VISIONFOUNDRY: TEACHING VLMS VISUAL PERCEPTION WITH SYNTHETIC IMAGES

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language models (VLMs) still struggle with visual perception tasks such as spatial understanding and viewpoint recognition, largely because natural image datasets provide limited supervision for low-level visual skills. Can targeted synthetic supervision address these weaknesses without reference images or manual annotation? To investigate this, we introduce **VisionFoundry**, an automated pipeline that takes only a task name as input, uses LLMs to synthesize paired questions, answers, and text-to-image (T2I) prompts, generates images with T2I models, and filters samples via multimodal verification. With **VisionFoundry**, we construct **VisionFoundry-10K**, a synthetic VQA dataset spanning 10 perception tasks. Finetuning on **VisionFoundry-10K** consistently improves perception benchmarks across three open-source backbones (e.g., +6.7% on MMVP-pair and +10.5% on CV-Bench-3D for Qwen2.5-VL-3B-Instruct) while preserving broader capabilities and showing positive data scaling. The same synthetic supervision also yields consistent gains under reinforcement learning (RL) across all three backbones, and the framework remains effective under open-source synthesis and self-verification. Our findings demonstrate that automated synthetic supervision offers an effective and scalable path toward systematic VLM training. Code: link.

1 INTRODUCTION

Vision-language models (VLMs) have rapidly evolved into general-purpose systems capable of processing and reasoning over interleaved text and images (Liu et al., 2023; Team, 2023; Dai et al., 2023; Zhu et al., 2024). They now support a broad spectrum of applications, from visual question answering (Liu et al., 2023; 2024a) and optical character recognition (Liu et al., 2024b) to multimodal reasoning (Yue et al., 2024; Chen et al., 2024e), graphical user interface (GUI) grounding (Li et al., 2025), and mathematical problem-solving (Lu et al., 2024). Despite these advances, VLMs still exhibit persistent weaknesses in visual perception, as highlighted by recent diagnostic benchmarks (e.g., MMVP, CV-Bench, and RealWorldQA). Tong et al. (2024b;a) construct benchmarks that decouple visual perception from pure language priors, while RealWorldQA (xAI, 2024) stress-tests geometric and spatial reasoning.

One plausible contributing factor behind this perception bottleneck is limited supervision for these low-level visual skills in natural image datasets. Natural image–text corpora, while vast, may not systematically cover the combinatorial range of variations required for robust visual perception. This motivates a practical question: can we synthesize targeted supervision to address these weaknesses without relying on reference images or expensive human annotation?

Synthetic data is a core ingredient in LLM training. It can be generated on demand, targeted to specific failures, and iterated rapidly (Gunasekar et al., 2023; Abdin et al., 2024; Qin et al., 2025). A similar trend is emerging in vision, where text-to-image (T2I) models enable synthetic-caption generation and image–text corpus construction (Tian et al., 2023; Fan et al., 2024; Sharifzadeh et al., 2024). They also support multimodal tuning focused on specific visual skills (Liu et al., 2025; Jiao et al., 2025; Li & Li, 2025; Wu et al., 2025a). Synthetic data is near-unbounded and highly controllable, allowing systematic coverage of entity, attribute, relation, viewpoint, and style combinations that are rare in natural corpora and otherwise expensive to curate. For VLMs, the more useful question is what makes synthetic images good supervision: they should provide task-specific, reliable, and scalable training signals without reference images or expensive human annotation, while improving visual perception without noticeably degrading broader capabilities (Singh et al., 2024; Hu et al., 2025; Ben-Melech Stan et al., 2025; Wu et al., 2025c).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

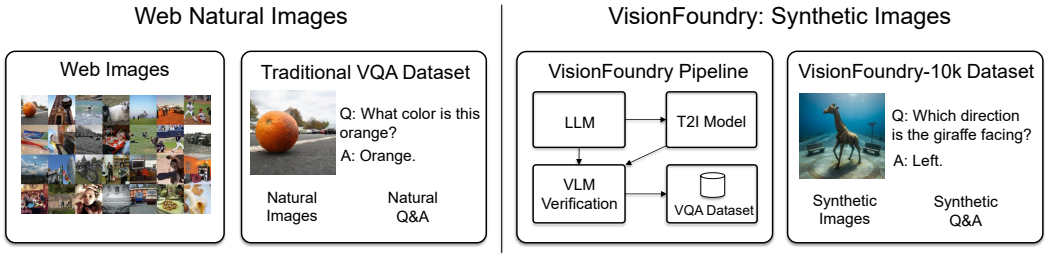

Figure 1: **Training VLMs with synthetic images.** VisionFoundry uses a task-keyword-only pipeline that requires no reference text or images: it generates task-aware text-to-image (T2I) prompts and question–answer pairs, synthesizes images, and uses the resulting supervision to improve capabilities such as visual perception in VLMs. Compared with traditional web-image collection pipelines, this approach provides more controllable and task-targeted supervision with less retrieval noise.

We build **VisionFoundry**, an end-to-end synthetic data-generation pipeline that takes only a task name and produces verifier-filtered image–question–answer triples (see Figure 1). **VisionFoundry** composes three components into a closed-loop system (Figure 2): (1) a large language model generates question–answer pairs and detailed T2I prompts conditioned on the target task; (2) a modern T2I model synthesizes images conditioned on those prompts; and (3) a strong multimodal judge verifies alignment between the generated image and the answer-determining visual statement, filtering out misaligned samples. This pipeline eliminates the need for reference images and manual labeling, reducing cost while using the native cross-modal capabilities of T2I models.

Using **VisionFoundry**, we construct **VisionFoundry-10K**, a synthetic visual question answering (VQA) dataset of 10k image–question–answer triples spanning 10 low-level visual perception tasks (1k samples per task), including spatial understanding, relative depth, and viewpoint variation (Section 3). We use this controlled setting as a validation to isolate the direct contribution of synthetic images to visual capability learning, rather than gains primarily driven by text-heavy reasoning supervision. These tasks are motivated by benchmark analyses that identify spatial and perceptual failures as persistent weaknesses of contemporary VLMs (Tong et al., 2024b;a; xAI, 2024).

Across three representative open-source models, Qwen2.5-VL-3B-Instruct (Bai et al., 2025), Llama-3.2-11B-Vision-Instruct (Meta AI, 2024), and MiMo-VL-7B-SFT (Yue et al., 2025), finetuning on **VisionFoundry-10K** consistently improves visual perception benchmarks in our experiments, especially on Qwen2.5-VL-3B-Instruct, with gains such as +6.7% on MMVP-pair and +10.5% on CV-Bench-3D, while preserving broader capabilities and showing an upward trend as the data size increases (Section 4). We also evaluate on broader multimodal reasoning, OCR, and GUI grounding benchmarks, where models maintain competitive general capabilities. We further observe that an equal-sized synthetic–natural mixture outperforms pure natural subsets, suggesting that synthetic supervision provides complementary signals that are hard to obtain from limited real data alone (Tian et al., 2023; Fan et al., 2024; Singh et al., 2024). Beyond supervised finetuning (SFT), we further find that the same synthetic supervision improves visual perception under RL.

Our results suggest that limited task-targeted supervision is an important factor behind the perception bottleneck in VLMs and that synthetic data curation can meaningfully alleviate part of this weakness. Beyond immediate benchmark improvements, **VisionFoundry** represents a systematic step toward more systematic synthetic-data workflows for training VLMs. Our results further suggest that synthetic images can serve as a useful primitive for post-training beyond SFT, addressing capability gaps as new evaluations emerge. They may also support pretraining regimes with broader compositional coverage than web-scale natural images can provide. We hope this work can catalyze a broader shift toward fully automated synthetic multimodal data as a promising path to stronger VLMs.

2 VISIONFOUNDRY

VisionFoundry is a task-aware synthetic data generation pipeline that produces high-quality VQA supervision for VLMs using only high-level task specifications, without reference images, manual QA annotations, or real image–caption pairs. The methodology is guided by three core principles: *controllability*, *visual determinism*, and *verification*. *Controllability* leverages explicit task configurations and entity pools to systematically cover targeted capabilities. *Visual determinism* encodes

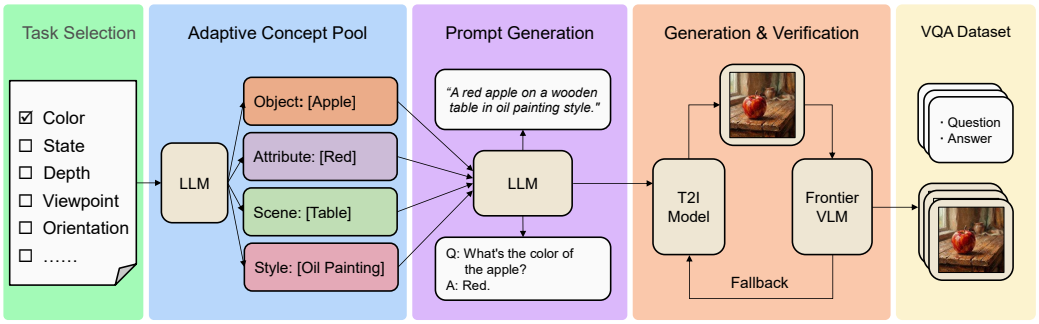

Figure 2: **VisionFoundry overview.** Given only task keywords, an LLM builds an adaptive concept pool; compositional sampling forms entities for T2I prompts; and a T2I model synthesizes images that a frontier VLM verifies to yield a high-quality VQA dataset.

answer-determining visual facts directly into the prompt so that questions are answerable solely from the synthesized image. *Verification* employs an automated verifier to filter subtle misalignments between text specifications and visual content that would otherwise inject noise into supervision.

Operationally, given a high-level task specification (e.g., spatial understanding, counting, depth reasoning), **VisionFoundry** has three stages: (1) task-aware generation of questions, answers, and T2I prompts; (2) image synthesis conditioned on the generated prompts; and (3) alignment verification and filtering using a strong multimodal judge. Together, these stages produce supervision that is visually grounded, unambiguous, and robust. Appendix I and Appendix J show that **VisionFoundry** remains effective with open-source alternatives. Figure 2 provides an overview of the pipeline, with its left, middle, and right parts corresponding to Sections 2.1, 2.2, and 2.3, respectively. We provide the exact pipeline prompts in the appendix for transparency and reproducibility in downstream audits.

2.1 VQA Triplet Generation

Task-aware generation. The left part of Figure 2 corresponds to the first stage of **VisionFoundry**, which constructs a triple consisting of a question, its answer, and a corresponding T2I prompt. This stage is driven by GPT-5.2 (OpenAI, 2025) and conditions generation on an explicit *task configuration*. Each task specifies the target capability (e.g., spatial relations, color discrimination), the number of objects involved, and optional constraints such as required attributes or relations. To support systematic coverage, we construct an *entity pool* composed of objects, attributes, scenes, styles, and task-specific custom dimensions. Entities are randomly sampled from the Cartesian product of these dimensions, yielding structured visual configurations.

Given a sampled entity, the LLM is prompted to generate: (i) a question whose answer is fully determined by visible content, (ii) a concise and deterministic answer, and (iii) a detailed T2I prompt that explicitly encodes the answer-determining visual facts. To avoid ambiguity, the LLM is instructed to rely exclusively on visually verifiable properties and to avoid commonsense assumptions.

Design rationale. This design enforces tight coupling between language supervision and visual content at generation time. By embedding the correct answer directly into the T2I prompt, we reduce the risk of producing images that are irrelevant or underspecified for the question. This is a common failure mode in synthetic multimodal data.

2.2 Image Synthesis

Prompt-conditioned generation. The second stage of **VisionFoundry** synthesizes images conditioned on the T2I prompts (Figure 2, middle). We use Gemini-2.5-Flash-Image (Google Cloud, 2025a; Comanici et al., 2025) as the generator. Each image is generated from its corresponding prompt, which specifies key objects, attributes, spatial arrangements, and visual styles to ensure that the image contains all information required to answer the associated question. Our pipeline does not rely on model-specific internals, ensuring straightforward compatibility with alternative generators.

Selective iterative refinement. To improve yield, **VisionFoundry** allows limited iterative refinement of images. If an initially generated image fails downstream verification, the system can request a localized edit that minimally modifies the image to better satisfy the intended visual statement. This



Figure 3: **VisionFoundry-10K examples.** Randomly selected examples from our dataset, covering all 10 tasks. The task names are annotated at the top and serve as the only valid input to the pipeline. Each panel shows a generated image, its corresponding question, and a ground-truth answer.

refinement process is conceptually similar to prompt-based image editing, but is only used when necessary and does not alter the original question and answer pair.

2.3 Alignment Verification and Filtering

Statement-based verification. In the final stage of **VisionFoundry** (Figure 2, right), the pipeline verifies whether each synthesized image is strictly consistent with its corresponding question and answer, ensuring dataset quality while remaining fully automatic. Given a question and candidate answer, we convert them into a short declarative visual statement capturing the key visual fact (e.g., "The red cube is to the left of the blue sphere"). A strong multimodal judge, instantiated as Gemini-3-Pro (Google Cloud, 2025b), inspects the image and statement, returning an accept/reject decision based on whether the core visual relation is present while ignoring minor stylistic variations. Importantly, this verification stage is fully automated and involves no manual intervention.

Binary filtering criterion. Only samples for which the judge confirms alignment are retained. If an image fails verification after limited refinement attempts, the entire sample is discarded and a new entity is sampled. By removing human annotation and manual correction, our design deliberately prioritizes end-to-end automation and scalability, allowing us to directly evaluate the viability and scaling dynamics of fully automated synthetic multimodal supervision.

Why verification matters. Without explicit verification, synthetic datasets are prone to subtle misalignments, such as missing objects, incorrect spatial relations, or visually ambiguous scenes. By incorporating a multimodal verifier, **VisionFoundry** provides an automated filtering stage that screens out inconsistent samples before downstream training. Empirically, current multimodal models provide reliable alignment signals for these visual distinctions (detailed verification-accuracy records are provided in Appendix E). Appendix J further shows that the downstream Qwen backbone itself can serve as the verifier: a full 10-task dataset constructed with Qwen-based image synthesis and self-verification still improves all five visual-perception benchmarks after finetuning.

Benchmark	Qwen2.5-VL-3B Instruct		MiMo-VL-7B SFT		Llama-3.2-11B Vision-Instruct	
	Baseline	Synth	Baseline	Synth	Baseline	Synth
Visual Perception Benchmarks						
$\text{MMVP}_{\text{pair}}$ (Tong et al., 2024b)	35.3	**42.0**	43.3	**57.3**	42.7	**46.7**
$\text{MMVP}_{\text{single}}$ (Tong et al., 2024b)	64.3	**68.3**	66.7	**77.7**	70.3	**71.7**
CV-Bench-2D (Tong et al., 2024a)	67.3	**72.4**	74.3	**79.0**	70.4	**71.7**
CV-Bench-3D (Tong et al., 2024a)	66.0	**76.5**	72.3	**83.7**	74.4	**75.3**
RealWorldQA (xAI, 2024)	65.0	**66.9**	65.9	**67.5**	63.0	**64.6**
General Understanding and Reasoning Benchmarks						
BLINK (Fu et al., 2024b)	48.1	47.9	58.9	58.7	34.3	**35.5**
MMMU_{Val} (Yue et al., 2024)	48.0	47.1	49.4	**52.3**	40.1	**40.7**
$\text{MMBench}_{\text{EN v1.1}}$ (Liu et al., 2024a)	75.8	**76.1**	50.5	**81.6**	66.6	**73.8**
MMStar (Chen et al., 2024e)	55.7	54.5	53.7	**62.7**	49.0	**50.9**
$\text{MathVista}_{\text{Mini}}$ (Lu et al., 2024)	62.3	**62.9**	39.1	**52.4**	47.0	**48.3**
Application-Specific Benchmarks						
$\text{ScreenSpot-Pro}_{\text{avg}}$ (Li et al., 2025)	19.4	**20.2**	3.4	**5.8**	0.0	**6.4**
OCRBench (Liu et al., 2024b)	82.7	82.4	83.3	82.9	75.4	75.2
MMSI-Bench (Yang et al., 2026)	9.2	**9.9**	8.4	**11.5**	3.9	**8.9**
3DSRBench (Ma et al., 2025)	35.2	**36.8**	32.0	**40.0**	27.5	**39.2**
LEGO (Tang et al., 2025)	15.8	14.4	24.0	**28.8**	5.3	**6.2**

Table 1: **Main Benchmark Results.** Main results on 13 benchmarks with 15 reported metrics across three VLMs. Models trained on **VisionFoundry-10K** improve visual perception performance while maintaining general multimodal capabilities. We report the mean score over four independent runs. Visual perception benchmarks are highlighted at the top.

Model	$\text{MMVP}_{\text{pair}}$	$\text{MMVP}_{\text{single}}$	CV-Bench-2D	CV-Bench-3D	RealWorldQA
Qwen2.5-VL-3B-Instruct	46.0 (+10.7)	71.3 (+7.0)	72.9 (+5.6)	72.3 (+6.3)	67.2 (+2.2)
Llama-3.2-11B-Vision-Instruct	45.3 (+2.6)	70.7 (+0.4)	72.4 (+2.0)	76.3 (+1.9)	63.8 (+0.8)
MiMo-VL-7B-SFT	48.0 (+4.7)	73.3 (+6.6)	78.7 (+4.4)	75.8 (+3.5)	69.2 (+3.3)

Table 2: **RL results on visual perception benchmarks.** Evaluation of GRPO across three VLM backbones on the 10-task **VisionFoundry-10K** dataset. Numbers in parentheses indicate absolute gains over each model's baseline.

The output of the pipeline is a fully synthetic collection of image–question–answer triples, formatted for instruction tuning. Each retained sample is verifier-approved and intended to be visually grounded and answerable solely from the image. Overall, **VisionFoundry** demonstrates how modern LLMs and T2I models can be composed into a principled system for synthetic multimodal data generation and provides a practical framework for studying improvements in current VLMs' visual understanding.

3 VisionFoundry-10K Dataset

Using **VisionFoundry**, we construct **VisionFoundry-10K**, a synthetic VQA dataset targeting visual perception. **VisionFoundry-10K** contains 10k image–question–answer triples across 10 tasks, with 1k cases per task. All images are generated by T2I models and filtered through automated multimodal alignment verification as described in Section 2. Figure 3 shows **VisionFoundry-10K** examples.

Task selection. The tasks are chosen to emphasize visual perception skills that are known to be challenging for current VLMs and are weakly correlated with pure language ability. We treat **VisionFoundry-10K** as a first-step validation of synthetic images as training supervision. We therefore prioritize visual perception tasks over text-heavy or long-chain reasoning tasks to isolate the visual learning effect from improvements that come mainly from textual reasoning signals. In particular, we focus on perception-oriented tasks that depend on spatial layout, depth, and relative attributes, rather than high-level semantics or commonsense knowledge. The 10 tasks included in **VisionFoundry-10K** are *Orientation and Direction*, *Viewpoint and Perspective*, *Positional and Relational Context*, *Spatial Relationship*, *State and Condition*, *Structural and Physical Characteristics*, *Color and Appearance*, *Depth Order*, *Relative Distance*, and *Real-World Spatial Understanding*.

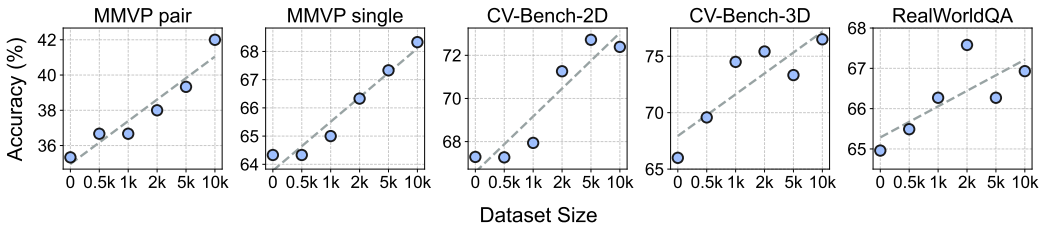

Figure 4: **Data-Size Effects of Synthetic Supervision.** We report performance trends as the amount of **VisionFoundry** synthetic VQA supervision increases. "10k" corresponds to the full **VisionFoundry-10K** dataset (10k samples), while "500/1k/2k/5k" are random subsets sampled from it. The training recipe matches the main experimental setting. Overall, results show an upward trend on visual perception benchmarks as synthetic data increases.

Dataset construction and statistics. For each task, we define a task-specific configuration that constrains the number of objects, admissible attributes, and relational predicates. Entity configurations are systematically sampled from an underlying entity pool, ensuring diversity in object categories, attributes, scenes, and visual styles. For every sampled configuration, **VisionFoundry** generates a single question with a deterministic answer, plus a corresponding T2I prompt encoding the answer-determining visual facts. After image synthesis and alignment verification, exactly 1k verifier-accepted samples are retained per task, resulting in the balanced **VisionFoundry-10K** dataset of 10k examples. All questions are short and unambiguous, and all answers are concise (e.g., binary, categorical, or short phrases), enabling reliable finetuning and evaluation.

Positioning vs. existing datasets. Unlike natural-image VQA datasets, **VisionFoundry-10K** is not designed to maximize semantic coverage or linguistic diversity. Instead, it prioritizes precise visual grounding and controlled difficulty, making it suitable for diagnosing and improving core perceptual weaknesses of VLMs. Compared with manually curated datasets, **VisionFoundry-10K** is fully synthetic and automatically quality-controlled by verification. As shown in our experiments, training on **VisionFoundry-10K** yields consistent improvements on multiple visual perception benchmarks. This validates the effectiveness of targeted synthetic supervision for visual perception.

4 Experiments

We evaluate the effectiveness of **VisionFoundry**-generated data for improving VLMs' visual perception. Our experiments answer five questions: (i) whether synthetic data generated by **VisionFoundry** improves performance on visual perception benchmarks; (ii) whether such improvements generalize across different VLM backbones and model scales; (iii) whether gains can be achieved without noticeably affecting performance on general-purpose tasks; (iv) how performance changes with data scale and epochs; and (v) whether the same synthetic data remains effective under RL.

4.1 Experimental Setup

Models. We conduct experiments on three representative open-source VLMs covering different scales: Qwen2.5-VL-3B-Instruct (Bai et al., 2025), Llama-3.2-11B-Vision-Instruct (Meta AI, 2024; Dubey et al., 2024), and MiMo-VL-7B-SFT (Yue et al., 2025). For the main experiments, all models are finetuned without LoRA; model-specific settings and hyperparameters are given in Appendix B.1. We additionally evaluate RL post-training on the same data, with settings provided in Appendix B.4.

Training data. The **VisionFoundry-10K** dataset consists of 10k image–question–answer triples generated by **VisionFoundry**, evenly distributed across 10 visual perception tasks (Section 3). For the main experiments, we train on **VisionFoundry-10K** alone for one epoch.

Benchmarks and evaluation protocol. We evaluate models on 13 widely used benchmarks spanning three categories: visual perception, general reasoning, and application-specific tasks, using the VLMEvalKit framework (Duan et al., 2024). MMSI-Bench and LEGO use VLMEvalKit's default circular setting. For 3DSRBench, we report VLMEvalKit's default FlipEval result in circular-evaluation mode. Since our primary focus is visual perception, we evaluate MiMo-VL-7B-SFT in non-thinking mode (detailed settings in Appendix B.2). All reported numbers for the main SFT experiments in Table 1 are averaged over four independent runs.

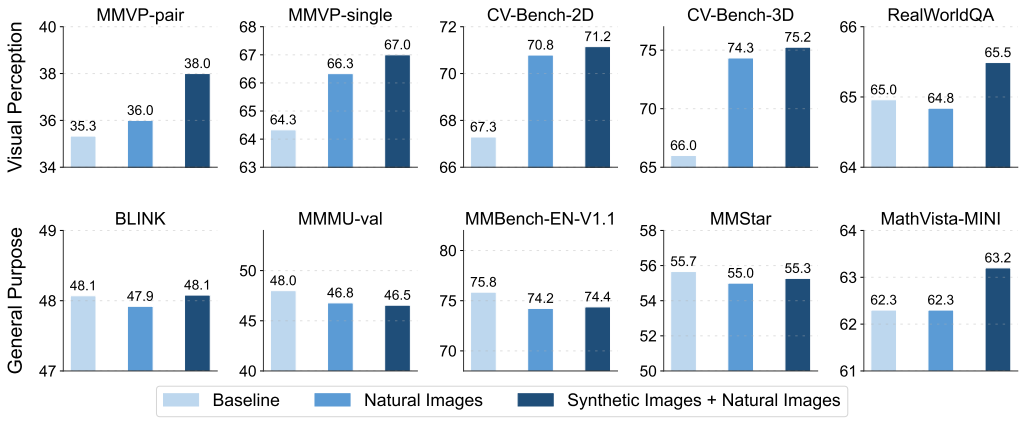

Figure 5: **Equal-sized mixture vs. pure natural data.** We compare training on an equal-sized mix of **VisionFoundry-10K** samples and natural data against training on a size-matched natural-only subset. In the figure, the bold-bordered first row denotes visual perception benchmarks, while the second row reports general-purpose benchmarks. The mixed setting yields consistently higher accuracy on visual perception benchmarks while maintaining comparable performance on general-purpose evaluations.

4.2 Main Results

Table 1 summarizes the main results across all 13 benchmarks. For all VLMs, training with **VisionFoundry** synthetic data consistently improves performance on visual perception benchmarks, while general-purpose benchmarks exhibit more varied, model-dependent responses across tasks.

Visual perception gains. The largest improvements are observed on MMVP, CV-Bench, and RealWorldQA, which explicitly stress spatial reasoning, depth ordering, attribute recognition, and other low-level visual perception skills. These gains validate that **VisionFoundry** effectively targets core visual weaknesses identified in prior work (Tong et al., 2024b;a).

General-purpose tasks. On general-purpose and application-specific benchmarks, outcomes vary across models and benchmarks; for example, Llama improves on 3DSRBench, while MiMo-VL-7B-SFT exhibits particularly pronounced gains on MMBench and MathVista, suggesting that enhanced perceptual grounding provides substantial support on multimodal benchmarks for this backbone. On OCRBench, performance remains largely stable with minor variations, reflecting the domain boundary between low-level visual perception and document text recognition. While perceptual benchmarks show uniform gains, general-purpose metrics exhibit more variation across models, indicating that targeted perception tuning does not compromise overall multimodal reasoning.

Reinforcement learning. We next test whether the utility of VisionFoundry supervision is specific to SFT. Using the same dataset, we apply GRPO to all three backbones. As shown in Table 2, RL improves performance on all five visual perception benchmarks. The gains are particularly large on MMVP and CV-Bench, indicating that the targeted synthetic supervision remains useful under a different post-training objective.

4.3 Data-Size Sensitivity

We analyze how model performance changes with the amount of synthetic data. Instead of varying task types, we vary only the number of synthetic image–question–answer triples. Concretely, we train on subsets of size 0.5k, 1k, 2k, and 5k, as well as the full 10k dataset. All subsets are obtained by random sampling. This lets us compare performance across a consistent sequence of subset sizes, while remaining agnostic about how the trend might vary under other random draws.

Setup. All experiments use Qwen2.5-VL-3B-Instruct. This model offers headroom and reveals data-size effects clearly in the low-data regime. We follow the training setup from the main experiments. We use synthetic-only finetuning under identical hyperparameters described in Appendix B.1.

Results and conclusions. Figure 4 shows consistent gains on visual perception benchmarks as the synthetic data budget increases. At the full 10k setting, we observe sizeable gains over the baseline on key diagnostics: MMVP-pair improves from 35.3 to 42.0 (+6.7), MMVP-single from 64.3 to

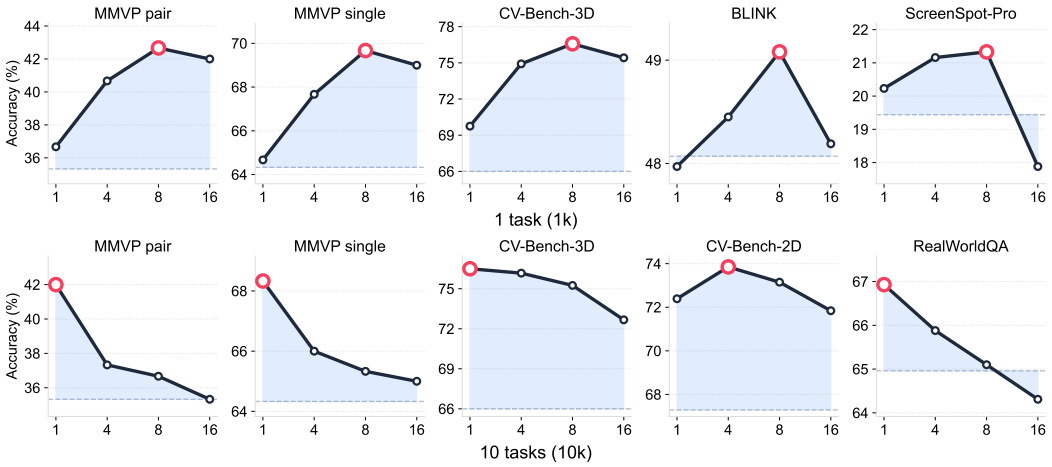

Figure 6: **Epoch trade-off.** The first row uses 1k samples from one randomly selected task, while the second row uses the full **VisionFoundry-10K**. With only a single-task 1k subset, performance typically converges after around 8 epochs. With the full 10-task set, convergence is reached with fewer training epochs. The dashed line indicates the baseline.

68.3 (+4.0), CV-Bench-2D from 67.3 to 72.4 (+5.1), and CV-Bench-3D from 66.0 to 76.5 (+10.5). RealWorldQA also improves from 65.0 to 66.9 (+1.9). A few curves exhibit mild non-monotonicity (e.g., due to subset sampling variance). However, the overall trend is robust: more verifier-filtered synthetic supervision consistently translates into better low-level perception performance. The largest gains appear on 3D/spatial diagnostics (e.g., CV-Bench-3D), suggesting that **VisionFoundry** provides especially effective coverage for geometry- and relation-heavy visual skills.

4.4 Further Analysis

Mix vs. natural. We compare training on an equal-sized mixture of synthetic and natural data against training on pure natural data. Specifically, we construct: (i) a mixed dataset of 2k **VisionFoundry-10K** samples and 2k randomly sampled LLaVA-Instruct-80K samples (4k in total); and (ii) a pure natural dataset of 4k LLaVA-Instruct-80K samples, where the first 2k are the same natural samples used in (i) and kept in the same order, and the remaining 2k are additionally randomly sampled. All experiments use Qwen2.5-VL-3B-Instruct and are trained for one epoch; full hyperparameters are provided in Appendix B.2.

Results (Figure 5) show that the mixed dataset consistently outperforms pure natural data on visual perception benchmarks, while achieving comparable performance on other benchmarks. This suggests that synthetic data provides complementary visual supervision that is difficult to obtain from small natural subsets alone and supports broader synthetic–natural training paradigms.

Training epochs. We analyze the effect of training epochs under two settings. In the first setting, we use 1k samples from one randomly selected task; in the second setting, we use the full **VisionFoundry-10K** (10 tasks). For both settings, we train Qwen2.5-VL-3B-Instruct for 1, 4, 8, and 16 epochs.

Results (Figure 6) show that on the single-task 1k subset, performance improves up to around 8 epochs and saturates thereafter, whereas the full 10-task dataset reaches optimal performance with fewer training epochs. These observations provide practical guidance for choosing training schedules under different task-scale settings.

Task-wise analysis. To understand how different synthetic capabilities transfer across benchmarks, we perform a task-wise dissection of **VisionFoundry-10K** and summarize representative patterns in Figure 7. Following the main training recipe, we train ten task-specific models, each on a 1k subset corresponding to one task, and evaluate all models on the same benchmark suite. Figure 7(a) visualizes five representative tasks for readability, while complete results for all ten task-specific models are reported in Appendix Tables 7 and 8. Detailed experimental settings (identical to the main experiment except task-specific 1k subsets) are provided in Appendix B.2.

The left panel of Figure 7 shows two clear patterns. First, most visual perception benchmarks exhibit consistent improvements across tasks, supporting the effectiveness of task-aware synthetic

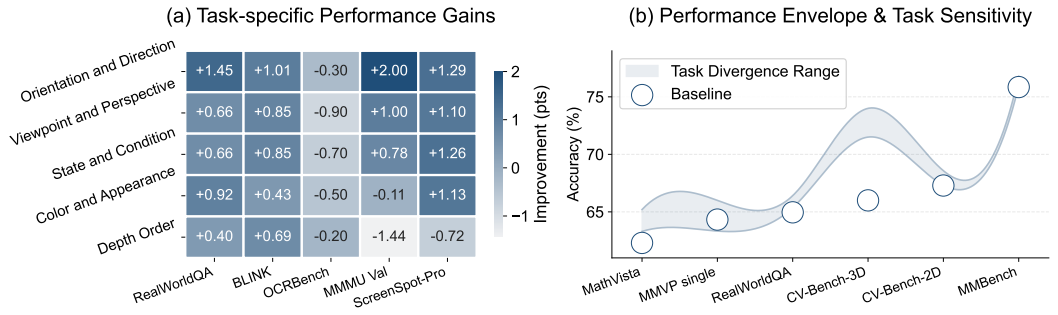

Figure 7: **Task-wise gains and cross-benchmark divergence. (a):** Visualization of five representative task-specific 1k subsets; complete results for all ten task-specific models are reported in Appendix Tables 7 and 8. Darker colors indicate larger gains, while lighter colors indicate smaller gains or regressions. **(b):** Per-benchmark response divergence across tasks, where a larger top-to-bottom accuracy span indicates stronger task sensitivity. While visual perception benchmarks generally benefit from synthetic supervision, the magnitude and direction of gains remain benchmark-dependent.

supervision for visual understanding. Second, OCRBench performance remains essentially stable across tasks, delineating the scope of perception-oriented data from document text understanding.

Importantly, benchmark responses are not uniform across tasks. For example, *Depth Order* provides strong gains on spatially grounded benchmarks. However, it yields limited transfer on ScreenSpot-Pro (a 2D GUI-grounding setting) and MMMU (STEM-heavy multimodal reasoning). This matches the intuition that depth-centric supervision is less aligned with those evaluation demands.

The right panel quantifies cross-task divergence per benchmark. A larger top-to-bottom accuracy span indicates that performance is more sensitive to which synthetic task is used for training. This reveals both task-dependent and benchmark-dependent transfer behavior. Even among visual perception benchmarks, response profiles differ. RealWorldQA shows small divergence across tasks because it places stronger emphasis on reasoning. CV-Bench responds more strongly to structured tasks.

Overall, these analyses indicate that synthetic data can play a targeted improvement role when aligned with specific capabilities and can complement natural data in our current setting. Expanding task diversity may unlock gains across a broader range of benchmarks.

5 Conclusions and Discussion

Limitations and future work. **VisionFoundry-10K** focuses on visual perception tasks, and our study investigates whether targeted synthetic supervision can improve capabilities such as spatial understanding, attribute recognition, and viewpoint reasoning. An open question is whether a synthetic data pipeline such as **VisionFoundry** can extend to more complex visual reasoning tasks that require longer inference chains and stronger compositional reasoning. We leave this direction to future work and hope our findings motivate further study.

Conclusions. We present **VisionFoundry**, a synthetic data generation pipeline that uses an LLM to produce questions, answers, and T2I prompts, synthesizes images with T2I models, and applies multimodal alignment verification, all without reference images or human annotation. Using **VisionFoundry**, we construct **VisionFoundry-10K**, a VQA dataset containing 10k image–question–answer triples spanning 10 visual perception tasks. Experiments across three VLMs show that training on **VisionFoundry-10K** yields consistent improvements on visual perception benchmarks while preserving general multimodal capabilities. The same targeted supervision also remains effective under RL, and a self-verified open-source pipeline continues to improve visual perception.

Our findings indicate that the perception bottleneck in VLMs is partly a data problem and that synthetic data can serve as a complementary source of supervision for strengthening core perception capabilities. Progress in multimodal systems may depend not only on scaling model size or raw data volume, but also on constructing targeted supervision for perceptual distinctions that current models underlearn. An automated synthetic data pipeline offers a practical and scalable building block for more systematic multimodal training.

AI USE STATEMENT

In this work, generative AI tools (LLMs and T2I models) were used as part of the core research methodology to synthesize questions, answers, and images, as described in Section 2. Generative AI was also used to assist with writing refinement and text polishing. All research ideas, experimental designs, algorithmic pipelines, and empirical analyses were developed and validated by the authors. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our work uses synthetic visual question answering data generated by LLMs and T2I models. The generation prompts do not target sensitive personal data or harmful content, and the dataset is intended for research on visual perception in vision-language models. No human subjects were involved in the generation or annotation process.

REPRODUCIBILITY STATEMENT

To ensure reproducibility, Section 2 and Section 4 detail the VisionFoundry data synthesis pipeline, model architectures, hyperparameters, training setups, and evaluation protocols. In addition, Appendix B provides full implementation details, and Appendix C presents the exact prompts used for task generation, question–answer synthesis, and verification. An anonymous codebase containing complete scripts and pipeline configurations is available at https://anonymous.4open.science/r/VisionFoundry_Anonymous/README.md.

REFERENCES

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves ImageNet classification. *TMLR*, 2023.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Gabriela Ben-Melech Stan, Estelle Aflalo, Avinash Madasu, Vasudev Lal, and Phillip Howard. Learning from reasoning failures via synthetic data generation. In *AAAI*, 2025.

Garrick Brazil, Abhinav Kumar, Julian Straub, Nikhila Ravi, Justin Johnson, and Georgia Gkioxari. Omni3D: A large benchmark and model for 3d object detection in the wild. In *CVPR*, 2023.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Danny Driess, Pete Florence, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. SpatialVLM: Endowing vision-language models with spatial reasoning capabilities. In *CVPR*, 2024a.

Guiming Hardy Chen, Shunian Chen, Ruifei Zhang, Junying Chen, Xiangbo Wu, Zhiyi Zhang, Zhihong Chen, Jianquan Li, Xiang Wan, and Benyou Wang. ALLaVA: Harnessing GPT4V-synthesized data for lite vision-language models. *arXiv preprint arXiv:2402.11684*, 2024b.

Kai Chen, Enze Xie, Zhe Chen, Yibo Wang, Lanqing Hong, Zhenguo Li, and Dit-Yan Yeung. Geodiffusion: Text-prompted geometric control for object detection data generation. In *ICLR*, 2024c.

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 *ECCV*, 2024d.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models? In *NeurIPS*, 2024e.

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

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *NeurIPS*, 2023.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, Dahua Lin, and Kai Chen. VLMEvalKit: An open-source toolkit for evaluating large multi-modality models. In *MM*, 2024.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akash Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Lisa Dunlap, Alyssa Umino, Han Zhang, Jiezhi Yang, Joseph E Gonzalez, and Trevor Darrell. Diversify your vision datasets with automatic diffusion-based augmentation. In *NeurIPS*, 2023.

Lijie Fan, Kaifeng Chen, Dilip Krishnan, Dina Katabi, Phillip Isola, and Yonglong Tian. Scaling laws of synthetic images for model training ... for now. In *CVPR*, 2024.

Yunhao Fang, Ligeng Zhu, Yao Lu, Yan Wang, Pavlo Molchanov, Jan Kautz, Jang Hyun Cho, Marco Pavone, Song Han, and Hongxu Yin. VILA^2: VILA augmented VILA. In *CVPR*, 2025.

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

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: Multimodal large language models can see but not perceive. In *ECCV*, 2024b.

Google Cloud. Gemini 2.5 Flash Image model documentation. https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-flash-image, 2025a.

Google Cloud. Gemini 3 Pro model documentation. https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/3-pro, 2025b.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip H. S. Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? In *ICLR*, 2023.

Zizhao Hu, Mohammad Rostami, and Jesse Thomason. Multimodal synthetic data finetuning and model collapse: Insights from VLMs and diffusion models. In *ICMI*, 2025.

Qirui Jiao, Daoyuan Chen, Yilun Huang, Bolin Ding, Yaliang Li, and Ying Shen. Img-Diff: Contrastive data synthesis for multimodal large language models. In *CVPR*, 2025.

Haoxin Li and Boyang Li. Enhancing vision-language compositional understanding with multimodal synthetic data. In *CVPR*, 2025.

Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Ma, Zhiyong Huang, and Tat-Seng Chua. ScreenSpot-Pro: GUI grounding for professional high-resolution computer use. In *MM*, 2025.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common objects in context. In *ECCV*, 2014.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. MMBench: Is your multi-modal model an all-around player? In *ECCV*, 2024a.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xucheng Yin, Cheng-lin Liu, Lianwen Jin, and Xiang Bai. OCRBench: On the hidden mystery of OCR in large multimodal models. *SCIS*, 2024b.

Zheng Liu, Hao Liang, Bozhou Li, Wentao Xiong, Chong Chen, Conghui He, Wentao Zhang, and Bin Cui. SynthVLM: Towards high-quality and efficient synthesis of image-caption datasets for vision-language models. In *MM*, 2025.

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. In *ICLR*, 2024.

Run Luo, Haonan Zhang, Longze Chen, Ting-En Lin, Xiong Liu, Yuchuan Wu, Min Yang, Yongbin Li, Minzheng Wang, Pengpeng Zeng, Lianli Gao, Heng Tao Shen, Yunshui Li, Hamid Alinejad-Rokny, Xiaobo Xia, Jingkuan Song, and Fei Huang. MMEvol: Empowering multimodal large language models with evol-instruct. In *ACL*, 2025.

Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Jieneng Chen, Celso de Melo, and Alan Yuille. 3DSRBench: A comprehensive 3d spatial reasoning benchmark. In *ICCV*, 2025.

Meta AI. Llama 3.2 vision: Multimodal large language models. https://huggingface.co/meta-llama/Llama-3.2-11B-Vision-Instruct, 2024.

Quang Nguyen, Truong Vu, Anh Tran, and Khoi Nguyen. Dataset diffusion: Diffusion-based synthetic dataset generation for pixel-level semantic segmentation. In *NeurIPS*, 2023.

Minheng Ni, Zitong Huang, Kailai Feng, and Wangmeng Zuo. Imaginarynet: Learning object detectors without real images and annotations. In *ICLR*, 2023.

OpenAI. Introducing GPT-5.2. https://openai.com/index/introducing-gpt-5-2/, 2025.

Zeyu Qin, Qingxiu Dong, Xingxing Zhang, Dong Li, Xiaolong Huang, Ziyi Yang, Mahmoud Khademi, Dongdong Zhang, Hany Hassan Awadalla, Yi R. Fung, Weizhu Chen, Minhao Cheng, and Furu Wei. Scaling laws of synthetic data for language models. In *Conference on Language Modeling*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Sahand Sharifzadeh, Christos Kaplanis, Shreya Pathak, Dharshan Kumaran, Anastasija Ilic, Jovana Mitrovic, Charles Blundell, and Andrea Banino. Synth^2: Boosting visual-language models with synthetic captions and image embeddings. *arXiv preprint arXiv:2403.07750*, 2024.

Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See-Kiong Ng, Lidong Bing, and Roy Ka-Wei Lee. Math-LLaVA: Bootstrapping mathematical reasoning for multimodal large language models. In *EMNLP*, 2024.

Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion. In *CVPR*, 2023.

Krishnakant Singh, Thanush Navaratnam, Jannik Holmer, Simone Schaub-Meyer, and Stefan Roth. Is synthetic data all we need? benchmarking the robustness of models trained with synthetic images. In *CVPR*, 2024.

Kexian Tang, Junyao Gao, Yanhong Zeng, Haodong Duan, Yanan Sun, Zhening Xing, Wenran Liu, Kaifeng Lyu, and Kai Chen. LEGO-Puzzles: How good are MLLMs at multi-step spatial reasoning? In *CVPR*, 2025.

Qwen Team. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.

Yonglong Tian, Lijie Fan, Phillip Isola, Huiwen Chang, and Dilip Krishnan. Stablerep: Synthetic images from text-to-image models make strong visual representation learners. In *NeurIPS*, 2023.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Austin Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal LLMs. In *NeurIPS*, 2024a.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal LLMs. In *CVPR*, 2024b.

Bin Wu, Wuxuan Shi, Jinqiao Wang, and Mang Ye. Synthetic data is an elegant GIFT for continual vision-language models. In *CVPR*, 2025a.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025b.

Weijia Wu, Yuzhong Zhao, Hao Chen, Yuchao Gu, Rui Zhao, Yefei He, Hong Zhou, Mike Zheng Shou, and Chunhua Shen. DatasetDM: Synthesizing data with perception annotations using diffusion models. In *NeurIPS*, 2023a.

Weijia Wu, Yuzhong Zhao, Mike Zheng Shou, Hong Zhou, and Chunhua Shen. Diffumask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models. In *ICCV*, 2023b.

Xindi Wu, Hee Seung Hwang, Polina Kirichenko, Esin Tureci, and Olga Russakovsky. COMPACT: COMPositional atomic-to-complex visual capability tuning. In *ICLR*, 2025c.

xAI. Realworldqa dataset (xai). https://huggingface.co/datasets/xai-org/RealworldQA, 2024.

Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, Dahua Lin, Tai Wang, and Jiangmiao Pang. MMSI-Bench: A benchmark for multi-image spatial intelligence. In *International Conference on Learning Representations*, 2026.

Yuwei Yang, Zeyu Zhang, Yunzhong Hou, Zhuowan Li, Gaowen Liu, Ali Payani, Yuan-Sen Ting, and Liang Zheng. Effective training data synthesis for improving MLLM chart understanding. In *ICCV*, 2025.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. In *ICML*, 2024.

Jianhao Yuan, Francesco Pinto, Adam Davies, and Philip H. S. Torr. Not just pretty pictures: Toward interventional data augmentation using text-to-image generators. In *ICML*, 2024.

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, Huan Sun, Yu Su, and Wenhu Chen. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *CVPR*, 2024.

Zihao Yue, Zhenru Lin, Yifan Song, Weikun Wang, Shuhuai Ren, et al. MiMo-VL technical report. *arXiv preprint arXiv:2506.03569*, 2025.

Jieyu Zhang, Le Xue, Linxin Song, Jun Wang, Weikai Huang, Manli Shu, An Yan, Zixian Ma, Juan Carlos Niebles, Silvio Savarese, Caiming Xiong, Zeyuan Chen, Ranjay Krishna, and Ran Xu. ProVision: Programmatically scaling vision-centric instruction data for multimodal language models. In *ICLR*, 2025.

Bo Zhao, Boya Wu, Muyang He, and Tiejun Huang. SVIT: Scaling up visual instruction tuning. *arXiv preprint arXiv:2307.04087*, 2023.

Henry Hengyuan Zhao, Pan Zhou, Difei Gao, Zechen Bai, and Mike Zheng Shou. LOVA3: Learning to visual question answering, asking and assessment. In *NeurIPS*, 2024.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through the ADE20K dataset. In *CVPR*, 2017.

Yongchao Zhou, Hshmat Sahak, and Jimmy Ba. Training on thin air: Improve image classification with generated data. *arXiv preprint arXiv:2305.15316*, 2023.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *ICLR*, 2024.

Appendix

This appendix collects related work, implementation details, prompt specifications, verifier case studies, data-quality analyses, task-wise results, and controlled ablations that support the main paper.

A Related Work **16**

B Implementation Details **17**

B.1 Optimization and Hyperparameters 17

B.2 Evaluation Protocol 17

B.3 Supplementary Protocols 17

B.4 RL Settings 18

C Exact Pipeline Prompts **18**

C.1 Prompt Set A 18

C.2 Prompt Set B 20

D Verifier Cases **21**

D.1 True Positive Cases 21

D.2 False Negative Cases 22

D.3 False Positive Cases 22

D.4 True Negative Cases 22

E Data Quality Analysis **23**

E.1 Verification-Accuracy Records 23

F Task-wise Results **24**

F.1 Task-wise Experimental Settings 24

F.2 Complete Per-task Results 24

G Ablation on Verification Helpfulness **25**

H Synthetic Process Ablations **25**

I T2I Model Ablations **27**

J Ablation on Verifier Choice **27**

A Related Work

Synthetic data for VLMs. Recent work increasingly uses synthetic multimodal data to scale vision–language learning and instruction tuning. For caption generation, methods generate high-quality captions at scale: ShareGPT4V produces 1.2M captions (Chen et al., 2024d), ALLaVA synthesizes 3.4M captioning and reasoning QA pairs (Chen et al., 2024b), SynthVLM refines caption quality (Liu et al., 2025), and VILA^2 enables self-recaptioning (Fang et al., 2025). For instruction tuning, SVIT scales to 4.2M instructions (Zhao et al., 2023), ProVision generates 10M+ vision-centric instructions (Zhang et al., 2025), MMEvol evolves 447K instructions (Luo et al., 2025), LOVA3 teaches VQA generation (Zhao et al., 2024), and Img-Diff creates contrastive pairs (Jiao et al., 2025). Domain-specific methods include SpatialVLM for 3D reasoning (Chen et al., 2024a), Math-LLaVA for mathematical reasoning (Shi et al., 2024), and chart QA pipelines (Yang et al., 2025). Other work explores compositional generalization (Li & Li, 2025), failure-driven synthesis (Ben-Melech Stan et al., 2025; Wu et al., 2025c), and robustness analysis (Hu et al., 2025). VisionFoundry differs from prior work in being a fully automated pipeline that generates datasets from task keywords alone, while many related methods operate at a different scope and are therefore not directly comparable.

T2I-generated training data. T2I-generated training data has evolved from representation learning to supervised recognition. For representation learning, StableRep applies diffusion images to contrastive learning (Tian et al., 2023), Synth^2 uses T2I embeddings for VLM training (Sharifzadeh et al., 2024), and scaling laws show CLIP pretraining behavior (Fan et al., 2024). For classification, work examines synthetic-only versus mixed training (He et al., 2023), filtered diffusion samples (Azizi et al., 2023), low-data regimes (Zhou et al., 2023), robustness interventions (Shipard et al., 2023; Yuan et al., 2024), and generation-selection loops (Dunlap et al., 2023). For dense prediction, methods generate structured annotations (Wu et al., 2023a), segmentation masks (Nguyen et al., 2023; Wu et al., 2023b), and detection data with geometric control (Ni et al., 2023; Chen et al., 2024c). Beyond recognition, diffusion samples mitigate catastrophic forgetting (Wu et al., 2025a), while benchmarks support hybrid synthetic+natural recipes (Singh et al., 2024).

VLM training and evaluation. Progress in visual instruction tuning relies on scalable supervision and evaluation. Early pipelines, including LLaVA (Liu et al., 2023), InstructBLIP (Dai et al., 2023), MiniGPT-4 (Zhu et al., 2024), and Qwen-VL (Team, 2023), demonstrate that converting image–text corpora into instruction-following data aligns perception and generation in VLMs. Visual perception benchmarks repurpose CV resources: CV-Bench (Tong et al., 2024a) uses Microsoft COCO (Lin et al., 2014), ADE20K (Zhou et al., 2017), and Omni3D (Brazil et al., 2023) for 2D/3D perception, while RealWorldQA (xAI, 2024) targets real-scene spatial understanding. Complementary benchmarks cover visual pitfalls such as MMVP (Tong et al., 2024b), perception limitations such as BLINK (Fu et al., 2024b), expert reasoning benchmarks such as MMMU (Yue et al., 2024), and integrated-competence benchmarks such as MM-Vet (Yu et al., 2024) and MME (Fu et al., 2024a). Multi-domain benchmarks include MMBench (Liu et al., 2024a), MMStar (Chen et al., 2024e), MathVista (Lu et al., 2024), ScreenSpot-Pro (Li et al., 2025), and OCRBench (Liu et al., 2024b). These resources help diagnose limitations and motivate targeted synthesis strategies.

B Implementation Details

B.1 Optimization and Hyperparameters

Table 3 summarizes the model-specific hyperparameters used in the main experiments. We report settings for Qwen2.5-VL-3B-Instruct (Bai et al., 2025), MiMo-VL-7B-SFT (Yue et al., 2025), and Llama-3.2-11B-Vision-Instruct (Meta AI, 2024; Dubey et al., 2024). In particular, η_{ViT}, η_{adapter}, and η_{LLM} denote the learning rates for the ViT encoder, adapter modules, and LLM backbone, respectively. All main-experiment runs use a global batch size of 128 and are trained for one epoch.

Model	Epochs	Batch	η_{ViT}	η_{adapter}	η_{LLM}	LLM
Qwen2.5-VL-3B-Instruct	1	128	5×10^{-7}	5×10^{-6}	5×10^{-6}	Unfrozen
MiMo-VL-7B-SFT	1	128	5×10^{-7}	2.5×10^{-6}	2.5×10^{-6}	Unfrozen
Llama-3.2-11B-Vision-Instruct	1	128	5×10^{-7}	5×10^{-6}	N/A	Frozen

Table 3: **Main-experiment hyperparameters.** Training settings for Qwen2.5-VL-3B-Instruct, MiMo-VL-7B-SFT, and Llama-3.2-11B-Vision-Instruct in the primary comparison (all runs use 1 epoch and global batch size 128). η_{ViT}, η_{adapter}, and η_{LLM} denote learning rates for the vision encoder, adapter modules, and LLM backbone, respectively; **LLM** indicates whether the LLM backbone is frozen or unfrozen.

For MiMo-VL-7B-SFT (Yue et al., 2025), we target the non-thinking setting during both training and evaluation. During training, we prepend `<think></think>` to each answer as a placeholder to indicate disabled thinking. During evaluation, we append `/no_think` to the end of the input, following the official repository protocol, to measure visual perception gains more directly, rather than relying on indirect gains from extended reasoning[1]. For Llama-3.2-11B-Vision-Instruct (Meta AI, 2024; Dubey et al., 2024), we freeze the LLM backbone and only unfreeze ViT and adapters to mitigate catastrophic forgetting caused by full-parameter finetuning.

B.2 Evaluation Protocol

All evaluations are conducted using the open-source VLMEvalKit framework (Duan et al., 2024) with default settings unless explicitly noted below. For Qwen2.5-VL-3B-Instruct, we keep VLMEvalKit defaults and do not impose explicit minimum or maximum pixel overrides.

Benchmark suite. We evaluate models on 13 benchmarks in three categories. Visual perception benchmarks include MMVP (Tong et al., 2024b), CV-Bench (Tong et al., 2024a), and RealWorldQA (xAI, 2024). General reasoning benchmarks include BLINK (Fu et al., 2024b), MMMU-Val (Yue et al., 2024), MMBench (EN v1.1) (Liu et al., 2024a), MMStar (Chen et al., 2024e), and MathVista-Mini (Lu et al., 2024). Application-specific benchmarks include OCRBench (Liu et al., 2024b), ScreenSpot-Pro (Li et al., 2025), MMSI-Bench (circular) (Yang et al., 2026), 3DSRBench (Ma et al., 2025), and LEGO (circular) (Tang et al., 2025).

Benchmark-specific settings. For MMVP, we report both the official pair accuracy and the commonly used single-image accuracy for reference. For MMMU, we evaluate on the `val` split only and exclude the `dev` split. For ScreenSpot-Pro, we report the average of overall scores across all subtasks. For LEGO and MMSI-Bench, we use the default circular evaluation mode in VLMEvalKit. For 3DSRBench, we use VLMEvalKit's default FlipEval result in circular-evaluation mode (the most challenging mode). All reported numbers are averaged over four independent runs.

B.3 Supplementary Protocols

Data-size sensitivity. We reuse exactly the same model hyperparameters as the main experiments. Subsets are built by random sampling from the full dataset at sizes 0.5k, 1k, 2k, and 5k. For each subset size, results are averaged over four independent runs; each run is trained for one epoch, and evaluation settings are identical to the main experiments.

[1] https://github.com/XiaomiMiMo/MiMo-VL

Synthetic vs. natural data. We keep model hyperparameters unchanged from the main experiments. The synthetic set is a random subset sampled from **VisionFoundry-10K**, and the natural set is a random subset sampled from LLaVA-Instruct-80K (Liu et al., 2023), with matched subset sizes. Results are averaged over four independent runs, each trained for one epoch with the same evaluation protocol as the main experiments.

Training epochs and task-wise settings. For the training-epoch ablation, we use a randomly selected 1k subset and vary only the number of epochs, while keeping training and evaluation settings identical to the main experiments. For task-wise experiments, we train one model per task-specific dataset for one epoch under the same training and evaluation settings as the main experiments.

B.4 RL Settings

For all RL experiments, we use Group Relative Policy Optimization (GRPO) (Shao et al., 2024) with four rollouts per prompt and a KL divergence penalty coefficient of 0.01. We use the same 10-task 10k synthetic dataset (**VisionFoundry-10K**) as in the main experiments and train for one epoch. Model-specific learning rates, batch sizes, and trainable modules are summarized in Table 4. The reward model (Qwen2.5-3B) serves as a language judge to evaluate rollout responses against the ground-truth answers, providing a binary correctness reward.

Model	Trainable modules	Reward model	Epoch	LR	Batch	Rollouts	Details
Qwen2.5-VL-3B-Instruct	Full model	Qwen2.5-3B	1	4×10^{-6}	128	4	GRPO, step 70
Llama-3.2-11B-Vision-Instruct	Vision + adapters	Qwen2.5-3B	1	1×10^{-6}	32	4	GRPO, step 281
MiMo-VL-7B-SFT	Full model	Qwen2.5-3B	1	1×10^{-6}	64	4	GRPO, step 140

Table 4: **RL hyperparameters.** Experimental configurations for GRPO post-training on **VisionFoundry-10K**. All runs use a KL coefficient of 0.01 and four rollouts per prompt.

Across vision benchmarks, RL post-training consistently maintains or improves model performance across backbones, confirming that **VisionFoundry** supervision transfers effectively beyond SFT.

C Exact Pipeline Prompts

We report the exact prompt templates used in **VisionFoundry** for generation, verification, and feedback. Fields in angle brackets (e.g., `<TASK_DESCRIPTION>`) denote runtime substitutions from `TaskConfig` or states.

C.1 Prompt Set A

A1. Custom attribute expansion (e.g., spatial relations, difficulty labels).

System prompt:

```
Output ONLY valid JSON array of strings.
```

User prompt:

```
For the task '<TASK_DESCRIPTION>', generate <COUNT> diverse options
for '<ATTR_NAME>'.
These should be specific, varied, and visually verifiable if
possible.
Return ONLY a valid JSON array of strings.
```

This prompt expands task attributes into a controlled candidate pool that is later used to instantiate concrete samples. The JSON-only requirement keeps outputs machine-readable, while the diversity and visual-verifiability constraints encourage attributes that are both varied and practically checkable during downstream generation and verification.

A2. Single-image QA + T2I case construction.

System prompt template:

```
You are an expert VQA data creator for the task: "<TASK_ID>"
Task description: <TASK_DESCRIPTION>

<IF NUM_OBJECTS > 1>
This task requires <NUM_OBJECTS> objects in the image with clear
relationships.

<IF MODE == "multi">
MULTI-IMAGE MODE: Generate <NUM_IMAGES> interconnected images with a
coherent story.

<IF "spatial" IN TASK_ID OR CONSTRAINTS>
SPATIAL FOCUS: The question MUST involve spatial relationships
between objects.

<IF "color" IN TASK_ID OR CONSTRAINTS>
COLOR FOCUS: Object colors must be distinct, accurate, and central to
the question.

CRITICAL RULES:
1. The answer-determining fact MUST be 100% visually verifiable from
the final image.
2. Text prompt must explicitly describe content matching the correct
answer.
3. Never rely on invisible properties.
4. Generate deterministic, unambiguous questions.

Constraints: <COMMA_JOINED_CONSTRAINTS_OR_NONE>
Return EXACTLY ONE JSON object with keys:
- "prompt": extremely detailed text-to-image prompt (English)
- "question": clear VQA question
- "answer": clear deterministic answer
- "metadata": {"difficulty": "easy", "category": "<TASK_ID>",
"num_objects": <NUM_OBJECTS>}
```

This prompt is the main sample-construction template: it jointly produces the image prompt, question, and answer in one structured record. Conditional blocks (e.g., spatial/color focus and object-count constraints) adapt instructions to task requirements, and the required rules explicitly enforce visual grounding and determinism so that each generated case remains auditable and suitable for training.

C.2 PROMPT SET B

B1. Question–answer to declarative verification statement.

System prompt:

```
Rewrite question+answer into ONE concise sentence. Output only the
sentence.
```

User prompt:

```
Q: <QUESTION>
A: <ANSWER>
```

This prompt standardizes each QA pair into a single declarative statement that can be checked directly against visual evidence. By removing interrogative wording and enforcing one concise sentence, it reduces ambiguity for downstream verification and keeps the judged content aligned with the original answer target.

B2. Image consistency verification (binary judge).

System prompt:

```
You are an image verification assistant. Decide if an image matches
the statement overall,
ignoring minor details. ANSWER WITH EXACTLY ONE LINE: 'Answer: YES'
or 'Answer: NO'
```

User payload text field:

```
Statement: '<STATEMENT>'

Does this image match?
```

This prompt performs the core filtering decision: whether the generated image supports the target statement at an overall semantic level. The instruction to ignore minor details makes the check robust to inconsequential rendering noise, while the strict one-line `YES/NO` format enables deterministic parsing in the data pipeline. The verifier receives the image in the same request (URL or base64 data URI), and only `Answer: YES` counts.

B3. Optional edit-based regeneration after failed verification.

Initial image-generation instruction:

```
generate an image: <PROMPT>
```

Edit instruction (used when `use_edit=True`):

```
Make the image exactly match: '<STATEMENT>'. Only modify what's
needed.
```

This prompt defines the fallback correction stage for cases that fail verification. The first instruction generates an image from the original prompt, and the edit instruction then applies minimal, targeted changes to satisfy the verification statement without unnecessary drift. Operationally, each sample uses outer-loop retries up to the configured maximum; each verification call has retry handling; and retention requires approval.

D VERIFIER CASES

We provide qualitative case studies to interpret how synthetic samples are handled by the verifier. Following a manual audit, we classify each sample along two dimensions: (i) whether the generation is correct (correct/wrong), and (ii) whether the verifier's judgment is correct. We then report the filter's final decision (accept/reject): *true positive* (correct and accepted), *false negative* (correct but rejected), *false positive* (wrong but accepted), and *true negative* (wrong and rejected).

D.1 TRUE POSITIVE CASES

We present representative examples where generation is correct and the verifier accepts the sample.

Case 1: surface glossiness judgment



Question: Does the snowbank in the foreground appear glossy (wet and reflective)?

Answer: Yes, it appears glossy.

Case 2: relative depth comparison



Question: In the scene, which object is closer to the camera: the dumpster or the candle?

Answer: The dumpster is closer to the camera.

Case 3: high-viewpoint recognition



Question: Is the photo taken from above the statue looking downward, as if from a balcony-like high viewpoint?

Answer: Yes, it is taken from above looking downward.

D.2 False Negative Cases

We present a false-negative case in which generation is correct but the verifier rejects the sample.

Case 4



Statement: The forklift is facing toward the camera.

Reason: The image matches the statement (the forklift indeed faces toward the camera), but the VLM verifier incorrectly judges it as wrong, leading to a false reject.

D.3 False Positive Cases

We present a false-positive case in which generation is wrong but the verifier accepts the sample.

Case 5



Statement: The sailboat's bow is pointing left.

Reason: The generated image is incorrect: the sailboat's bow points right instead of left. However, the VLM verifier mistakenly accepts this sample, resulting in a false accept.

D.4 True Negative Cases

We present a true-negative case in which generation is wrong and the verifier rejects the sample.

Case 6



Statement: The shopping cart is facing toward the lower-left corner of the scene.

Reason: The generation is wrong: it should face toward the lower-left corner but is rendered toward the lower-right. The VLM verifier correctly identifies this mismatch and rejects the sample.

E Data Quality Analysis

To quantify the reliability of generated supervision, we report the acceptance dynamics of our filtering pipeline and manual agreement statistics. This section analyzes one concrete random-sampling run in which we track the full generation trajectory to obtain the final valid subset. These diagnostics show whether high-quality supervision emerges systematically rather than by chance.

E.1 Verification-Accuracy Records

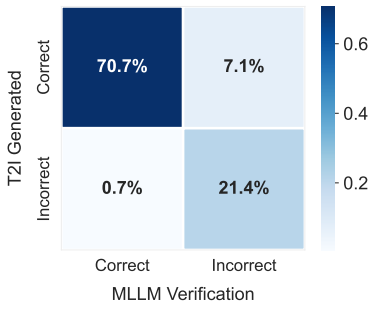

Figure 8: **Confusion matrix for the random-sampling audit.** Rows are manual correctness labels, and columns are verifier outputs. Values summarize the audited trajectory.

We summarize the audited trajectory with a confusion matrix. We set a target of retaining 100 samples and therefore audited one full production trajectory with 140 generation attempts, in which the verifier filtered out 40 candidates. Here, a valid case is defined as *generated correct & filter pass* under manual audit. To avoid survivorship bias, we audit the full trajectory rather than only the retained outputs. The resulting outcome shares are: generated correct & filter pass = 70.7%, generated correct & filter fail = 7.1%, generated wrong & filter pass = 0.7%, and generated wrong & filter fail = 21.4%. In absolute counts, this corresponds to 140 audited candidates with 100 retained after verification: 99 valid samples (generated correct & filter pass) and 1 false accept (generated wrong & filter pass); the remaining 40 are rejected (10 generated correct & filter fail and 30 generated wrong & filter fail). As visualized in Figure 8, most mass lies on the diagonal, with *correct & pass* and *wrong & fail* as the dominant outcomes. This indicates that the verifier is directionally aligned with manual judgments on this audited run, effectively capturing the dominant correctness patterns in the generation stream. Using generation correctness as ground truth and filter pass/fail as predictions, the filter output rates are 71.4% pass and 28.6% fail, and the filter-pass set is highly reliable (99.0% correct).

Split	Accuracy	Precision	Recall	F1
Overall audited set	92.1	99.0	90.8	94.7

Table 5: **Verifier performance on the audited sample set.** Accuracy, precision, recall, and F1 are computed by comparing verifier pass/fail outputs against manual labels over all audited candidates.

Table 5 shows high precision (99.0%) and strong F1 (94.7%), suggesting that accepted samples are typically correct under manual audit. Recall (90.8%) is lower than precision, indicating that some correct samples are filtered out. Overall, the verifier behaves as a high-precision filter in this trajectory. The audit reflects the automated filtering distribution without manual post-processing, characterizing the raw data quality yielded by the verifier.

Metric	Definition	Value
Final valid proportion	generated correct & filter pass over all audited candidates	70.7%
Acceptance rate	retained / generated candidates	71.4%
Rejection rate	rejected / generated candidates	28.6%
False accept rate	generated wrong but filter pass / generated wrong candidates	3.2%
False reject rate	generated correct but filter fail / generated correct candidates	9.2%
Human–judge agreement	Cohen's κ on audited subset	0.794

Table 6: **Pipeline quality metrics on the audited trajectory.** The table reports final valid proportion, acceptance/rejection rates, false accept/reject rates, and human–judge agreement (Cohen's κ).

Table 6 provides an operational view: the pipeline retains 71.4% of candidates, which in this audit is exactly 100/140 retained (99 valid and 1 false accept) and 40/140 rejected. This is equivalent to a final valid proportion of 70.7% and low false accepts (3.2%). False rejects remain non-zero (9.2%), showing a measurable quality–coverage trade-off. The observed Cohen's κ of 0.794 is consistent with substantial agreement, but this estimate is based on one audited trajectory and should be interpreted as indicative rather than universal.

F Task-wise Results

This section reports full task-wise experimental settings and complete per-task benchmark results.

F.1 Task-wise Experimental Settings

Each task-specific dataset is trained for one epoch. Training and evaluation settings are identical to those used in the main experiments (Appendix B.1 and B.2). This ensures that performance differences are attributable to task specialization rather than variations in evaluation protocols.

F.2 Complete Per-task Results

Table 7 and Table 8 report the full benchmark results of all ten task-specific models, all based on Qwen2.5-VL-3B-Instruct. To keep the tables compact, we use task abbreviations: **OaD** (*Orientation and Direction*), **VaP** (*Viewpoint and Perspective*), **PaRC** (*Positional and Relational Context*), **SR** (*Spatial Relationship*), **SaC** (*State and Condition*), **SPC** (*Structural and Physical Characteristics*), **CaA** (*Color and Appearance*), **DO** (*Depth Order*), **RD** (*Relative Distance*), and **RWSU** (*Real-World Spatial Understanding*). Benchmark abbreviations are: **MVP-S** (MMVP single-image), **MVP-P** (MMVP pair), **CV2D** (CV-Bench-2D), **CV3D** (CV-Bench-3D), **RWQA** (RealWorldQA), **BLK** (BLINK), **MMS** (MMStar), **OCR** (OCRBench), **MMMU** (MMMU-Val), **MMB** (MMBench), **MVis** (MathVista), **SSP** (ScreenSpot-Pro), **MMSI** (MMSI-Bench), and **3DSR** (3DSRBench). Corresponding benchmark references are MMVP (Tong et al., 2024b), CV-Bench (Tong et al., 2024a), RealWorldQA (xAI, 2024), BLINK (Fu et al., 2024b), MMStar (Chen et al., 2024e), OCRBench (Liu et al., 2024b), MMMU (Yue et al., 2024), MMBench (Liu et al., 2024a), MathVista (Lu et al., 2024), ScreenSpot-Pro (Li et al., 2025), MMSI-Bench (Yang et al., 2026), 3DSRBench (Ma et al., 2025), and LEGO (Tang et al., 2025).

Task	MVP-S	MVP-P	CV2D	CV3D	RWQA	BLK	MMS	OCR
OaD	64.0	36.0	68.4	73.9	66.4	49.1	55.8	82.4
VaP	65.0	36.0	68.4	73.4	65.6	48.9	55.8	81.8
PaRC	66.0	38.0	67.9	73.5	66.1	49.5	55.4	82.2
SR	64.3	36.0	67.9	72.7	66.1	49.0	55.3	82.3
SaC	64.7	36.7	68.3	71.5	65.6	48.9	55.4	82.0
SPC	65.0	36.0	67.9	73.7	66.0	48.5	55.7	82.4
CaA	64.7	36.7	68.6	73.7	65.9	48.5	56.1	82.2
DO	64.3	35.3	67.7	72.4	65.4	48.8	55.5	82.5
RD	63.3	34.7	67.5	74.0	66.3	48.9	55.9	82.3
RWSU	64.3	36.0	67.3	73.0	65.6	48.9	55.5	82.2

Table 7: **Task-wise benchmark performance across 10 task-specialized models (Part I).** Scores (%) on MVP-S, MVP-P, CV2D, CV3D, RWQA, BLK, MMS, and OCR; higher is better.

Task	LEGO	MMMU	MMB	MVis	SSP	MMSI	3DSR
OaD	15.7	50.0	76.2	64.2	20.7	8.8	35.2
VaP	16.3	49.0	75.9	65.2	20.5	8.7	35.2
PaRC	16.4	48.8	76.8	64.6	20.6	9.4	35.3
SR	16.0	48.1	76.2	63.3	20.2	9.4	34.9
SaC	16.1	48.8	76.1	64.4	20.7	9.7	35.2
SPC	15.8	48.4	76.3	63.9	20.4	9.5	35.3
CaA	15.8	47.9	76.0	64.9	20.6	8.9	35.4
DO	15.8	46.6	76.2	63.6	18.7	9.1	35.3
RD	15.9	48.9	76.3	65.2	19.9	9.6	35.6
RWSU	16.0	48.2	76.2	64.2	20.6	9.1	35.2

Table 8: **Task-wise benchmark performance across 10 task-specialized models (Part II).** Scores (%) on LEGO, MMMU, MMB, MVis, SSP, MMSI, and 3DSR; higher is better.

G Ablation on Verification Helpfulness

To isolate the contribution of the verification stage in **VisionFoundry**, we conduct a controlled ablation on a single task with a fixed training size of 1k samples.

Goal and protocol. This ablation tests whether verification helps under a matched 1k-sample training budget. We use Qwen2.5-VL-3B-Instruct as the backbone and evaluate three settings: (i) the baseline, (ii) finetuning with verification, and (iii) finetuning without verification. For both finetuned models, we keep the training recipe identical: unfreeze adapters, ViT, and LLM; use learning rates of 5×10^{-6} for adapter/LLM and 5×10^{-7} for ViT; train for one epoch with batch size 128. The training task is *Orientation and Direction*. Both finetuned models use 1k training samples: the verified branch keeps 1k verifier-accepted samples, while the non-verified branch uses an unfiltered 1k-sample set from the same task recipe.

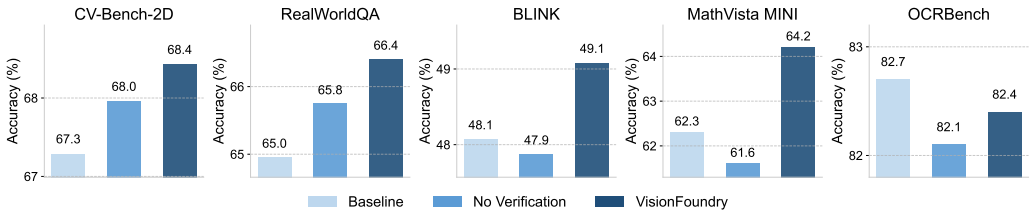

Figure 9: **Verification is helpful under a matched 1k-sample training budget.** The figure compares baseline, finetuning without verification, and finetuning with verification (VisionFoundry) on representative benchmarks. The verified branch performs better than the non-verified variant, with the clearest gains on visual perception evaluations.

Results and discussion. Figure 9 shows that verification is helpful under a matched 1k-sample training budget. Compared with *without verification*, *with verification* improves CV-Bench-2D by +0.5 points, RealWorldQA by +0.7 points, BLINK by +1.2 points, MathVista-Mini by +2.6 points, and OCRBench by +0.3 points. Relative to the baseline, the verified run improves on four of five benchmarks (CV-Bench-2D, RealWorldQA, BLINK, MathVista-Mini), while the non-verified run falls below baseline on BLINK, MathVista-Mini, and OCRBench. Overall, the gains are most visible on the vision-focused benchmarks listed above, while differences on other benchmarks are comparatively small. This confirms that verification is beneficial in this matched-1k setting.

H Synthetic Process Ablations

This section provides a practical comparison between a partially synthetic setting, real images + synthetic QA, and the full synthetic process used by VisionFoundry, synthetic images + synthetic QA. We defer stricter image-source isolation to the strict-control analysis.

Goal and protocol. The goal is to provide practical evidence on whether improvements mainly come from synthetic QA construction alone, or whether the full synthetic process itself provides additional training value. In VisionFoundry, the T2I prompt, question, and answer are generated in the same dialogue turn, so synthetic images are naturally aligned with text supervision and do not require manual captioning. For completeness, we construct a natural-image counterpart: we sample 1k images from LLaVA-Instruct-80K, generate captions with Gemini-3-Pro, then generate QA pairs from these captions and process them with the same pipeline to obtain a 1k training set. We compare this dataset against the matched 1k synthetic *Orientation and Direction* set under identical finetuning settings on Qwen2.5-VL-3B-Instruct (same unfreezing strategy, learning rates, epoch, and batch size), and report both against the no-finetuning baseline.

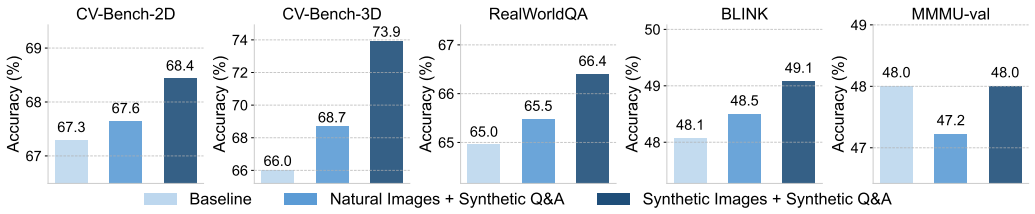

Figure 10: **Natural images with synthetic QA vs. synthetic images with synthetic QA under matched 1k-data training.** With the same model and setup, the synthetic-image setting shows stronger overall performance, with particularly clear gains on visual perception benchmarks while remaining comparable on the rest.

Results and discussion. Figure 10 shows that the synthetic-image setting outperforms the natural-image setting on all five reported benchmarks. Compared with real images + synthetic QA, synthetic images + synthetic QA improves CV-Bench-2D by +0.8 points, CV-Bench-3D by +5.3 points, RealWorldQA by +0.9 points, BLINK by +0.6 points, and MMMU-Val by +0.8 points. Both settings are broadly competitive on general-purpose evaluation, while the synthetic-image setting provides clearly larger gains on visual perception metrics (especially CV-Bench-3D). These results suggest that the full synthetic process provides added value beyond QA generation in VisionFoundry; we test this more rigorously in a strict-control analysis.

Strict-control ablation. To further isolate the image-source effect, we conduct a more strictly controlled variant where the synthetic QA component is explicitly matched across branches. The natural-image branch follows *Natural Image* $\rightarrow$ *Gemini-3-Pro Caption* $\rightarrow$ *LLM Synthetic QA*. For the synthetic-image branch, we reuse the same captions and QA, and only replace the image source via *Caption* $\rightarrow$ *T2I Model* $\rightarrow$ *Synthetic Image*. Hence, the key controlled variable is the image source (natural vs. caption-conditioned synthetic reconstruction), while QA supervision is kept consistent. The finetuning setup follows the controlled protocol established in Appendix G for Qwen2.5-VL-3B-Instruct: unfreeze adapters, ViT, and LLM; learning rates of 5×10^{-6} (adapter/LLM) and 5×10^{-7} (ViT); one epoch; batch size 128.

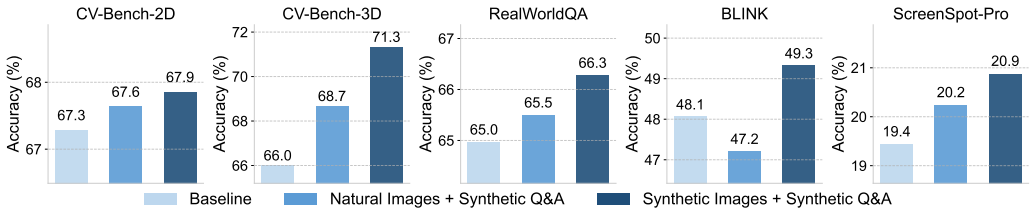

Figure 11: **Strict control experiment with matched synthetic QA.** Natural images and caption-conditioned synthetic images are compared under identical synthetic QA supervision and finetuning setup. Synthetic images achieve consistently stronger performance, with the largest gain on CV-Bench-3D.

Findings and discussion. Figure 11 shows that, even under matched synthetic QA, the synthetic-image branch still performs better overall. Compared with *natural images + synthetic QA*, *synthetic images + synthetic QA* improves CV-Bench-2D by +0.3 points, CV-Bench-3D by +2.6 points, RealWorldQA by +0.8 points, BLINK by +2.1 points, and ScreenSpot-Pro by +0.7 points. The natural-image branch remains competitive and improves over the baseline on several metrics, but the synthetic-image branch provides consistently stronger performance, especially on visual perception evaluations (particularly CV-Bench-3D). These findings further support that the synthetic process, particularly the caption-conditioned T2I reconstruction step, offers additional training value beyond caption-derived synthetic QA alone.

I T2I Model Ablations

This section examines whether the gains from VisionFoundry depend on the proprietary T2I model used in the main pipeline, or whether a strong open-source image generator can already provide similarly useful supervision.

Goal and protocol. The goal is to evaluate the reproducibility of the image-synthesis stage under the same downstream training budget and data construction recipe as in the preceding controlled ablations. We again use Qwen2.5-VL-3B-Instruct (Bai et al., 2025) as the backbone and keep the finetuning setup unchanged: unfreeze adapters, ViT, and LLM; use learning rates of 5×10^{-6} for adapter/LLM and 5×10^{-7} for ViT; train for one epoch with batch size 128. The training task remains *Orientation and Direction* with a matched final budget of 1k samples. We compare three settings: (i) the no-finetuning baseline, (ii) VisionFoundry data synthesized with the open-source T2I model Qwen-Image-2512 (Wu et al., 2025b), and (iii) VisionFoundry data synthesized with the proprietary T2I model Gemini-2.5-Flash-Image (Google Cloud, 2025a; Comanici et al., 2025) used in the main pipeline. All non-T2I components, including QA construction, verification, and the downstream finetuning recipe, are kept fixed, so the comparison isolates the choice of image generator.

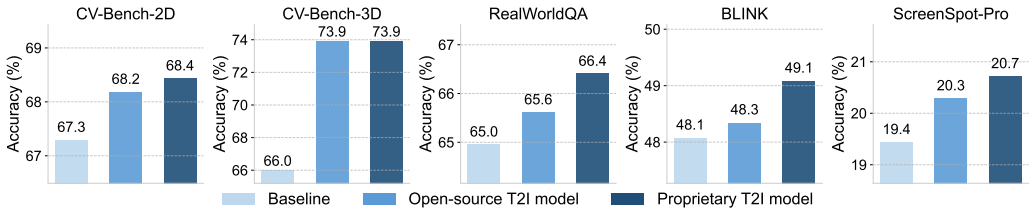

Figure 12: **Open-source vs. proprietary T2I models under matched 1k-data training.** Replacing the proprietary image generator in VisionFoundry with the open-source Qwen-Image-2512 still yields consistent improvements over the no-finetuning baseline. The proprietary T2I model remains slightly stronger overall, but the open-source variant retains most of the benefit, especially on visual perception benchmarks.

Results and discussion. Figure 12 shows that the open-source T2I setting is already effective. Compared with the baseline, finetuning on data synthesized by Qwen-Image-2512 improves CV-Bench-2D by +0.9 points, CV-Bench-3D by +7.9 points, RealWorldQA by +0.7 points, BLINK by +0.3 points, and ScreenSpot-Pro by +0.9 points. The proprietary T2I variant remains the strongest overall: relative to the open-source branch, it improves CV-Bench-2D by +0.3 points, RealWorldQA by +0.8 points, BLINK by +0.7 points, and ScreenSpot-Pro by +0.4 points, while both variants achieve the same CV-Bench-3D score. Thus, the open-source model does not fully match the proprietary frontier, but it still produces training data that consistently improves the downstream VLM. These results support meaningful open-source reproducibility for VisionFoundry and suggest that its gains do not rely specifically on a proprietary T2I model; a strong open-source image generator is already sufficient to make the pipeline effective.

J Ablation on Verifier Choice

One might view the verifier as imposing a hard upper bound on the downstream gains achievable by VisionFoundry. This section tests that interpretation directly.

Goal and protocol. The goal is to evaluate whether VisionFoundry mainly inherits supervision quality from a stronger verifier, or whether the pipeline remains effective even when the verifier is replaced by an open-source model that is not obviously stronger than the downstream baseline. We again use Qwen2.5-VL-3B-Instruct (Bai et al., 2025) as the backbone and keep the downstream finetuning setup unchanged: unfreeze adapters, ViT, and LLM; use learning rates of 5×10^{-6} for adapter/LLM and 5×10^{-7} for ViT; train for one epoch with batch size 128. The training task remains *Orientation and Direction* with a matched final budget of 1k samples. To isolate the verifier choice, we fix the T2I model to the open-source Qwen-Image-2512 (Wu et al., 2025b) used in the previous

section and compare three settings: (i) the no-finetuning baseline, (ii) VisionFoundry data filtered by an open-source verifier instantiated as Qwen2.5-VL-3B-Instruct itself (Bai et al., 2025), and (iii) VisionFoundry data filtered by the proprietary verifier Gemini-3-Pro (Google Cloud, 2025b) used in the main pipeline. All non-verifier components, including QA construction, T2I synthesis, and the downstream finetuning recipe, are kept fixed.

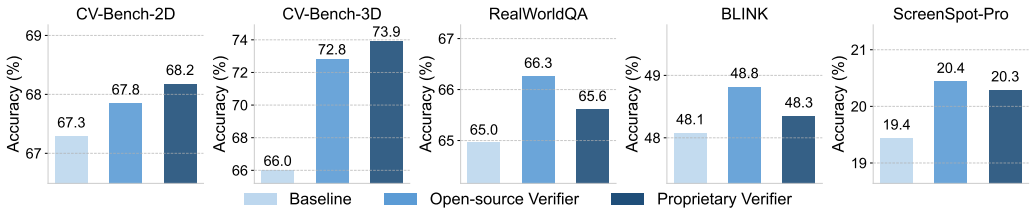

Figure 13: **Open-source vs. proprietary verifiers under matched 1k-data training.** With the T2I model fixed to Qwen-Image-2512, replacing Gemini-3-Pro with the open-source Qwen2.5-VL-3B-Instruct verifier still yields strong downstream gains. The two verifier choices trade wins across benchmarks rather than showing a one-sided dominance, suggesting that VisionFoundry's improvements are not simply inherited from the verifier ceiling.

Results and discussion. Figure 13 shows that both verifier choices are effective and that the comparison is genuinely mixed. Compared with the baseline, the open-source verifier improves CV-Bench-2D by +0.6 points, CV-Bench-3D by +6.8 points, RealWorldQA by +1.3 points, BLINK by +0.7 points, and ScreenSpot-Pro by +1.0 points. The proprietary verifier also improves strongly over the baseline, with gains of +0.9 points on CV-Bench-2D, +7.9 points on CV-Bench-3D, +0.7 points on RealWorldQA, +0.3 points on BLINK, and +0.9 points on ScreenSpot-Pro. Directly comparing the two verifier branches, the proprietary verifier is higher on CV-Bench-2D by +0.3 points and on CV-Bench-3D by +1.1 points, whereas the open-source verifier is higher on RealWorldQA by +0.7 points, BLINK by +0.5 points, and ScreenSpot-Pro by +0.2 points. Hence, the proprietary verifier does not uniformly dominate, even though it is stronger in the main pipeline. These results suggest that VisionFoundry is not simply obtaining useful supervision by delegating correctness to a stronger verifier alone; rather, the pipeline's task-aware generation-and-filtering structure itself contributes systematic training value. They also demonstrate a practical degree of verifier-side generality and reproducibility: the framework remains effective with an open-source verifier, including one instantiated from the same model family as the downstream baseline.

Full-scale self-verification. We further scale the self-verifier setting from the matched 1k controlled study to all 10 tasks. We construct a 10K dataset using local Qwen-Image-2512 synthesis and Qwen2.5-VL-3B-Instruct itself as the verifier, and finetune the same Qwen2.5-VL-3B-Instruct backbone. The resulting model improves over the pretrained baseline on all five visual-perception benchmarks, including +9.2 points on CV-Bench-3D and +5.4 points on $\mathrm{MMVP}_{\mathrm{pair}}$ (Table 9). Together with the matched verifier ablation above, this shows that useful VisionFoundry supervision does not require a stronger proprietary verifier. The downstream model is Qwen2.5-VL-3B-Instruct with full-parameter SFT for one epoch. We use a global batch size of 128, learning rates of 5×10^{-6} for the LLM/adapters and 5×10^{-7} for the vision encoder, and the full 10K self-verifier dataset.

Setting	$\mathrm{MMVP}_{\mathrm{pair}}$	$\mathrm{MMVP}_{\mathrm{single}}$	CV-Bench-2D	CV-Bench-3D	RealWorldQA
Qwen2.5-VL-3B baseline	35.3	64.3	67.3	66.0	65.0
Self-verifier 10K SFT	40.7	68.7	69.3	75.2	65.9
Improvement	+5.4	+4.4	+2.0	+9.2	+0.9

Table 9: **Full-scale self-verification on visual perception benchmarks.** Performance of Qwen2.5-VL-3B-Instruct trained on the 10K self-verified VisionFoundry variant, where both image synthesis (Qwen-Image-2512) and verifier (Qwen2.5-VL-3B-Instruct) are performed using open-source models.