Adapting General-Purpose Foundation Models for X-ray Ptychography in Low-Data Regimes

Robinson Umeike*
The University of Alabama
crumeike@crimson.ua.edu

Neil Getty
Argonne National Laboratory
ngetty@anl.gov

Yin Xiangyu
Argonne National Laboratory
xyin@anl.gov

Yi Jiang
Argonne National Laboratory
yjiang@anl.gov

Abstract

The automation of workflows in advanced microscopy is a key goal where foundation models like Language Models (LLMs) and Vision-Language Models (VLMs) show great potential. However, adapting these general-purpose models for specialized scientific tasks is critical, and the optimal domain adaptation strategy is often unclear. To address this, we introduce PtychoBench, a new multi-modal, multi-task benchmark for ptychographic analysis. Using this benchmark, we systematically compare two specialization strategies: Supervised Fine-Tuning (SFT) and In-Context Learning (ICL). We evaluate these strategies on a visual artifact detection task with VLMs and a textual parameter recommendation task with LLMs in a data-scarce regime. Our findings reveal that the optimal specialization pathway is task-dependent. For the visual task, SFT and ICL are highly complementary, with a fine-tuned model guided by context-aware examples achieving the highest mean performance (Micro-F1 of 0.728). Conversely, for the textual task, ICL on a large base model is the superior strategy, reaching a peak Micro-F1 of 0.847 and outperforming a powerful "super-expert" SFT model (0-shot Micro-F1 of 0.839). We also confirm the superiority of context-aware prompting and identify a consistent *contextual interference* phenomenon in fine-tuned models. These results, benchmarked against strong baselines including GPT-4o and a DINOv3-based classifier, offer key observations for AI in science: the optimal specialization path in our benchmark is dependent on the task modality, offering a clear framework for developing more effective science-based agentic systems.

Introduction

Ptychography is a popular characterization technique used across materials science [1,2], from semiconductors to biological specimens [3-6]. It computationally leverages redundant information in measured data to reconstruct the sample's structure at a spatial resolution far surpassing the limits of physical lenses. However, the quality of a ptychographic reconstruction depends on a complex, multi-dimensional optimization of experimental and algorithmic parameters. This process has typically relied on expert intuition and time-consuming, trial-and-error workflows. Recent work [7], demonstrated that agentic workflows orchestrating multiple Large Language Model (LLMs) and Vision-Language Model (VLM) can facilitate more streamlined and automated data analysis. The workflow includes a central Ptychography Agent (LLM) that gathers experimental context and recommends parameters, a Coding Agent that generates and execute reconstruction scripts, and a Diagnosis Agent (VLM) that visually assesses the reconstructed images for quality issues.
Although these agentic frameworks have demonstrated the viability of an LLM-driven workflow, the performance of their two core decision-making components, the VLM-based Diagnosis Agent and the LLM-based Ptychography Agent, had not been rigorously benchmarked. Their effectiveness, especially with new types of structures or artifacts that constitute out-of-distribution data for general-purpose models, remained a critical open question. To address this gap and systematically evaluate these agents, this paper makes two key contributions. First, we introduce PtychoBench, a novel, expert-annotated multi-modal, multi-task benchmark dataset for X-ray ptychographic analysis. This benchmark is designed to independently evaluate the two key

*Work performed while at Argonne National Laboratory.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) : AI4MAT-NeurIPS 2025.

agents with tasks of artifact detection for the VLM and parameter recommendation for the LLM, which is annotated on a smaller, more complex subset of the data.
This benchmark enables a systematic investigation into two distinct approaches for specializing these models: parameter-based specialization via supervised fine-tuning (SFT) [8-10] and in-context learning (ICL) [11-13]. We evaluate these strategies using open-weight Llama models of various scales (8B to 90B) and their fine-tuned counterparts [14], benchmarking their performance against a leading proprietary model like GPT-4o [15] and traditional state-of-the-art vision classifiers using DINOv3 embeddings [16]. This allows us to dissect the relationship between these two learning paradigms across different modalities and task complexities.
Our findings reveal two distinct, task-dependent pathways to expertise within our data-scarce setting. For the VLM-based artifact detection task, we find SFT and ICL are highly complementary, while for the LLM-based parameter tuning task, ICL on a large base model is the superior strategy, outperforming an overspecialized fine-tuned expert. Across all experiments, we identify a consistent *contextual interference* phenomenon where irrelevant context degrades SFT model performance. These observations highlight a critical trade-off between training-time and inference-time specialization, underscoring that the relevance of retrieved data is a primary driver of success. This work provides both a benchmark and a clear direction for future research, motivating the application of advanced algorithms like Group Relative Policy Optimization (GRPO) to enhance model reasoning [17].

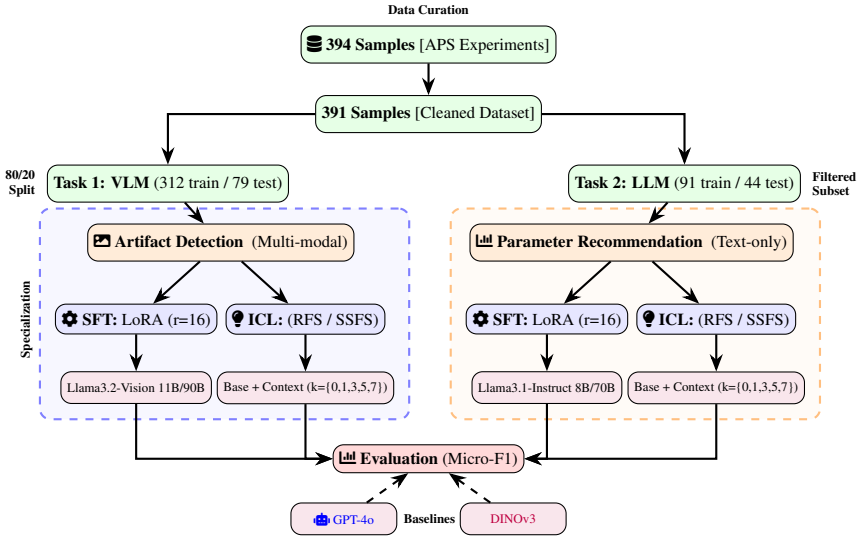


Figure 1: **Methodology Overview**: PtychoBench workflow from data curation through specialization strategies (SFT and ICL) to evaluation. The dataset is partitioned for two tasks: VLM-based artifact detection (blue) and LLM-based parameter recommendation (orange). Both tasks are evaluated using Micro-F1 against baselines including GPT-4o and DINOv3 (see supplementary material for more details).

1 Methodology and Experimental Design

This study, illustrated in Figure 1, systematically evaluates two specialization strategies using the PtychoBench dataset. The benchmark contains 391 expert-annotated samples for a multi-modal Artifact Detection (VQA) task and a 135-sample subset for a text-only Parameter Recommendation (QA) task. The full dataset was first partitioned into an 80/20 train/test split, from which the task-specific subsets were then derived.
We investigate two primary specialization strategies. The first is Supervised Fine-Tuning (SFT) [23], which finds specialized parameters θ_{SFT} by minimizing a loss function $\mathcal{L}$ over the training data D_{train}. For this, we employed a parameter-efficient approach using Low-Rank Adaptation (LoRA) with a rank (r) of 16 [21] and the AdamW 8-bit optimizer [24]. The second strategy is in-Context Learning (ICL), where a model's prediction $\hat{Y}$ for a test input x_{test} is conditioned on a context set of k examples, $C = \{(x_i, y_i)\}_{i=1}^{k}$, such that $\hat{Y} = f_\theta(x_{test}|C)$ [11]. We tested two context selection protocols for $k = \{0, 1, 3, 5, 7\}$: the Random Few-Shot (RFS), where examples are sampled randomly, and Sample-Specific Few-Shot (SSFS), where examples are selected based on matching sample type data.
This comprehensive setup allows us to investigate key questions relevant to deploying specialized AI agents for X-ray ptychography: **(1)** The optimal specialization strategy (SFT vs. ICL) for robust deployment in a data-scarce setting; **(2)** The impact of contextual relevance (SSFS vs. RFS) on agent performance; **(3)** Whether the distinct systems in an autonomous workflow require different specialization strategies; and **(4)** The effect of model scale on performance.

To investigate these questions, our evaluation includes open-weight Llama models at multiple scales (11B/90B for VLM, 8B/70B for LLM) and two powerful external benchmarks, GPT-4o and a DINOv3-based classifier. We also established two crucial internal baselines to contextualize performance: 1) 0-Shot Performance to measure the standalone capability of each model; and 2) the RFS strategy to provide a naive ICL baseline that quantifies the value of our retrieval strategy. The primary metric for all experiments is the Micro-F1 score. Complete hyperparameter configurations, prompt templates, and a detailed dataset breakdown are provided in the supplementary material.

Table 1: VLM Performance on Artifact Detection (Mean ± Std)

Model	Fewshot Strategy	0-shot	1-shot	3-shot	5-shot	7-shot
SFT Llama 3.2-Vision 11B	RFS	0.493 ± 0.041	0.322 ± 0.036 ↓	0.319 ± 0.041	0.389 ± 0.041	0.395 ± 0.041
	SSFS		**0.547 ± 0.044**	**0.636 ± 0.044**	**0.667 ± 0.043**	**0.713 ± 0.038**
Base Llama 3.2-Vision 11B	RFS	0.219 ± 0.025	0.254 ± 0.034	0.250 ± 0.035	0.313 ± 0.033	0.342 ± 0.036
	SSFS		**0.609 ± 0.047**	**0.628 ± 0.046**	**0.696 ± 0.041**	**0.694 ± 0.039**
SFT Llama 3.2-Vision 90B	RFS	0.430 ± 0.041	0.333 ± 0.036 ↓	0.409 ± 0.036	0.401 ± 0.037	0.446 ± 0.035
	SSFS		**0.586 ± 0.043**	**0.596 ± 0.046**	**0.671 ± 0.041**	**0.728 ± 0.036** *
Base Llama 3.2-Vision 90B	RFS	0.019 ± 0.014	0.255 ± 0.034	0.176 ± 0.035	0.090 ± 0.028	0.151 ± 0.038
	SSFS		**0.610 ± 0.048**	**0.628 ± 0.050**	**0.623 ± 0.049**	**0.706 ± 0.039**
GPT-4o	RFS	0.175 ± 0.026	0.163 ± 0.023	0.189 ± 0.027	0.248 ± 0.030	0.280 ± 0.039
	SSFS		**0.334 ± 0.037**	**0.529 ± 0.045**	**0.635 ± 0.044**	**0.656 ± 0.043**
DINOv3	-	0.628	-	-	-	-

Table 2: LLM Performance on Parameter Recommendation (Mean ± Std)

Model	Fewshot Strategy	0-shot	1-shot	3-shot	5-shot	7-shot
SFT Llama 3.1 8B	RFS	0.470 ± 0.051	0.316 ± 0.037 ↓	0.416 ± 0.044	0.497 ± 0.053	0.552 ± 0.039
	SSFS		**0.318 ± 0.050**	**0.602 ± 0.062**	**0.661 ± 0.056**	**0.697 ± 0.054**
Base Llama 3.1 8B	RFS	0.092 ± 0.022	0.183 ± 0.037	0.271 ± 0.034	0.452 ± 0.048	0.388 ± 0.047
	SSFS		**0.411 ± 0.037**	**0.602 ± 0.058**	**0.667 ± 0.066**	**0.709 ± 0.056**
SFT Llama 3.1 70B	RFS	0.839 ± 0.050	0.457 ± 0.047 ↓	0.493 ± 0.041	0.527 ± 0.049	0.595 ± 0.049
	SSFS		**0.518 ± 0.042**	**0.608 ± 0.050**	**0.683 ± 0.044**	**0.695 ± 0.047**
Base Llama 3.1 70B	RFS	0.228 ± 0.020	0.267 ± 0.038	0.362 ± 0.045	0.546 ± 0.055	0.575 ± 0.056
	SSFS		**0.543 ± 0.048**	**0.746 ± 0.040**	**0.784 ± 0.042**	**0.847 ± 0.042** *
GPT-4o	RFS	0.313 ± 0.030	0.340 ± 0.038	0.324 ± 0.044	0.518 ± 0.048	0.555 ± 0.050
	SSFS		**0.617 ± 0.048**	**0.709 ± 0.047**	**0.766 ± 0.046**	**0.781 ± 0.039**

2 Results and Analysis

Our experiments reveal a set of consistent findings regarding the importance of contextual relevance, alongside a striking divergence in optimal specialization strategies between the visual and textual tasks. The full performance metrics for Artifact Detection and Parameter Recommendation are presented in Table 1 and Table 2, respectively, with a comprehensive breakdown of all metrics, including 95% confidence intervals computed via bootstrapping ($n = 10,000$), provided in the Supplementary Material.

A. The Superiority of Context Relevant Prompting. The most consistent finding across all models and tasks is the dramatic superiority of Sample-Specific Few-Shot (SSFS) prompting over Random Few-Shot (RFS). Providing contextually relevant examples consistently yields a significant performance boost. For instance, in the artifact detection task (Table 1), the Base Llama 3.2-Vision 90B model's Micro-F1 score at 3-shots leaps from 0.176 ± 0.035 with RFS to 0.628 ± 0.050 with SSFS, a relative increase of 256%. This power of sample-specific context is most pronounced in base models, where a single relevant example can transform a model from non-functional to highly effective; the Base Llama 3.2-Vision 90B model's performance jumps from a 0-shot F1-score of 0.019 ± 0.014 to 0.610 ± 0.048 after seeing just one relevant sample. Conversely, the results highlight the danger of irrelevant context, indicated by performance drops in Tables 1 and 2 (↓). For fine-tuned models, RFS often leads to *contextual interference*, where performance degrades below the 0-shot baseline. This is evident in the SFT Llama 3.2-Vision 11B model, where a single random example causes the F1 score to drop from 0.493 ± 0.041 to 0.322 ± 0.036, a relative performance decrease of 35%. This counter-intuitive *contextual interference* phenomenon suggests that a single, irrelevant example can create ambiguity that overrides the model's specialized SFT training, making it a critical failure case to consider.

B. Task-Dependent Specialization Strategies. Aligning with established findings that context relevance is critical for zero- or few-shot performance, as demonstrated by the success of techniques such as Retrieval-Augmented Generation (RAG) [9, 12, 13, 20, 22], our results show that the optimal strategy to achieving peak performance is largely dependent on the task's modality.

I. Visual Material Diagnosis. For the artifact detection task, the highest mean performance is achieved when SFT and ICL are used together, with the SFT Llama 3.2-Vision 90B model and 7-shot SSFS reaching a peak

Micro-F1 of 0.728 $\pm$ 0.036 (Table 1). This outperforms the best-performing base model (Base 90B with 7-shot SSFS at 0.706 $\pm$ 0.039), demonstrating a clear combined effect where fine-tuning primes the model to better leverage in-context examples. The value of this domain-specific priming is highlighted by a comparison with the generalist model, GPT-4o. While GPT-4o is a capable in-context learner (reaching 0.656 $\pm$ 0.043 with SSFS), its 0-shot performance is a low 0.175 $\pm$ 0.026, far below that of our specialized SFT models (e.g., SFT-11B at 0.493 $\pm$ 0.041). This gap underscores the necessity of domain adaptation for this task. Ultimately, the peak score of SFT-90B + SSFS strategy surpasses that of the guided GPT-4o, confirming that for complex visual analysis, a specialized model combined with relevant context remains the state-of-the-art approach. However, SFT in a low-data regime reveals a paradox of scale: the smaller SFT-11B model has a stronger 0-shot performance than the larger SFT-90B (0.493 $\pm$ 0.041 vs. 0.430 $\pm$ 0.041). We attribute this to our training protocol: the number of training steps for each model was determined by monitoring performance on a validation set to prevent overfitting. The larger 90B model demonstrated faster convergence, and thus was trained for fewer steps ($\approx$ 20 epochs) to reach its optimal performance compared to the 11B model (50 epochs). Despite its lower baseline, the SFT-90B model shows a higher potential that is unlocked by ICL; its performance increases by over 69% from its 0-shot baseline when provided with 7 specific examples, a much larger gain than that seen for the SFT-11B model(45%).

II. Textual Recommendation. In stark contrast to the visual task, the results show that the most effective strategy is In-Context Learning on a large, general-purpose base model. The peak performance across all experiments was achieved by the Base Llama 3.1-70B model with 7-shot SSFS, reaching a Micro-F1 of 0.847 $\pm$ 0.042 (Table 2). This strategy not only surpassed all fine-tuned configurations but also outperformed the GPT-4o baseline, which peaked at 0.781 $\pm$ 0.039 under the same conditions. The fine-tuned models exhibit a fascinating and contrary behavior. The SFT-70B model emerges as a specialized "super-expert," achieving an exceptional 0-shot performance of 0.839 $\pm$ 0.050. This indicates that while SFT was highly effective at instilling expert knowledge, this knowledge became rigid; any form of ICL, even with relevant sample-specific examples, consistently degraded its performance. This finding, where more context hurts performance, highlights a key risk of over-specialization in a low-data regime. This antagonistic relationship between SFT and ICL for this task suggests that for certain reasoning pathways, fine-tuning may create a rigid model that is easily disturbed by external context. Finally, the results also underscore the critical role of scale for ICL on base models. The peak performance of the Base-70B model (0.847 $\pm$ 0.042) represents a significant leap over the Base-8B model (0.709 $\pm$ 0.056), confirming that for this text-based reasoning task, a larger model is substantially more capable of leveraging in-context examples to achieve expert-level performance.

3 Discussion and Conclusion

A key finding from our work is that the optimal strategy for specializing AI models in our benchmark is not universal but appears to be dependent on the task's modality. Our results show two distinct pathways to expert performance: a complementary relationship between SFT and ICL for the visual-perceptual task, and the supremacy of ICL on a large base model for the textual-reasoning task. We hypothesize this divergence is rooted in what SFT learns in a low-data regime; for visual diagnosis, SFT builds a foundational "visual vocabulary" that ICL can effectively refine, while for the more abstract recommendation task, SFT may create a powerful but inflexible expert by overfitting to specific reasoning paths. We also note that these two tasks differ in data volume and complexity. While our results point to modality as a key factor, further research would be valuable to fully disentangle these variables.

The implications for designing autonomous science systems are significant, suggesting a design guideline where the specialization strategy is tailored to the model's function. For instance, our VLM results were contextualized against a strong DINOv3-based classifier (micro-F1 of 0.628), which used a 7B model as a feature extractor trained on our full 312-sample training set. The fact that our Base-90B + SSFS strategy (0.706 $\pm$ 0.039), using only 7 examples at inference, achieved a mean performance substantially higher than the fully-trained DINOv3 baseline, highlights the remarkable data efficiency of the ICL paradigm. This demonstrates that a well guided general purpose VLM can effectively match or even exceed the performance of a traditional, fully trained classifier, despite using a fraction of the data. This suggests perceptual AI models may benefit most from a hybrid SFT+ICL approach, while models for downstream reasoning may achieve higher performance and flexibility as large base models guided by a sophisticated retrieval system.

While our findings are robust across multiple model scales, this study is primarily based on the PtychoBench dataset and Llama model family; future work should validate these observations on other scientific domains. Furthermore, our results underscore that high-quality context is paramount. This motivates our future work in exploring data augmentation strategies to improve robustness and applying reinforcement learning algorithms, such as GRPO, to move beyond providing static examples and instead foster genuine, step-by-step reasoning pathways in scientific AI systems.

4 Acknowledgments and Disclosure of Funding

This work was supported by the Laboratory Directed Research and Development (LDRD) Program at Argonne National Laboratory under Project Number 2025-0495. This research used resources of the Advanced Photon Source, a U.S. Department of Energy (DOE) Office of Science User Facility operated for the DOE Office of Science by Argonne National Laboratory under Contract No. DE-AC02-06CH11357.

5 Data and Code Availability

To foster further research in this domain, our code and LoRA adapters are available at `https://github.com/crumeike/PtychoBench`. The PtychoBench dataset is available upon request and subject to institutional approval. Requests regarding data access should be directed to `yjiang@anl.gov`. For questions about the code, please contact `crumeike@crimson.ua.edu` or `ngetty@anl.gov`.

References

[1] F. Pfeiffer, (2017) X-ray ptychography. *Nature Photonics* **12**(1):9–17.

[2] J. Miao, (2025) Computational microscopy with coherent diffractive imaging and ptychography. *Nature* **637**(8045):281–295.

[3] M. Holler, M. Guizar-Sicairos, E. H. R. Tsai, R. Dinapoli, E. Müller, O. Bunk, J. Raabe, and G. Aeppli, (2017) High-resolution non-destructive three-dimensional imaging of integrated circuits. *Nature* **543**(7645):402–406.

[4] T. Aidukas, N. W. Phillips, A. Diaz, E. Poghosyan, E. Müller, A. F. J. Levi, G. Aeppli, M. Guizar-Sicairos, and M. Holler, (2024) High-performance 4-nm-resolution X-ray tomography using burst ptychography. *Nature* **632**(8023):81–88.

[5] Y. Young-Sang, M. Farmand, C. Kim, Y. Liu, C. P. Grey, F. C. Strobridge, T. Tyliszczak, R. Celestre, P. Denes, J. Joseph, H. Krishnan, F. R. N. C. Maia, A. L. D. Kilcoyne, S. Marchesini, T. P. C. Leite, T. Warwick, H. Padmore, J. Cabana, and D. A. Shapiro, (2018) Three-dimensional localization of nanoscale battery reactions using soft X-ray tomography. *Nat Commun* **9**(1).

[6] J. Deng, Y. H. Lo, M. Gallagher-Jones, S. Chen, A. Pryor Jr, Q. Jin, Y. P. Hong, Y. S. G. Nashed, S. Vogt, J. Miao, and C. Jacobsen, (2018) Correlative 3D x-ray fluorescence and ptychographic tomography of frozen-hydrated green algae. *Sci. Adv.* **4**(11).

[7] X. Yin, C. Shi, Y. Han, and Y. Jiang, (2024) PEAR: A Robust and Flexible Automation Framework for Ptychography Enabled by Multiple Large Language Model Agents. arXiv.

[8] C. Hyung Won, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. Shane Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, D. Valter, S. Narang, G. Mishra, A. Wei Yu, V. Zhao, Y. Huang, A. M. Dai, H. Yu, S. Petrov, Ed H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. v Le, and J. Wei, (2022) Scaling Instruction-Finetuned Language Models. arXiv.

[9] R. Umeike, N. Getty, F. Xia, and R. Stevens, (2025) Scaling Large Vision-Language Models for Enhanced Multimodal Comprehension in Biomedical Image Analysis. In *2025 IEEE 22nd International Symposium on Biomedical Imaging (ISBI)*, pp. 1–4.

[10] H. Liu, C. Li, Q. Wu, and Y. J. Lee, (2023) Visual Instruction Tuning. In *Advances in Neural Information Processing Systems*, Curran Associates, Inc., pp. 34892–34916.

[11] Q. Huang, H. Ren, P. Chen, G. Krzmanc, D. Zeng, P. Liang, and J. Leskovec, (2023) PRODIGY: enabling in-context learning over graphs. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023*, New Orleans, LA, USA, December 10–16.

[12] Q. Dong, L. Li, D. Dai, C. Zheng, J. Ma, R. Li, H. Xia, J. Xu, Z. Wu, T. Liu, B. Chang, X. Sun, L. Li, and Z. Sui, (2023) A Survey on In-context Learning. arXiv.

[13] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, (2020) Language Models are Few-Shot Learners. arXiv.

[14] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang, A. Mitra, A. Sravankumar, A. Korenev, A. Hinsvark, A. Rao, A. Zhang, A. Rodriguez, A. Gregerson, A. Spataru, B. Roziere, B. Biron, B. Tang, B. Chern, C. Caucheteux, C. Nayak, C. Bi, C. Marra, C. McConnell, C. Keller, C. Touret, C. Wu, C. Wong, C. C. Ferrer, C. Nikolaidis, D. Allonsius, D. Song, D. Pintz, D. Livshits, D. Wyatt, D. Esiobu, D. Choudhary, D. Mahajan, D. Garcia-Olano, D. Perino, D. Hupkes, E. Lakomkin, E. AlBadawy, E. Lobanova, E. Dinan, E. M. Smith, F. Radenovic, F. Guzmán, F. Zhang, G.

Synnaeve, G. Lee, G. L. Anderson, G. Thattai, G. Nail, G. Mialon, G. Pang, G. Cucurell, H. Nguyen, H. Korevaar, H. Xu, H. Touvron, I. Zarov, I. A. Ibarra, I. Kloumann, I. Misra, I. Evtimov, J. Zhang, J. Copet, J. Lee, J. Geffert, J. Vranes, J. Park, J. Mahadeokar, J. Shah, J. van der Linde, J. Billock, J. Hong, J. Lee, J. Fu, J. Chi, J. Huang, J. Liu, J. Wang, J. Yu, J. Bitton, J. Spisak, J. Park, J. Rocca, J. Johnstun, J. Saxe, J. Jia, K. V. Alwala, K. Prasad, K. Upasani, K. Plawiak, K. Li, K. Heafield, K. Stone, K. El-Arini, K. Iyer, K. Malik, K. Chiu, K. Bhalla, K. Lakhotia, L. Rantala-Yeary, L. van der Maaten, L. Chen, L. Tan, L. Jenkins, L. Martin, L. Madaan, L. Malo, L. Blecher, L. Landzaat, L. de Oliveira, M. Muzzi, M. Pasupuleti, M. Singh, M. Paluri, M. Kardas, M. Tsimpoukelli, M. Oldham, M. Rita, M. Pavlova, M. Kambadur, M. Lewis, M. Si, M. K. Singh, M. Hassan, N. Goyal, N. Torabi, N. Bashlykov, N. Bogoychev, N. Chatterji, N. Zhang, O. Duchenne, O. Çelebi, P. Alrassy, P. Zhang, P. Li, P. Vasic, P. Weng, P. Bhargava, P. Dubal, P. Krishnan, P. S. Koura, P. Xu, Q. He, Q. Dong, R. Srinivasan, R. Ganapathy, R. Calderer, R. S. Cabral, R. Stojnic, R. Raileanu, R. Maheswari, R. Girdhar, R. Patel, R. Sauvestre, R. Polidoro, R. Sumbaly, R. Taylor, R. Silva, R. Hou, R. Wang, S. Hosseini, S. Chennabasappa, S. Singh, S. Bell, S. S. Kim, S. Edunov, S. Nie, S. Narang, S. Raparthy, S. Shen, S. Wan, S. Bhosale, S. Zhang, S. Vandenhende, S. Batra, S. Whitman, S. Sootla, S. Collot, S. Gururangan, S. Borodinsky, T. Herman, T. Fowler, T. Sheasha, T. Georgiou, T. Scialom, T. Speckbacher, T. Mihaylov, T. Xiao, U. Karn, V. Goswami, V. Gupta, V. Ramanathan, V. Kerkez, V. Gonguet, V. Do, V. Vogeti, V. Albiero, V. Petrovic, W. Chu, W. Xiong, W. Fu, W. Meers, X. Martinet, X. Wang, X. Wang, X. E. Tan, X. Xia, X. Xie, X. Jia, X. Wang, Y. Goldschlag, Y. Gaur, Y. Babaei, Y. Wen, Y. Song, Y. Zhang, Y. Li, Y. Mao, Z. D. Coudert, Z. Yan, Z. Chen, Z. Papakipos, A. Singh, A. Srivastava, A. Jain, A. Kelsey, A. Shajnfeld, A. Gangidi, A. Victoria, A. Goldstand, A. Menon, A. Sharma, A. Boesenberg, A. Baevski, A. Feinstein, A. Kallet, A. Sangani, A. Teo, A. Yunus, A. Lupu, A. Alvarado, A. Caples, A. Gu, A. Ho, A. Poulton, A. Ryan, A. Ramchandani, A. Dong, A. Franco, A. Goyal, A. Saraf, A. Chowdhury, A. Gabriel, A. Bharambe, A. Eisenman, A. Yazdan, B. James, B. Maurer, B. Leonhardi, B. Huang, B. Loyd, B. De Paola, B. Paranjape, B. Liu, B. Wu, B. Ni, B. Hancock, B. Wasti, B. Spence, B. Stojkovic, B. Gamido, B. Montalvo, C. Parker, C. Burton, C. Mejia, C. Liu, C. Wang, C. Kim, C. Zhou, C. Hu, C.-H. Chu, C. Cai, C. Tindal, C. Feichtenhofer, C. Gao, D. Civin, D. Beaty, D. Kreymer, D. Li, D. Adkins, D. Xu, D. Testuggine, D. David, D. Parikh, D. Liskovich, D. Foss, D. Wang, D. Le, D. Holland, E. Dowling, E. Jamil, E. Montgomery, E. Presani, E. Hahn, E. Wood, E.-T. Le, E. Brinkman, E. Arcaute, E. Dunbar, E. Smothers, F. Sun, F. Kreuk, F. Tian, F. Kokkinos, F. Ozgenel, F. Caggioni, F. Kanayet, F. Seide, G. M. Florez, G. Schwarz, G. Badeer, G. Swee, G. Halpern, G. Herman, G. Sizov, Guangyi, Zhang, G. Lakshminarayanan, H. Inan, H. Shojanazeri, H. Zou, H. Wang, H. Zha, H. Habeeb, H. Rudolph, H. Suk, H. Aspegren, H. Goldman, H. Zhan, I. Damlaj, I. Molybog, I. Tufanov, I. Leontiadis, I.-E. Veliche, I. Gat, J. Weissman, J. Geboski, J. Kohli, J. Lam, J. Asher, J.-B. Gaya, J. Marcus, J. Tang, J. Chan, J. Zhen, J. Reizenstein, J. Teboul, J. Zhong, J. Jin, J. Yang, J. Cummings, J. Carvill, J. Shepard, J. McPhie, J. Torres, J. Ginsburg, J. Wang, K. Wu, K. H. U, K. Saxena, K. Khandelwal, K. Zand, K. Matosich, K. Veeraraghavan, K. Michelena, K. Li, K. Jagadeesh, K. Huang, K. Chawla, K. Huang, L. Chen, L. Garg, L. A, L. Silva, L. Bell, L. Zhang, L. Guo, L. Yu, L. Moshkovich, L. Wehrstedt, M. Khabsa, M. Avalani, M. Bhatt, M. Mankus, M. Hasson, M. Lennie, M. Reso, M. Groshev, M. Naumov, M. Lathi, M. Keneally, M. Liu, M. L. Seltzer, M. Valko, M. Restrepo, M. Patel, M. Vyatskov, M. Samvelyan, M. Clark, M. Macey, M. Wang, M. J. Hermoso, M. Metanat, M. Rastegari, M. Bansal, N. Santhanam, N. Parks, N. White, N. Bawa, N. Singhal, N. Egebo, N. Usunier, N. Mehta, N. P. Laptev, N. Dong, N. Cheng, O. Chernoguz, O. Hart, O. Salpekar, O. Kalinli, P. Kent, P. Parekh, P. Saab, P. Balaji, P. Rittner, P. Bontrager, P. Roux, P. Dollar, P. Zvyagina, P. Ratanchandani, P. Yuvraj, Q. Liang, R. Alao, R. Rodriguez, R. Ayub, R. Murthy, R. Nayani, R. Mitra, R. Parthasarathy, R. Li, R. Hogan, R. Battey, R. Wang, R. Howes, R. Rinott, S. Mehta, S. Siby, S. J. Bondu, S. Datta, S. Chugh, S. Hunt, S. Dhillon, S. Sidorov, S. Pan, S. Mahajan, S. Verma, S. Yamamoto, S. Ramaswamy, S. Lindsay, S. Lindsay, S. Feng, S. Lin, S. C. Zha, S. Patil, S. Shankar, S. Zhang, S. Zhang, S. Wang, S. Agarwal, S. Sajuyigbe, S. Chintala, S. Max, S. Chen, S. Kehoe, S. Satterfield, S. Govindaprasad, S. Gupta, S. Deng, S. Cho, S. Virk, S. Subramanian, S. Choudhury, S. Goldman, T. Remez, T. Glaser, T. Best, T. Koehler, T. Robinson, T. Li, T. Zhang, T. Matthews, T. Chou, T. Shaked, V. Vontimitta, V. Ajayi, V. Montanez, V. Mohan, V. S. Kumar, V. Mangla, V. Ionescu, V. Poenaru, V. T. Mihailescu, V. Ivanov, W. Li, W. Wang, W. Jiang, W. Bouaziz, W. Constable, X. Tang, X. Wu, X. Wang, X. Wu, X. Gao, Y. Kleinman, Y. Chen, Y. Hu, Y. Jia, Y. Qi, Y. Li, Y. Zhang, Y. Zhang, Y. Adi, Y. Nam, Yu, Wang, Y. Zhao, Y. Hao, Y. Qian, Y. Li, Y. He, Z. Rait, Z. DeVito, Z. Rosnbrick, Z. Wen, Z. Yang, Z. Zhao, and Z. Ma, (2024) The Llama 3 Herd of Models. arXiv.

[15] OpenAI, :, A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, A. Mądry, A. Baker-Whitcomb, A. Beutel, A. Borzunov, A. Carney, A. Chow, A. Kirillov, A. Nichol, A. Paino, A. Renzin, A. T. Passos, A. Kirillov, A. Christakis, A. Conneau, A. Kamali, A. Jabri, A. Moyer, A. Tam, A. Crookes, A. Tootoonchian, A. Tootoonchian, A. Kumar, A. Vallone, A. Karpathy, A. Braunstein, A. Cann, A. Codispoti, A. Galu, A. Kondrich, A. Tulloch, A. Mishchenko, A. Baek, A. Jiang, A. Pelisse, A. Woodford, A. Gosalia, A. Dhar, A. Pantuliano, A. Nayak, A. Oliver, B. Zoph, B. Ghorbani, B. Leimberger, B. Rossen, B. Sokolowsky, B. Wang, B. Zweig, B. Hoover, B. Samic, B. McGrew, B. Spero, B. Giertler, B. Cheng, B. Lightcap, B. Walkin, B. Quinn, B. Guarraci, B. Hsu, B. Kellogg, B. Eastman, C. Lugaresi, C. Wainwright, C. Bassin, C. Hudson, C. Chu, C. Nelson, C. Li, C. J. Shern, C. Conger, C. Barette, C. Voss, C. Ding, C. Lu, C. Zhang, C. Beaumont, C. Hallacy, C. Koch, C. Gibson, C. Kim, C. Choi, C. McLeavey, C. Hesse, C. Fischer, C. Winter, C. Czarnecki, C. Jarvis, C. Wei, C. Koumouzelis, D. Sherburn, D. Kappler, D. Levin, D. Levy, D. Carr, D. Farhi, D. Mely, D. Robinson, D. Sasaki, D. Jin, D. Valladares, D. Tsipras, D. Li, D. P. Nguyen, D. Findlay, E. Oiwoh, E. Wong, E. Asdar, E. Proehl, E. Yang, E. Antonow, E. Kramer, E. Peterson, E. Sigler, E. Wallace, E. Brevdo, E. Mays, F. Khorasani, F. P. Such, F. Raso, F. Zhang, F. von Lohmann, F. Sulit, G. Goh, G. Oden, G. Salmon, G. Starace, G. Brockman, H. Salman, H. Bao, H. Hu, H. Wong, H. Wang, H. Schmidt, H. Whitney, H. Jun, H. Kirchner, H. P. de O. Pinto, H. Ren, H. Chang, H. W. Chung,

I. Kivlichan, I. O'Connell, I. O'Connell, I. Osband, I. Silber, I. Sohl, I. Okuyucu, I. Lan, I. Kostrikov, I. Sutskever, I. Kanitscheider, I. Gulrajani, J. Coxon, J. Menick, J. Pachocki, J. Aung, J. Betker, J. Crooks, J. Lennon, J. Kiros, J. Leike, J. Park, J. Kwon, J. Phang, J. Teplitz, J. Wei, J. Wolfe, J. Chen, J. Harris, J. Varavva, J. G. Lee, J. Shieh, J. Lin, J. Yu, J. Weng, J. Tang, J. Yu, J. Jang, J. Q. Candela, J. Beutler, J. Landers, J. Parish, J. Heidecke, J. Schulman, J. Lachman, J. McKay, J. Uesato, J. Ward, J. W. Kim, J. Huizinga, J. Sitkin, J. Kraaijeveld, J. Gross, J. Kaplan, J. Snyder, J. Achiam, J. Jiao, J. Lee, J. Zhuang, J. Harriman, K. Fricke, K. Hayashi, K. Singhal, K. Shi, K. Karthik, K. Wood, K. Rimbach, K. Hsu, K. Nguyen, K. Gu-Lemberg, K. Button, K. Liu, K. Howe, K. Muthukumar, K. Luther, L. Ahmad, L. Kai, L. Itow, L. Workman, L. Pathak, L. Chen, L. Jing, L. Guy, L. Fedus, L. Zhou, L. Mamitsuka, L. Weng, L. McCallum, L. Held, L. Ouyang, L. Feuvrier, L. Zhang, L. Kondraciuk, L. Kaiser, L. Hewitt, L. Metz, L. Doshi, M. Aflak, M. Simens, M. Boyd, M. Thompson, M. Dukhan, M. Chen, M. Gray, M. Hudnall, M. Zhang, M. Aljubeh, M. Litwin, M. Zeng, M. Johnson, M. Shetty, M. Gupta, M. Shah, M. Yatbaz, M. J. Yang, M. Zhong, M. Glaese, M. Chen, M. Janner, M. Lampe, M. Petrov, M. Wu, M. Wang, M. Fradin, M. Pokrass, M. Castro, M. O. T. de Castro, M. Pavlov, M. Brundage, M. Wang, M. Khan, M. Murati, M. Bavarian, M. Lin, M. Yesildal, N. Soto, N. Gimelshein, N. Cone, N. Staudacher, N. Summers, N. LaFontaine, N. Chowdhury, N. Ryder, N. Stathas, N. Turley, N. Tezak, N. Felix, N. Kudige, N. Keskar, N. Deutsch, N. Bundick, N. Puckett, O. Nachum, O. Okelola, O. Boiko, O. Murk, O. Jaffe, O. Watkins, O. Godement, O. Campbell-Moore, P. Chao, P. McMillan, P. Belov, P. Su, P. Bak, P. Bakkum, P. Deng, P. Dolan, P. Hoeschele, P. Welinder, P. Tillet, P. Pronin, P. Tillet, P. Dhariwal, Q. Yuan, R. Dias, R. Lim, R. Arora, R. Troll, R. Lin, R. G. Lopes, R. Puri, R. Miyara, R. Leike, R. Gaubert, R. Zamani, R. Wang, R. Donnelly, R. Honsby, R. Smith, R. Sahai, R. Ramchandani, R. Huet, R. Carmichael, R. Zellers, R. Chen, R. Chen, R. Nigmatullin, R. Cheu, S. Jain, S. Altman, S. Schoenholz, S. Toizer, S. Miserendino, S. Agarwal, S. Culver, S. Ethersmith, S. Gray, S. Grove, S. Metzger, S. Hermani, S. Jain, S. Zhao, S. Wu, S. Jomoto, S. Wu, Shuaiqi, Xia, S. Phene, S. Papay, S. Narayanan, S. Coffey, S. Lee, S. Hall, S. Balaji, T. Broda, T. Stramer, T. Xu, T. Gogineni, T. Patwardhan, T. Cunninghman, T. Degry, T. Dimson, T. Raoux, T. Shadwell, T. Zheng, T. Underwood, T. Markov, T. Sherbakov, T. Rubin, T. Stasi, T. Kaftan, T. Heywood, T. Peterson, T. Walters, T. Eloundou, V. Qi, V. Moeller, V. Monaco, V. Kuo, V. Fomenko, W. Chang, W. Zheng, W. Zhou, W. Manassra, W. Sheu, W. Zaremba, Y. Patil, Y. Qian, Y. Kim, Y. Cheng, Y. Zhang, Y. He, Y. Zhang, Y. Jin, Y. Dai, and Y. Malkov, (2024) GPT-4o System Card. arXiv.

[16] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa, F. Massa, D. Haziza, L. Wehrstedt, J. Wang, T. Darcet, T. Moutakanni, L. Sentana, C. Roberts, A. Vedaldi, J. Tolan, J. Brandt, C. Couprie, J. Mairal, H. Jégou, P. Labatut, and P. Bojanowski, (2025) DINOv3. arXiv.

[17] Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. K. Li, Y. Wu, and D. Guo, (2024) DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv.

[18] I. Beltagy, K. Lo, and A. Cohan, (2019) SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 EMNLP-IJCNLP*, Hong Kong, China: Association for Computational Linguistics, Nov., pp. 3615–3620.

[19] H. Wang, T. Fu, Y. Du, W. Gao, K. Huang, Z. Liu, P. Chandak, S. Liu, P. Van Katwyk, A. Deac, A. Anandkumar, K. Bergen, C. P. Gomes, S. Ho, P. Kohli, J. Lasenby, J. Leskovec, T.-Y. Liu, A. Manrai, D. Marks, B. Ramsundar, L. Song, J. Sun, J. Tang, P. Veličković, M. Welling, L. Zhang, C. W. Coley, Y. Bengio, and M. Zitnik, (2023) Scientific discovery in the age of artificial intelligence. *Nature* **620**(7972):47–60.

[20] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, (2020) Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. arXiv.

[21] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, (2021) LoRA: Low-Rank Adaptation of Large Language Models. arXiv.

[22] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang, (2023) Retrieval-Augmented Generation for Large Language Models: A Survey. arXiv.

[23] Daniel Han, Michael Han, and Unsloth team, (2023) Unsloth.

[24] I. Loshchilov and F. Hutter, (2019) Decoupled Weight Decay Regularization. In *7th International Conference on Learning Representations, ICLR 2019*, New Orleans, LA, USA, May 6–9.

A Technical Appendices and Supplementary Material

A.1 Related Works

Our work is situated at the intersection of foundation models for scientific discovery [18, 19] and domain adaptation techniques. While many studies explore either parameter-efficient fine-tuning (PEFT) techniques like LoRA [9] or in-context learning (ICL) with retrieval augmentation [20], a systematic comparison of these strategies in specialized, low-data scientific domains has been lacking.

A.2 The PtychoBench Dataset

This section provides a detailed description of the PtychoBench benchmark, which was curated to facilitate a reproducible and quantitative study of AI agents for ptychographic analysis.

A.2.1 Data Curation and Partitioning

The dataset was constructed from an initial pool of 394 ptychographic reconstructions of experimental data acquired at the Advanced Photon Source, each annotated by a domain expert. After cleaning, the primary dataset consists of 391 samples. This full dataset was first partitioned into a training set of 312 samples and a test set of 79 samples. This primary 80/20 partition serves as the basis for the Artifact Detection task. Subsequently, the dataset for the Parameter Recommendation task was derived by creating subsets from these existing partitions. We filtered both the training and test sets to include only those instances which contained an expert's free-text recommendation label (caption_param). This resulted in a final training set of 91 samples and a test set of 44 samples for the recommendation task. See Table 3 for sample data field.

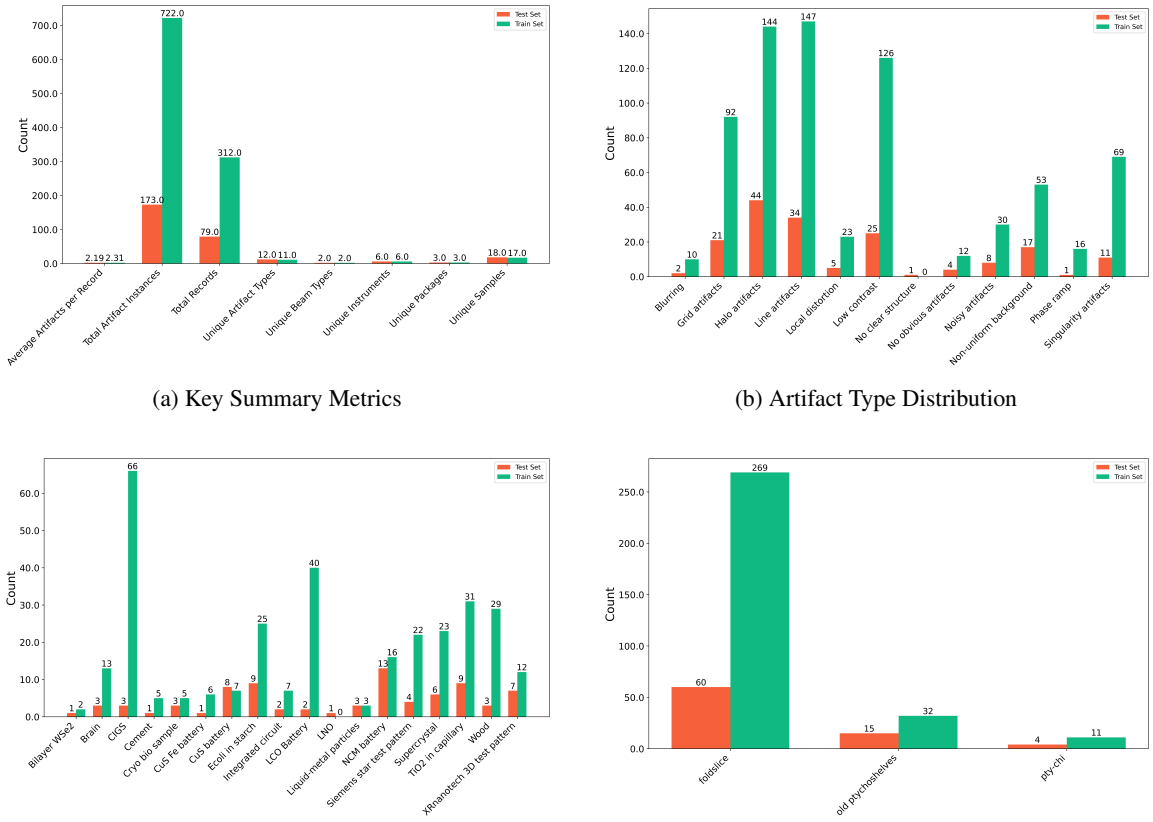
Table 3: **Dataset Field Descriptions and Examples**: The table presents the structure and annotation schema of the PtychoBench dataset, detailing each field's purpose and usage in the VLM and LLM training and evaluation tasks.

Field Name	Data Type	Description	Example Value	Usage
`id`	Integer	Unique sample identifier	2	Both tasks
`captioning`	String	Path to ptychographic reconstruction image	`/data/upload/1/af707e7b-...png`	VLM Training
`beam_choice`	Categorical	Type of radiation beam used	X-ray	LLM Training
`instrument_ choice`	Categorical	Specific beamline/instrument identifier	APS-2IDE-XFM	LLM Training
`sample_text`	Categorical	Material/sample type being imaged	Integrated circuit	SSFS retrieval, LLM Training
`package_ choice`	Categorical	Reconstruction software package	pty-chi	LLM Training
`artifact_ choice`	List	Expert-annotated visual artifacts present	[*"Local distortion"*]	**VLM Target**
`caption_obj`	String	Expert description of visual artifacts/quality	*"Lines look zig-zag with discontinuities..."*	LLM Training
`caption_ param`	String	Expert parameter recommendations	*"Increase iterations or use smaller patterns..."*	**LLM Target**
`recon_path_ text`	String	Full file path to reconstruction data	`/mnt/micdata1/2ide/2025-1/...`	Metadata
`annotator`	Integer	Expert annotator identifier	1	Quality control
`annotation_ id`	Integer	Annotation session identifier	3	Quality control
`created_at`	Timestamp	Initial annotation timestamp	2025-05-14T20:10:35Z	Metadata
`updated_at`	Timestamp	Last modification timestamp	2025-06-02T19:22:01Z	Metadata
`lead_time`	Float	Annotation time in seconds	180.11	Quality metrics

A.2.2 Task Definitions

The PtychoBench benchmark is designed for two sequential tasks central to autonomous characterization:

- **Artifact Detection**: A multi-modal, multi-label classification task where a Vision-Language Model (VLM) takes a ptychographic image as input and identifies a set of visual artifacts or defects.
- **Parameter Recommendation**: A text-only, multi-label classification task where a Language Model (LLM) takes a textual description of observed artifacts and experimental conditions as input and recommends corrective actions.



(a) Key Summary Metrics

(b) Artifact Type Distribution

(c) Sample Type Distribution

(d) Software Package Distribution

Figure 2: Statistical distributions of the PtychoBench dataset across the training and test sets. The summary plot (a) is followed by detailed breakdowns for artifact types (b), sample types (c), and software packages (d).

A.2.3 Dataset Composition and Characteristics

The PtychoBench is characterized by its diversity and complexity, reflecting real-world experimental conditions. The full dataset spans 18 distinct sample types and 6 unique instruments. The Artifact Detection task presents a challenging multi-label problem. The 79 samples in the test set contain a total of 173 annotated artifact instances, averaging 2.19 artifacts per image. Across the entire dataset, the most prevalent of the 12 possible artifact types are Halo artifacts and Line artifacts. The Parameter Recommendation task is similarly complex, drawing from a rich context of metadata and observed issues. A complete statistical breakdown of the dataset composition, including the distributions for all categorical metadata as provided, is detailed in Figure 1 and Table 3.

Table 4: **Task-Specific Data Usage**. The table details how different fields from the PtychoBench dataset are utilized in the VLM artifact detection task and LLM parameter recommendation task.

Task	Input Features	Target Labels	Context Selection	Sample Size
VLM Artifact Detection	captioning (image) + caption_obj (description) [sample_text, instrument_choice, beam_choice]	**artifact_choice** (multi-label)	sample_text matching	**391**
LLM Parameter Recommendation	prompt + context [beam_choice, instrument_choice, sample_text, package_choice, caption_obj (artifact description)]	**caption_param** (text gen)	sample_text matching	**135**

A.3 Task Formulation and Prompt Engineering

The prompts for both tasks were structured to emulate a real-world interaction with an expert AI agent, providing clear context and instructions. The data was formatted into a conversational structure with USER and ASSISTANT roles.

A.3.1 Task 1: LLM-based Parameter Recommendation

This task is a text-only, multi-label classification problem simulating the decision-making step that follows artifact diagnosis.

Prompt Template: The prompt provides a rich textual context constructed by populating the template below with a sample's specific metadata.

Parameter Recommendation Prompt

SAMPLE INFORMATION:

Package Choice: package
Instrument Used: instrument
Sample Type: sample
Beam Type: beam
Expert Description: caption_obj
Observed Artifacts: artifacts
Current Reconstruction Parameters:
param_info

Based on this sample information, what specific parameter adjustments are needed to improve image quality?

POSSIBLE PARAMETER UPDATES:
A) Change number of batches to 1
B) Disable momentum acceleration
C) Disable multimodal update
D) Disable position correction
E) Enable momentum acceleration
F) Enable multimodal update
G) Enable position correction
H) Increase batch size
I) Increase number of OPR modes
J) Increase number of probe modes
K) Increase the number of iterations
L) No changes needed
M) Recenter diffraction patterns
N) Reduce batch size
O) Reduce diffraction pattern size by factor of 2
P) Reduce or disable regularization
Q) Try other diffraction pattern orientations
R) Turn off affine constraint
S) Turn off variable probe correction (set OPR modes to 0)
T) Turn on variable probe correction (set OPR modes to 1)
U) Use Gaussian noise model
V) Use compact batch selection scheme
W) Use multislice model
X) Use sparse batch selection scheme

Your task:
1. Select all correct options (A, B, C, ...) and list them as a comma-separated list inside <answer></answer> tags (e.g., <answer>E, J, V</answer>).
2. The <answer></answer> tags must contain ONLY the letters. Do not include any explanations or other text.

Ground-Truth Generation: The ground-truth labels for the assistant response were generated via a complex procedure. The expert's original free-text advice (the caption_param field) was programmatically mapped to the corresponding alphabetic choices (A-X) using a curated set of regular expression patterns. This allowed us to convert natural language instructions (e.g., "Increase the number of iterations to 2000") into a standardized, multi-label format (e.g., <answer>K</answer>).

A.3.2 Task 2: VLM-based Artifact Detection

This task is formulated as a multi-modal, multi-label classification problem where a VLM identifies visual artifacts in a ptychographic image.

Prompt Template: The user prompt consists of an image paired with the following text instruction:

Artifact Detection Prompt

You are an expert in X-ray ptychography image analysis. What artifacts are visible?

Possible artifacts include: Grid artifacts, Halo artifacts, Line artifacts, Local distortion, Low contrast, No clear structure, Noisy artifacts, Non-uniform background, Phase ramp, Singularity artifacts, Blurring.

Format your response as: <answer>[Comma-separated list of artifacts, or "No obvious artifacts" if the image quality is good]</answer> Provide your answers now:

Ground-Truth Generation: The assistant response in the training data contains the expert-annotated list of artifacts (from the artifact_choice field), formatted within <answer> tags as requested by the prompt.

A.4 Model and Training Details

This section provides the specific identifiers, libraries, and hyperparameters used for all experiments.

A.4.1 Models

All open-weight models were used in their full-precision versions. The specific Hugging Face identifiers used are:

- **VLMs:** meta-llama/Llama-3.2-11B-vision and meta-llama/Llama-3.2-90B-vision.
- **LLMs:** meta-llama/Meta-Llama-3.1-8B-Instruct and meta-llama/Meta-Llama-3.1-70B-Instruct.
- **Library**: Both the LLM and VLM training processes utilized the Unsloth library [23].
- **Proprietary Baseline:** OpenAI's `gpt-4o` model was accessed via their API.
- **Vision Baseline:** The Meta AI `DINOv3` model was used as a fixed feature extractor for the traditional computer vision baseline.

Table 5: **SFT Hyperparameters for Vision-Language Models (VLMs)**. Training configuration used for supervised fine-tuning of Llama 3.2-Vision models on the artifact detection task.

Hyperparameter	Value
SFT Epoch (11B model)	50
SFT Epoch (90B model)	20
`learning_rate`	2e-4
`lora_r`	16
`lora_alpha`	16
`lora_dropout`	0
`bias`	"none"
`optim`	"adamw_8bit"
`weight_decay`	0.01
`max_seq_length`	2048
`per_device_train_batch_size`	1
`gradient_accumulation_steps`	8

Table 6: **SFT Hyperparameters for Language Models (LLMs)**. Training configuration used for supervised fine-tuning of Llama 3.1 models on the parameter recommendation task.

Hyperparameter	Value
num_epochs	50
learning_rate	2e-4
lora_r	16
lora_alpha	16
target_modules	["q_proj", "k_proj", "v_proj", "o_proj", "gate_proj", "up_proj", "down_proj"]
lora_dropout	0
bias	"none"
optim	"adamw_8bit"
weight_decay	0.01
max_seq_length	2048
per_device_train_batch_size	4
gradient_accumulation_steps	8

A.4.2 Supervised Fine-Tuning (SFT) Protocol

We employed parameter-efficient fine-tuning (PEFT) using Low-Rank Adaptation (LoRA). The loss function for our multi-label tasks was Binary Cross-Entropy with Logits Loss. The detailed hyperparameters are provided in Table 5 and Table 6.

A.4.3 In-Context Learning (ICL) Protocol

For the Sample-Specific Few-Shot (SSFS) strategy, examples were selected from the training set based on an exact match of the sample type metadata field with the test sample. If fewer than k examples with a matching sample type were available, the remaining slots in the prompt were filled by randomly sampling from the rest of the training set, ensuring the test sample itself was excluded.

A.5 Results

A.5.1 Comprehensive Results with Statistical Validation

The main paper presents the mean and standard deviation for our experimental results in Tables 1 and 2. To provide a complete picture of statistical significance and address the variability from a limited test set, this section provides the full 95% confidence intervals (CIs) for all VLM and LLM experiments. These CIs were computed via bootstrapping with 10,000 resamples. The tables allow for a direct comparison of the performance ranges between different models and strategies, directly supporting the statistical conclusions drawn in the main text.

Table 7: VLM Performance on Artifact Detection (95% Confidence Interval)

Model	Fewshot Strategy	0-shot[2]	1-shot	3-shot	5-shot	7-shot
SFT Llama 3.2-Vision 11B	RFS	[0.412 - 0.573]	[0.251 - 0.392] ↓	[0.239 - 0.399]	[0.308 - 0.468]	[0.315 - 0.475]
	SSFS		**[0.458 - 0.631]**	**[0.547 - 0.720]**	**[0.581 - 0.748]**	**[0.636 - 0.784]**
Base Llama 3.2-Vision 11B	RFS	[0.168 - 0.266]	[0.188 - 0.320]	[0.183 - 0.321]	[0.248 - 0.377]	[0.270 - 0.412]
	SSFS		**[0.512 - 0.696]**	**[0.535 - 0.715]**	**[0.610 - 0.773]**	**[0.615 - 0.769]**
SFT Llama 3.2-Vision 90B	RFS	[0.348 - 0.509]	[0.264 - 0.403] ↓	[0.337 - 0.479]	[0.326 - 0.473]	[0.377 - 0.516]
	SSFS		**[0.500 - 0.668]**	**[0.504 - 0.683]**	**[0.589 - 0.749]**	**[0.655 - 0.796]**
Base Llama 3.2-Vision 90B	RFS	[0.000 - 0.050]	[0.189 - 0.322]	[0.109 - 0.248]	[0.040 - 0.147]	[0.083 - 0.229]
	SSFS		**[0.514 - 0.700]**	**[0.525 - 0.723]**	**[0.525 - 0.713]**	**[0.626 - 0.779]**
GPT-4o	RFS	[0.126 - 0.226]	[0.117 - 0.209]	[0.137 - 0.242]	[0.189 - 0.308]	[0.206 - 0.357]
	SSFS		**[0.262 - 0.410]**	**[0.441 - 0.614]**	**[0.546 - 0.720]**	**[0.571 - 0.739]**

Table 8: LLM Performance on Parameter Recommendation (95% Confidence Intervals)

Model	Fewshot Strategy	0-shot	1-shot	3-shot	5-shot	7-shot
SFT Llama 3.1 8B	RFS	[0.370 - 0.572]	[0.247 - 0.389] ↓	[0.328 - 0.502]	[0.394 - 0.598]	[0.477 - 0.630]
	SSFS		**[0.225 - 0.417]**	**[0.479 - 0.722]**	**[0.549 - 0.770]**	**[0.587 - 0.797]**
Base Llama 3.1 8B	RFS	[0.051 - 0.136]	[0.114 - 0.258]	[0.205 - 0.337]	[0.359 - 0.546]	[0.297 - 0.481]
	SSFS		**[0.337 - 0.483]**	**[0.491 - 0.715]**	**[0.533 - 0.792]**	**[0.593 - 0.814]**
SFT Llama 3.1 70B	RFS	[0.734 - 0.928]	[0.364 - 0.545] ↓	[0.411 - 0.573]	[0.427 - 0.621]	[0.496 - 0.689]
	SSFS		**[0.437 - 0.599]**	**[0.508 - 0.703]**	**[0.593 - 0.764]**	**[0.597 - 0.781]**
Base Llama 3.1 70B	RFS	[0.189 - 0.266]	[0.193 - 0.343]	[0.273 - 0.452]	[0.436 - 0.652]	[0.463 - 0.684]
	SSFS		**[0.446 - 0.634]**	**[0.664 - 0.820]**	**[0.697 - 0.861]**	**[0.755 - 0.923]**
GPT-4o	RFS	[0.255 - 0.373]	[0.269 - 0.418]	[0.240 - 0.415]	[0.423 - 0.613]	[0.456 - 0.650]
	SSFS		**[0.521 - 0.709]**	**[0.614 - 0.796]**	**[0.671 - 0.851]**	**[0.701 - 0.854]**

[2]0-shot CIs from SSFS and RFS bootstrap runs differed negligibly; for simplicity, a single representative interval is reported.

A.5.2 Per-Class F1-Score for Artifact Detection (SFT-90B + SSFS)

To provide a more granular analysis of model performance, especially given the inherent class imbalance of the dataset, we provide a per-class F1-score analysis for our best-performing visual model (SFT-90B with SSFS) in Table 9. The results show that the model's performance generally improves with more in-context examples (k) and is strongest on the most well-represented classes (e.g., "Halo artifacts," "Low contrast"). The model struggles with extremely rare classes (e.g., "No clear structure," "Phase ramp"), which have only one test sample. This imbalance is the primary driver for the gap between the Micro-F1 and Macro-F1 scores reported in the main paper.

Table 9: Per-Class F1-Scores for SFT-90B (SSFS) on Artifact Detection. N indicates the number of test samples (out of 79) that contain the artifact.

Artifact Class	**N**	**0-shot F1**	**1-shot F1**	**3-shot F1**	**5-shot F1**	**7-shot F1**
Halo artifacts	44	0.47	0.72	0.72	0.81	0.90
Line artifacts	34	0.53	0.64	0.64	0.60	0.67
Low contrast	25	0.52	0.68	0.67	0.84	0.88
Grid artifacts	21	0.40	0.48	0.50	0.56	0.67
Non-uniform background	17	0.49	0.64	0.60	0.72	0.75
Singularity artifacts	11	0.40	0.58	0.62	0.59	0.64
Noisy artifacts	8	0.29	0.38	0.46	0.80	0.63
Local distortion	5	0.00	0.00	0.00	0.22	0.00
No obvious artifacts	4	0.00	0.00	0.00	0.00	0.00
Blurring	2	0.00	0.00	0.00	0.00	0.00
No clear structure	1	0.00	0.00	0.00	0.00	0.00
Phase ramp	1	0.00	0.00	0.00	0.00	0.00
Overall (Micro F1)	**173**	**0.43**	**0.59**	**0.60**	**0.67**	**0.73**

A.5.3 Per-Sample-Type Performance on Parameter Recommendation

While the preceding analysis addresses class imbalance, this section provides a scientific breakdown of performance by material category (sample_type). We analyzed the best-performing LLM (Base Llama 3.1-70B with SSFS) to understand which ptychographic samples are more challenging for the model. The performance, as shown in Table 10, is highly dependent on the material being analyzed.

Table 10: Per-Sample-Type Micro-F1 Scores for Base Llama 3.1-70B with SSFS on the Parameter Recommendation task. N indicates the number of test samples for each type.

Sample Type	**N**	**0-shot F1**	**1-shot F1**	**3-shot F1**	**5-shot F1**	**7-shot F1**
Ecoli in starch	9	0.301	0.771	0.966	0.941	0.978
TiO2 in capillary	9	0.233	0.654	0.800	0.783	0.973
Siemens star test pattern	4	0.320	0.480	0.889	0.960	0.923
XRnanotech 3D test pattern	7	0.213	0.278	0.462	0.571	0.684
Supercrystal	6	0.108	0.222	0.429	0.417	0.500
Liquid-metal particles	3	0.250	0.714	0.889	1.000	1.000
Integrated circuit	2	0.167	0.000	0.286	0.667	0.889
NCM battery	1	0.250	0.750	0.750	1.000	1.000
Bilayer WSe2	1	0.200	0.500	0.333	0.000	0.000
CuS battery	1	0.000	0.000	0.500	0.667	0.500
LNO	1	0.000	0.000	0.000	0.000	0.000
Overall (Micro F1)	**44**	**0.229**	**0.544**	**0.747**	**0.785**	**0.848**

The results in Table 10 show a strong correlation between the number of available test samples (N) and model performance, particularly for sample types with very few instances. The model consistently achieves near-perfect scores on types with many examples (e.g., Ecoli in starch) but struggles on types with only a single test sample (e.g., LNO, CuS battery). This suggests that while ICL is highly effective, its performance is still sensitive to the diversity and representation of the underlying data distribution, which is a key area for future work in dataset augmentation and expansion.