setup and exhaustive, bias-free analyses show that fine-tuning with controlled data is beneficial when tested on the spatial reasoning task of absolute position. However, absolute position represents one aspect of spatial reasoning, and future work should investigate whether similar benefits extend to other spatial reasoning tasks, as well as to other reasoning dimensions such as relational, causal, and temporal understanding. Moreover, due to computational and resource constraints, our experiments are limited to open-weight VLMs up to 8B parameters. Closed API-based models were also not considered, as they cannot be fine-tuned. Finally, the observed improvements are not present for CLIP, suggesting that the effectiveness of controlled fine-tuning may also depend on pre-training strategies and architectural design choices.

References

Manoj Acharya, Kushal Kafle, and Christopher Kanan. 2019. Tallyqa: Answering complex counting questions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):8076–8084.

Sumukh K Aithal, Pratyush Maini, Zachary C. Lipton, and J. Zico Kolter. 2024. Understanding hallucinations in diffusion models through mode interpolation. In *Advances in Neural Information Processing Systems*, volume 37, pages 134614–134644. Curran Associates, Inc.

Simone Alghisi, Gabriel Roccabruna, Massimo Rizzoli, Seyed Mahed Mousavi, and Giuseppe Riccardi. 2025. [del re] constructing vlms' reasoning in counting. *arXiv preprint arXiv:2510.19555*.

Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Didi Zhu, and 1 others. 2025. Llava-onevision-1.5: Fully open framework for democratized multimodal training. *arXiv preprint arXiv:2509.23661*.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. 2024a. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14455–14465.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. 2024b. Are we on the right way for evaluating large vision-language models? In *Advances in Neural Information Processing Systems*, volume 37, pages 27056–27087. Curran Associates, Inc.

Shiqi Chen, Tongyao Zhu, Ruochen Zhou, Jinghan Zhang, Siyang Gao, Juan Carlos Niebles, Mor Geva, Junxian He, Jiajun Wu, and Manling Li. 2025. Why is spatial reasoning hard for VLMs? an attention mechanism perspective on focus areas. In *Forty-second International Conference on Machine Learning*.

Chuanqi Cheng, Jian Guan, Wei Wu, and Rui Yan. 2024. From the least to the most: Building a plug-and-play visual reasoner via data synthesis. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4941–4957, Miami, Florida, USA. Association for Computational Linguistics.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, and 31 others. 2025. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 91–104.

Reza Esfandiarpoor, Cristina Menghini, and Stephen Bach. 2024. If CLIP could talk: Understanding vision-language model representations through their preferred concept descriptions. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 9797–9819, Miami, Florida, USA. Association for Computational Linguistics.

Stephanie Fu, tyler bonnen, Devin Guillory, and Trevor Darrell. 2025a. Hidden in plain sight: VLMs overlook their visual representations. In *Second Conference on Language Modeling*.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. 2025b. Blink: Multimodal large language models can see but not perceive. In *Computer Vision – ECCV 2024*, pages 148–166, Cham. Springer Nature Switzerland.

Xingyu Fu, Sheng Zhang, Gukyeong Kwon, Pramuditha Perera, Henghui Zhu, Yuhao Zhang, Alexander Hanbo Li, William Yang Wang, Zhiguo Wang, Vittorio Castelli, Patrick Ng, Dan Roth, and Bing Xiang. 2023. Generate then select: Open-ended visual question answering guided by world knowledge. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2333–2346, Toronto, Canada. Association for Computational Linguistics.

Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjun Zhong, Yufei Wang, Lanqing HONG, Jianhua Han, Hang Xu, Zhenguo Li, and Lingpeng Kong. 2025. G-LLaVA: Solving geometric problem with multi-modal large language model. In *The Thirteenth International Conference on Learning Representations*.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. 2017. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. 2023. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9161–9175, Singapore. Association for Computational Linguistics.

Seunghoi Kim, Chen Jin, Tom Diethe, Matteo Figini, Henry F. J. Tregidgo, Asher Mullokandov, Philip Teare, and Daniel C. Alexander. 2025. Tackling structural hallucination in image translation with local diffusion. In *Computer Vision – ECCV 2024*, pages 87–103, Cham. Springer Nature Switzerland.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. 2023. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, and 1 others. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73.

Haoxin Li and Boyang Li. 2025. Enhancing vision-language compositional understanding with multimodal synthetic data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24849–24861.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision – ECCV 2014*, pages 740–755, Cham. Springer International Publishing.

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

Zhenhua Ning, Zhuotao Tian, Shaoshuai Shi, Guangming Lu, Daojing He, Wenjie Pei, and Li Jiang. 2025. Enhancing spatial reasoning in multimodal large language models through reasoning-based segmentation. *arXiv preprint arXiv:2506.23120*.

Michael Ogezi and Freda Shi. 2025. SpaRE: Enhancing spatial reasoning in vision-language models with synthetic data. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7855–7875, Vienna, Austria. Association for Computational Linguistics.

Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. 2023. Teaching clip to count to ten. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3147–3157.

Jae Sung Park, Zixian Ma, Linjie Li, Chenhao Zheng, Cheng-Yu Hsieh, Ximing Lu, Khyathi Chandu, Quan Kong, Norimasa Kobori, Ali Farhadi, Yejin Choi, and Ranjay Krishna. 2025. Synthetic visual genome. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9073–9086.

Wujian Peng, Sicheng Xie, Zuyao You, Shiyi Lan, and Zuxuan Wu. 2024. Synthesize diagnose and optimize: Towards fine-grained vision-language understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13279–13288.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. 2024. Vision language models are blind. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 18–34.

Massimo Rizzoli, Simone Alghisi, Olha Khomyn, Gabriel Roccabruna, Seyed Mahed Mousavi, and Giuseppe Riccardi. 2025. CIVET: Systematic evaluation of understanding in VLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 4462–4480, Suzhou, China. Association for Computational Linguistics.

Karsten Roth, Jae Myung Kim, A. Sophia Koepke, Oriol Vinyals, Cordelia Schmid, and Zeynep Akata. 2023. Waffling around for performance: Visual classification with random words and broad concepts. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15700–15711.

William Rudman, Michal Golovanevsky, Amir Bar, Vedant Palit, Yann LeCun, Carsten Eickhoff, and Ritambhara Singh. 2025. Forgotten polygons: Multimodal large language models are shape-blind. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 11983–11998, Vienna, Austria. Association for Computational Linguistics.

Cristian Sbrolli, Matteo Matteucci, and Toshihiko Yamasaki. 2026. Auto-comp: An automated pipeline for scalable compositional probing of contrastive vision-language models. *arXiv preprint arXiv:2602.02043*.

Christoph Schuhmann, Robert Kaczmarczyk, Aran Komatsuzaki, Aarush Katta, Richard Vencu, Romain Beaumont, Jenia Jitsev, Theo Coombes, and Clayton Mullis. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. In *NeurIPS Workshop Datacentric AI*, FZJ-2022-00923. Jülich Supercomputing Center.

Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See-Kiong Ng, Lidong Bing, and Roy Ka-Wei Lee. 2024. Math-LLaVA: Bootstrapping mathematical reasoning for multimodal large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4663–4680, Miami, Florida, USA. Association for Computational Linguistics.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. 2022. Winoground: Probing vision and language models for visio-linguistic compositionality. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5228–5238.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Weizhen Wang, Chenda Duan, Zhenghao Peng, Yuxin Liu, and Bolei Zhou. 2025. Embodied scene understanding for vision language models via metavqa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22453–22464.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, and 3 others. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9556–9567.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. 2023. When and why vision-language models behave like bags-of-words, and what to do about it? In *The Eleventh International Conference on Learning Representations*.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Yu Qiao, Peng Gao, and Hongsheng Li. 2025a. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *Computer Vision – ECCV 2024*, pages 169–186, Cham. Springer Nature Switzerland.

Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Ziyu Guo, Yichi Zhang, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Shanghang Zhang, Peng Gao, and Hongsheng Li. 2025b. MAVIS: Mathematical visual instruction tuning with an automatic data engine. In *The Thirteenth International Conference on Learning Representations*.

Fangrui Zhu, Jianwei Yang, and Huaizu Jiang. 2024. Towards flexible visual relationship segmentation. In *Advances in Neural Information Processing Systems*, volume 37, pages 107633–107661. Curran Associates, Inc.

A Dataset Construction

We report details on the construction of the datasets complementing Section 3.2.

Synthetic Test Set We build an exhaustive synthetic evaluation dataset for the absolute position task. Each image contains a single object on a uniform black background. We systematically vary four object attributes: color, shape, size, and position. We use six colors (red, green, blue, cyan, magenta, yellow), four shapes (circle, triangle, square, star), and two sizes (regular and small[5]). Following the results of CIVET (Rizzoli et al., 2025), we generate images of 672×672 pixels, a multiple of the input size of the vision encoder of CLIP, shared across several VLMs. To capture fine-grained spatial variation, each image is divided into 9×9 cells. Objects are placed in these cells, allowing for more fine-grained analyses on how models' answers about the absolute position vary within the 3×3 ground truth regions. For each combination of attributes, we generate a corresponding VQA sample following the formulation in Section 3.1. This process yields 3,888[6] balanced image-question pairs.

Synthetic Training Set We construct a training dataset with the same structure as the evaluation data but distinct color-shape combinations to avoid overlap. We include the four shapes (*circle*, *triangle*, *square*, *star*) in white and introduce *plus* as an unseen shape in the aforementioned six colors. This preserves balance across visual attributes while ensuring no color-shape combination is shared between training and testing. Images follow the same 672×672 layout and VQA formulation described in Section 3.1. The resulting dataset comprises 1,620[7] image-question pairs, balanced across all positions.

Real-World Dataset We construct training and test datasets starting from the train and validation splits of COCO, as test annotation is not provided. For each image, we generate one or more VQA samples querying the position of a specific object category, e.g., *"Where is the person?"*. To ensure unambiguous questions, we include only objects that appear as the only instance of their category. The position of each target object is computed as the center of its bounding box and assigned to one of the nine grid regions defined in Section 3.1. We obtain a training set of 201,358 questions and 95,899 images, and an evaluation set of 8,548 questions and 4,109 images. We split the training set, keeping 80% (161,086) for training and 20% (40,272) for validation.

Synthetic Set with Distractors Real-world scenes often contain multiple objects, many of which are irrelevant to the query. To approximate this complexity and study robustness, we extend the synthetic datasets by adding distractor objects. Each image includes one target object (referenced in the question), and one or more distractors that differ in color, shape, or both. This design allows us to test whether exposure to cluttered visual contexts during fine-tuning improves the model's ability to attend to task-relevant information. We generate variants containing one, three, or five distractors per image. For images with white target shapes, distractors vary only in shape while retaining the white color; for colored plusses, distractors vary in color while maintaining the same shape. All distractors are placed in random non-overlapping positions within the 9×9 cell grid.

B Fine-tuning Details

Each model is fine-tuned using LoRA (Hu et al., 2022) with a rank of 32 and α of 64. Following standard practice and recent findings emphasizing the role of attention in spatial reasoning (Chen et al., 2025), LoRA adapters are applied to the query, key, and value matrices of the attention layers. Fine-tuning is performed for up to 10 epochs with early stopping patience of 2 epochs, a learning rate of 10^{-4}, using 80% of the training split for optimization and reserving the remaining 20% for validation. Models are fine-tuned and tested on the *Absolute Position* task (section 3.1). Each model is prompted with the image and a closed-ended question, and answers are obtained through greedy decoding. A response is marked as correct only if it contains exactly one of the predefined positional labels. To reduce output verbosity, which can hinder automatic evaluation, as observed in (Rizzoli et al., 2025), each question is prefixed with the instruction *"Answer with as few words as possible."*, which has been shown to reliably constrain the model's output to one of the valid options (see attached data samples for examples of the complete prompt). For CLIP, which follows a dual-encoder

[5] A small object has half the height and width of a regular one

[6] Computed as 6 colors $\times$ 4 shapes $\times$ 2 sizes $\times$ 81 positions (i.e., 9×9 cells)

[7] Computed as (6 colored plusses + 4 white shapes) $\times$ 2 sizes $\times$ 81 positions

Model	Accuracy (%)
LLaVA-NeXT	100 ±1 (↑ 58)
LLaVA-OneVision	100 ±0 (↑ 33)
Molmo	96 ±0 (↑ 34)
Qwen2-VL	99 ±0 (↑ 38)
CLIP	100 ±0 (↑ 88)

Table 4: **Effect of fine-tuning on synthetic data.** Accuracy on the *Absolute Position* task for models fine-tuned and evaluated on the Synthetic Test Set. (↑ Value) shows the absolute improvement with respect to the *Base Model*. Fine-tuning leads to near-perfect performance across all models.

Model	**Training Set** (#Samples)	**Test Set Accuracy (%)**	
		Synthetic	**COCO**
LLaVA-NeXT	*Base Model*	42	30
	Synthetic (1.3k)	100 ±1	43 ±17
	COCO *Complete* (161k)	0 ±0	0 ±0
LLaVA-OneVision	*Base Model*	67	45
	Synthetic (1.3k)	100 ±0	65 ±3
	COCO *Complete* (161k)	11 ±0	26 ±1
Molmo	*Base Model*	62	37
	Synthetic (1.3k)	96 ±3	58 ±5
	COCO *Complete* (161k)	4 ±5	6 ±4
Qwen2-VL	*Base Model*	61	39
	Synthetic (1.3k)	99 ±0	60 ±4
	COCO *Complete* (161k)	9 ±4	20 ±8
CLIP	*Base Model*	12	22
	Synthetic (1.3k)	100 ±0	22 ±9
	COCO *Complete* (161k)	11 ±0	36 ±0

Table 5: **Cross-domain transfer to real-world data**. Accuracy (%) on the *Absolute Position* task for models fine-tuned on balanced Synthetic (1.3k), and COCO Complete (161k) test sets. ± denotes the standard deviation obtained from 5 runs.

Model	**Test Set Accuracy (%)**	
	Synthetic	**COCO**
LLaVA-NeXT	71 ±10	67 ±2
LLaVA-OneVision	77 ±6	64 ±3
Molmo	80 ±4	45 ±2
Qwen2-VL	80 ±5	61 ±4
CLIP	13 ±0	36 ±1

Table 6: **Fine-Tuning on balanced real-world data**. Accuracy (%) on the *Absolute Position* task for models fine-tuned on a COCO Subset (1.3k), balanced in terms of object category and position. ± denotes the standard deviation obtained from 5 runs.

architecture, we reformulate the task as an image-text retrieval problem. For each of the nine possible answers, we generate a textual candidate consisting of the same question followed by the position label, encode both the image and the text, and select the answer corresponding to the text representation with the highest cosine similarity to the image embedding. We fine-tune CLIP starting from the cross-entropy loss used in its original training (Radford et al., 2021), but only using the component that optimizes for the selection of the correct text candidate given an image.

All experiments were run on a single Nvidia A100 GPU of 80GB. Following, we report the HuggingFace checkpoints used for each model:

- LLaVA-NeXT 7B: `https://huggingface.co/llava-hf/llava-v1.6-vicuna-7b-hf`
- LLaVA-OneVision-8B-Instruct: `https://huggingface.co/lmms-lab/LLaVA-OneVision-1.5-8B-Instruct`
- Molmo 7B-O: `https://huggingface.co/allenai/Molmo-7B-O-0924`
- Qwen2-VL-7B-Instruct: `https://huggingface.co/Qwen/Qwen2-VL-7B-Instruct`
- CLIP ViT-L/14-336px: `https://huggingface.co/openai/clip-vit-large-patch14`

C Additional Results

C.1 Fine-tuning on Synthetic Data

Table 4 reports the accuracy for models fine-tuned and tested on the synthetic data, averaged over 5 runs. All models achieve close to perfect accuracy with minimal variations across runs.

C.2 Cross-Domain Transfer

Table 5 reports the accuracy for models fine-tuned on synthetic data and tested on COCO (unmatched setting) and models fine-tuned and tested on COCO (matched setting), extending the results of Table 1 (section 4) with the standard deviation obtained from five runs. Similarly, Table 6 reports the standard deviation for models fine-tuned on the COCO Subset, balanced in terms of object category and position, extending the results of Table 2 (section 4). the

C.3 Cell-level Accuracy

We report the cell-level accuracy for LLaVA-NeXT (fig. 6), LLaVA-OneVision (fig. 7), Molmo (fig. 8), and CLIP (fig. 9). Similarly to Qwen2-VL (see Figure 5 in section 4), the encoder-decoder VLMs initially show strong spatial biases and after fine-tuning on synthetic data, the performance on the synthetic set becomes close to uniform. This is also

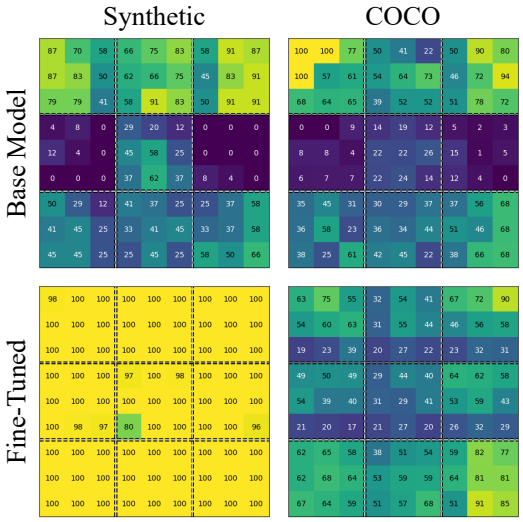


Figure 6: **Cell-level accuracy of LLaVA-NeXT 7B before and after fine-tuning** on synthetic data, on both the synthetic and COCO Absolute Position test sets.

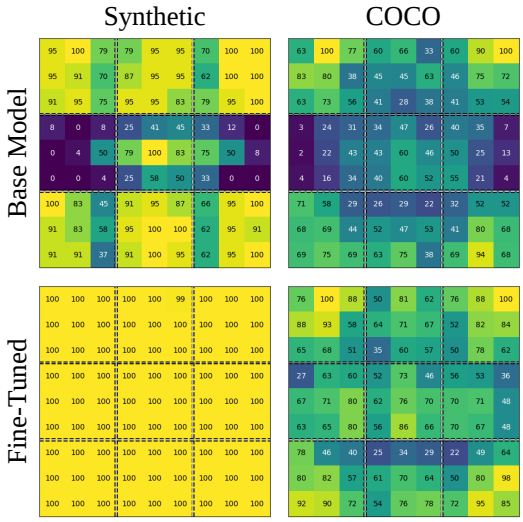


Figure 7: **Cell-level accuracy of LLaVA-OneVision 8B before and after fine-tuning** on synthetic data, on both the synthetic and COCO Absolute Position test sets.

reflected on the COCO test set, with LLaVA-NeXT and Molmo obtaining most of the improvement where performance was lowest. Instead, while CLIP obtains perfect accuracy across positions on the synthetic test set after fine-tuning, this improvement does not transfer to the COCO test set.

C.4 Scene Complexity & Distractors

In Table 7, we present additional results on increasing scene complexity. These results extend those in Table 3 (section 5) by including evaluating on the synthetic test set with the same number of distractors used during fine-tuning and evaluating on the synthetic test with five distractors. Regardless of fine-tuning data, the encoder-decoder models show a decrease in performance when scene complexity increases. When tested with five distractors, Molmo and Qwen2-VL show little to no benefit from fine-tuning with distractors, while LLaVA-NeXT and LLaVA-OneVision show a substantial gain with as few as one distractor seen during fine-tuning. However, adding five distractors to LLaVA-NeXT results in reduced performance on all test sets, suggesting only moderate complexity is beneficial for the model.

C.5 Layer-Wise Analysis

We report the results for the layer-wise analysis for LLaVA-NeXT (fig. 10), LLaVA-OneVision (fig. 11), Molmo (fig. 12), and Qwen2-VL (fig. 13). The models show a similar trend, rapidly improving performance in the initial layers on synthetic data, while having a slower rise on the more complex scenes of the COCO test set. For all models, fine-tuning improves the representations for synthetic data. For LLaVA-OneVision, Molmo, and Qwen2-VL, this improvement transfers to real-world data. However, LLaVA-NeXT shows mildly reduced performance after fine-tuning, with high variability across runs. Together with the improvement shown on the Synthetic test, this suggests LLaVA-NeXT is more prone to overfitting on the synthetic data. This is in line with the experiments on training set scale (fig. 3b in section 4), where LLaVA-NeXT obtains the most transfer after fine-tuning on 10% of the Synthetic training set, with performance decreasing with larger subsets.

Model	Training Set	Test Set			
		Synthetic	Synth. w. N Distr.	Synth. w. 5 Distr.	COCO
LLaVA-NeXT	*Base Model*	42	—	36	30
	Synthetic (1.3k)	100 ±1	—	58 ±21	42 ±16
	with $N = 1$ Distr.	100 ±0	94 ±3	77 ±6	42 ±16
	with $N = 3$ Distr.	100 ±0	89 ±7	82 ±0	54 ±6
	with $N = 5$ Distr.	81 ±23	65 ±22	65 ±22	48 ±21
LLaVA-OneVision	*Base Model*	67	—	64	45
	Synthetic (1.3k)	100 ± 0	—	89 ± 3	65 ± 3
	with $N = 1$ Distr.	100 ± 0	100 ± 1	98 ± 2	66 ± 3
	with $N = 3$ Distr.	100 ± 0	99 ± 1	98 ± 1	60 ± 6
	with $N = 5$ Distr.	100 ± 0	98 ± 1	98 ± 1	60 ± 9
Molmo	*Base Model*	62	—	59	39
	Synthetic (1.3k)	96 ±3	—	93 ±2	57 ±5
	with $N = 1$ Distr.	96 ±2	95 ±2	91 ±3	60 ±2
	with $N = 3$ Distr.	95 ±3	93 ±4	92 ±5	60 ±6
	with $N = 5$ Distr.	97 ±2	92 ±2	92 ±2	65 ±1
Qwen2-VL	*Base Model*	61	—	53	38
	Synthetic (1.3k)	99 ±0	—	92 ±8	58 ±4
	with $N = 1$ Distr.	98 ±2	98 ±2	93 ±4	59 ±3
	with $N = 3$ Distr.	93 ±5	93 ±4	92 ±5	58 ±4
	with $N = 5$ Distr.	90 ±3	88 ±7	88 ±7	54 ±3
CLIP	*Base Model*	12	—	11	22
	Synthetic (1.3k)	100 ±0	—	15 ±0	22 ±9
	with $N = 1$ Distr.	25 ±30	25 ±30	16 ±10	14 ±5
	with $N = 3$ Distr.	11 ±0	11 ±0	11 ±0	22 ±10
	with $N = 5$ Distr.	11 ±0	11 ±0	11 ±0	28 ±2

Table 7: **Effect of Distractors on the *Absolute Position* task** for VLMs fine-tuned on the synthetic dataset, when evaluated on Synthetic (no distractors), Synthetic with the same number of Distractors as fine-tuning, Synthetic with 5 Distractors, and on COCO. Results show the average accuracy (%) across five runs, and ± denotes the standard deviation.

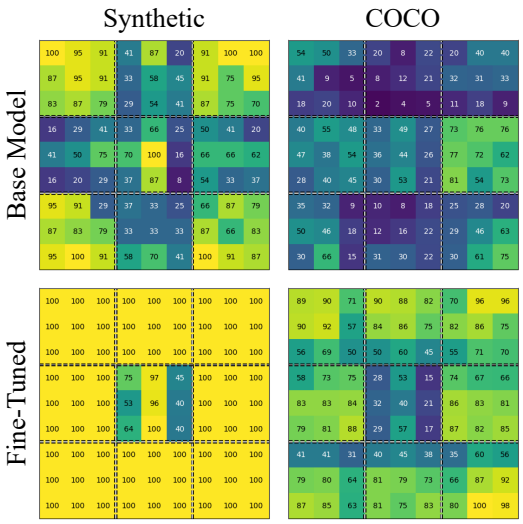


Figure 8: **Cell-level accuracy of Molmo 7B before and after fine-tuning** on synthetic data, on both the synthetic and COCO Absolute Position test sets.

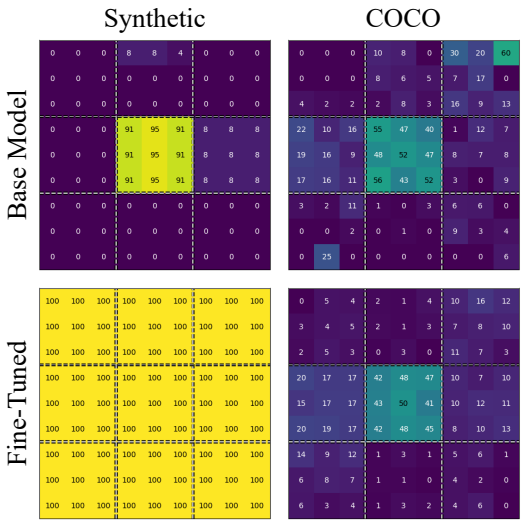


Figure 9: **Cell-level accuracy of CLIP before and after fine-tuning** on synthetic data, on both the synthetic and COCO Absolute Position test sets.

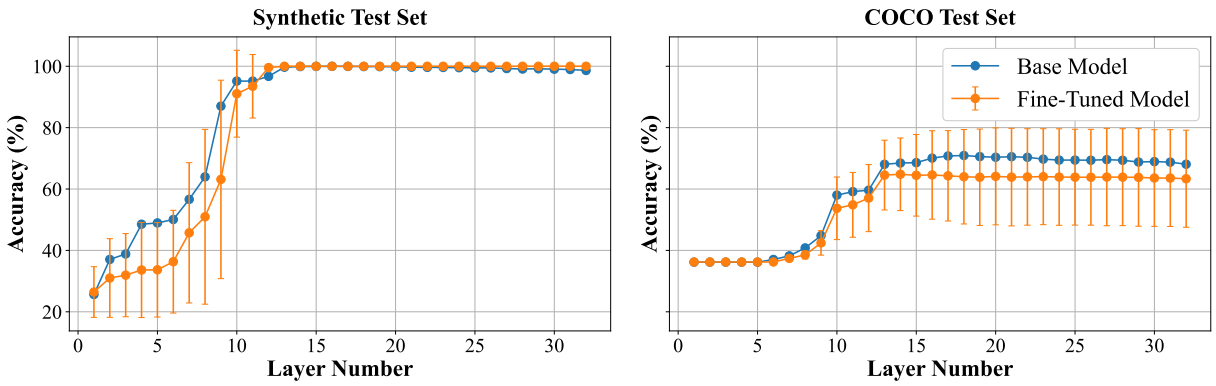


Figure 10: **Layer-wise probing accuracy of LLaVA-NeXT 7B** before (blue) and after (orange) fine-tuning on the synthetic dataset, evaluated on Synthetic (left) and COCO (right). Error bars represent standard deviation across fine-tuning runs.

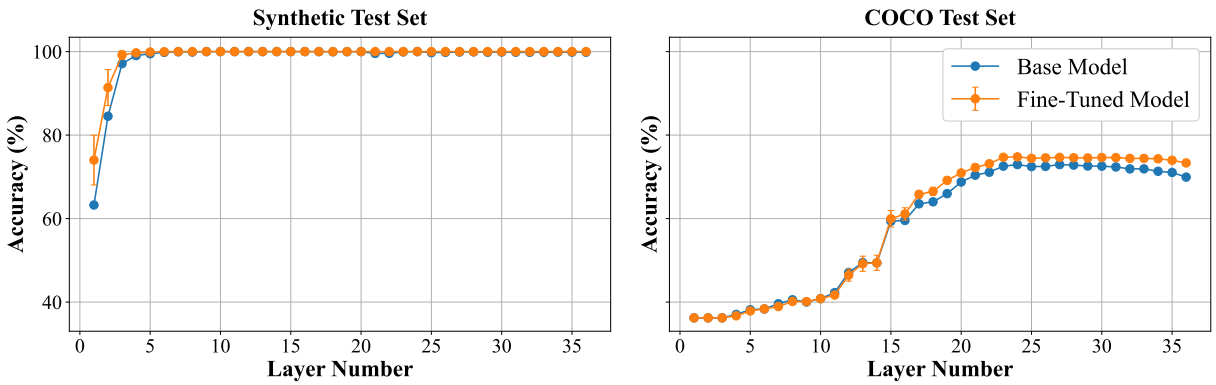


Figure 11: **Layer-wise probing accuracy of LLaVA-OneVision 8B** before (blue) and after (orange) fine-tuning on the synthetic dataset, evaluated on Synthetic (left) and COCO (right). Error bars represent standard deviation across fine-tuning runs.

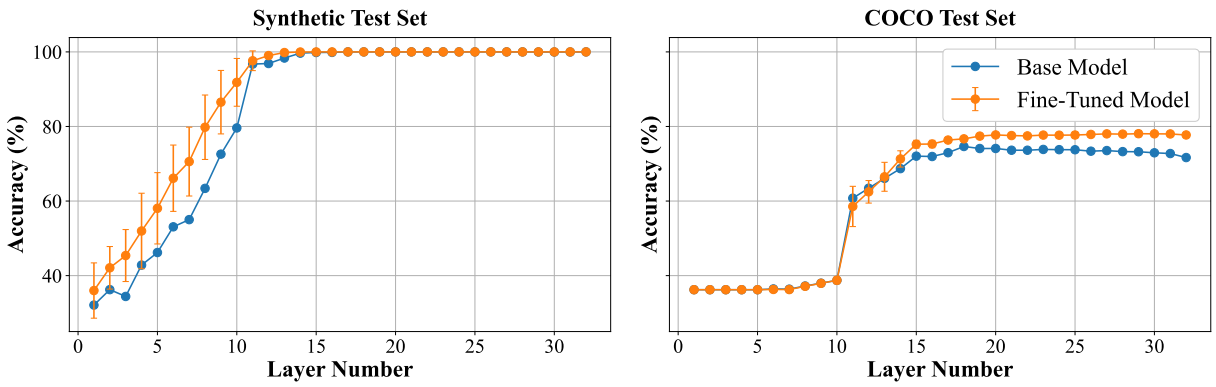


Figure 12: **Layer-wise probing accuracy of Molmo 7B** before (blue) and after (orange) fine-tuning on the synthetic dataset, evaluated on Synthetic (left) and COCO (right). Error bars represent standard deviation across fine-tuning runs.

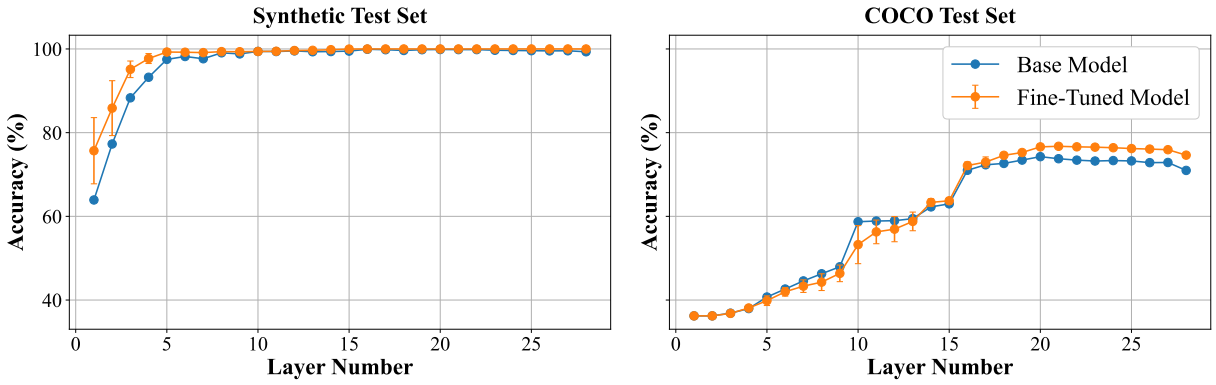


Figure 13: **Layer-wise probing accuracy of Qwen2-VL 7B** before (blue) and after (orange) fine-tuning on the synthetic dataset, evaluated on Synthetic (left) and COCO (right). Error bars represent standard deviation across fine-tuning runs.