OpenGVL - Benchmarking Visual Temporal Progress for Data Curation

Paweł Budzianowski[1] **Emilia Wiśnios** **Gracjan Góral**[1]
Igor Kulakov[3] **Viktor Petrenko**[3] **Krzysztof Walas**[2,4]
[1]University of Warsaw [2]IDEAS Research Institute
[3]Simple Automation [4]Poznań University of Technology

Abstract: Data scarcity remains one of the most limiting factors in driving progress in robotics. However, the amount of available robotics data in the wild is growing exponentially, creating new opportunities for large-scale data utilization. Reliable temporal task completion prediction could help automatically annotate and curate this data at scale. The Generative Value Learning (GVL) approach was recently proposed, leveraging the knowledge embedded in vision-language models (VLMs) to predict task progress from visual observations. Building upon GVL, we propose OpenGVL, a comprehensive benchmark for estimating task progress across diverse challenging manipulation tasks involving both robotic and human embodiments. We evaluate the capabilities of publicly available open-source foundation models, showing that open-source model families significantly underperform closed-source counterparts, achieving only approximately 70% of their performance on temporal progress prediction tasks. Furthermore, we demonstrate how OpenGVL can serve as a practical tool for automated data curation and filtering, enabling efficient quality assessment of large-scale robotics datasets. We release the benchmark along with the complete codebase.

 : https://huggingface.co/spaces/OpenGVL/OpenGVL

 : https://github.com/budzianowski/opengvl

1 Introduction

Advancements in hardware and modeling have accelerated progress in robotics. Various embodiments have recently been proposed with decreasing bill-of-material costs, leading to wider availability [1, 2, 3]. A variety of Vision-Language-Action (VLA) models are being created and open-sourced [4, 5, 6]. Furthermore, new benchmarks, repositories, and communities are being formed [7, 8]. These hardware innovations have led to different data collection approaches such as UMI and DexHub [9, 10]. However, this rapid progress is not matched by the availability of well-curated datasets. There are only a few large-scale datasets available, such as Agibot-World, OXE, and Droid [11, 12, 13]. Although these datasets are much larger than previously available ones, they remain an order of magnitude smaller than datasets used in vision or language domains [14, 15].

However, reduced entry barriers have led to wider adoption of different data collection methods and an increased propensity to share data. As of August 2025, more than 2.6 million episodes were publicly shared on Hugging Face's Dataset Hub alone.[1] This calls for building tools that allow efficient and cost-effective filtering of available data. Temporal prediction progress (general purpose reward functions) determines robots' own proficiency at the specified task from their own

[1]It is important to note that some datasets are copies of previously available datasets.

9th Conference on Robot Learning (CoRL 2025), Seoul, Korea.

observations [16, 17, 18]. Such ability can be repurposed to curate and filter already collected datasets [19].

Recently, Ma et al. [20] proposed a generative value function estimator (GVL) that leverages world knowledge embedded in VLMs to predict universal value functions and estimate task progress. To automatically measure episode or dataset quality, Ma et al. [20] introduced the Value-Order Correlation (VOC) metric, which exhibits useful characteristics for data curation applications.

Building on this foundation and motivated by the need for large-scale robotics datasets comparable to The Pile or C4 [14, 21], we develop an open-source temporal progress prediction system (OpenGVL) as a foundational tool for data management at scale. We replicate and extend the GVL approach for open-source models, creating the OpenGVL benchmark. Furthermore, we demonstrate how OpenGVL can serve as a practical tool for real-world data curation applications.

OpenGVL reveals significant performance gaps between open-source and proprietary models. It can also highlight different patterns per episode as well as across full datasets. Specifically, our contributions are threefold: 1) We create and release the OpenGVL Benchmark with accompanying code. 2) We analyze popular open-source VLMs, highlighting the performance gap compared to proprietary counterparts. 3) We demonstrate practical methods for how OpenGVL can curate large-scale open-source datasets.

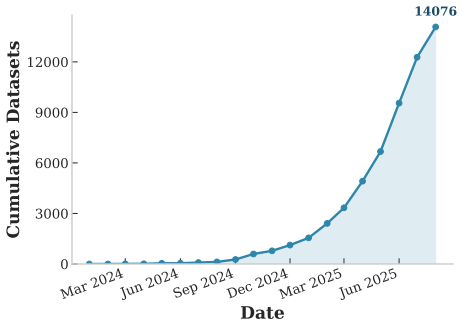

Figure 1: Cumulative number of shared datasets for the LeRobot tag on the HF Datasets Hub.

2 Related work

Learnable universal value functions and success detectors for robotics have been a long-standing challenge [22, 23, 17, 18, 24]. To formalize this concept, given a trajectory $\tau = (o_1, \ldots, o_T)$ with observations o_t, a value function $V : T \to \mathbb{R}$ assigns a scalar score to the entire trajectory, reflecting how well it achieves its goal. We define:

$$V(\tau) = \mathbb{E}\left[\sum_{t=1}^{T} \gamma^{t-1} r_t \,\middle|\, \tau\right],$$

where r_t denotes the task success signal at step t and $\gamma \in [0, 1]$ is a discount factor. In practice, $V(\tau)$ serves as a *temporal success measure* that can be approximated from visual-language inputs. Initial works focused on sparse signals of task success and typically relied on training or fine-tuning base models for specific tasks or domains [24, 22]. Alakuijala et al. [25] and Zhang et al. [26] fine-tune a VLM with a sequential ranking objective to encourage later frames in the video to have higher rewards.

Recently, Ma et al. [20] proposed deriving fine-grained temporal success predictions through in-context learning. Leveraging advancements in VLMs, this approach naturally frames the problem as a trajectory, goal, and prediction setup where traditional training can be replaced with few-shot learning. In this setup, the VLM is provided with a few examples of trajectories along with their temporal value function progress and a trajectory to be evaluated. Ma et al. [20] showed that due to VLMs' propensity for imitating behavioral patterns in context, shuffling input observations improves prediction quality.

To automatically measure prediction quality, GVL uses a rank correlation (Spearman or Kendall) between the predicted values and the temporal order of frames in the trajectory (Value-Order Correlation, VOC):

$$\text{VOC} = \text{rank-correlation}\big(\text{argsort}(v_1, \ldots, v_T), (1, 2, \ldots, T)\big).$$

where $v_1, \ldots, v_T$ are shuffled frames from the trajectory. VOC ranges from a perfect inverse correlation of -1 to a perfect alignment of 1. The proposed metric was shown to be effective for assessing data quality across different embodiments and human videos. Although the high score itself is a necessary but not sufficient condition, it provides a good signal of data quality.

3 OpenGVL Benchmark

Given the rapid increase in datasets shared online (see Figure 1), we introduce the OpenGVL benchmark to handle data curation needs. OpenGVL replicates the original GVL results with closed-source models while adding comparisons to open-source variants. Furthermore, we show how our benchmark can be easily used for data annotation and filtering in practice.

3.1 Experimental Setup

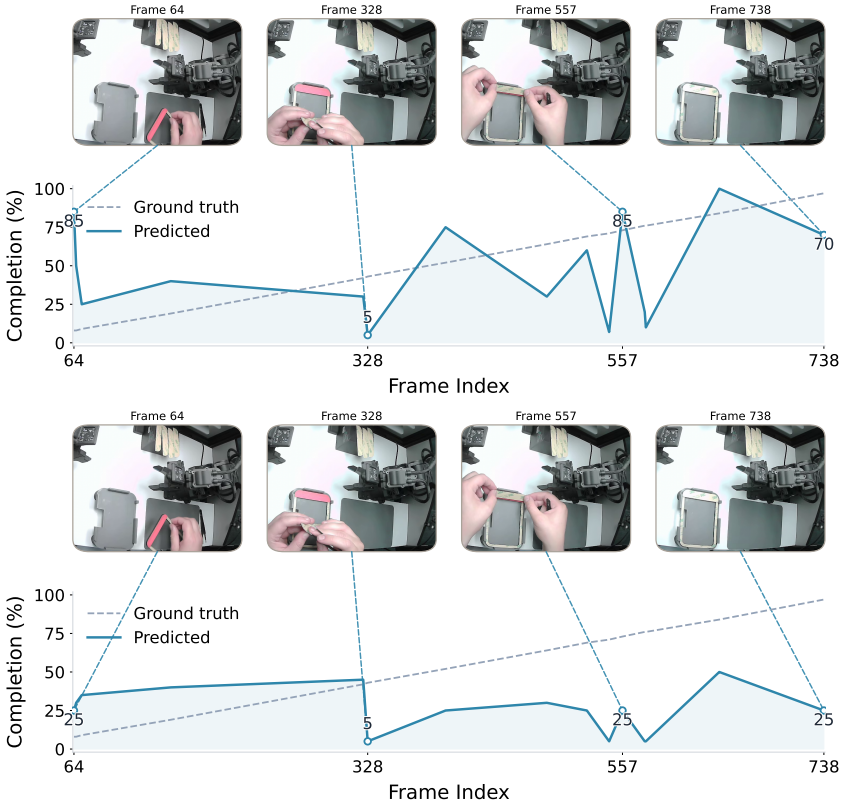

Figure 2: Trajectory prediction performance comparison on the hidden human task. Models were tasked with predicting task completion percentages from shuffled trajectory inputs. The predicted scores were then sorted by ground truth values for visualization. Top: Gemini-2.5-Pro shows signs of monotonic upward trend. Bottom: Gemma-3-27B-it shows minimal predictive alignment indicating difficulty in discerning task completion patterns from visual trajectory data.

Data Selection: We have targeted four initial datasets as a base for validation: `nyu_door`, `berkeley_mvp`, `cmu_stretch`, and `nyu_franka` [27, 11, 28]. These datasets represent diverse manipulation tasks spanning different robot embodiments with relatively low task complexity. From each dataset, we sampled 50 episodes using the same episode indices. We consider two conditioning scenarios: zero-shot and two-shot, balancing open-source context capabilities with performance gains from additional episode examples [20]. Due to the limits on context length, for each episode we sample 15 random frames and shuffle both context and evaluation frames to provide equal context for all models. In the Appendix A, we share the full prompt used across all runs.

To establish the OpenGVL benchmark, we created two hidden datasets to prevent contamination. These datasets are derived from real-world applications requiring long-horizon planning and dexterous manipulation abilities. Figure 2 illustrates prediction results compared to ground truths for both the MiMo-VL-7B-RL-2508 and Gemma-3-4b-it models on the hidden task.

Model Selection: We evaluate a comprehensive set of open-source VLMs spanning different parameter scales and architectural approaches. Our selection includes the Gemma-3 family [29], which provides models at 4B, 12B, and 27B parameter scales, and the Qwen2.5-VL-Instruct family [30] with 3B, 7B, and 32B parameter counts. Both families allow us to study the effect of model scaling on temporal progress prediction. Additionally, we include four models with integrated reasoning capabilities: GLM-4.1V-9B-Thinking [31] with 9B parameters, MiMo-VL-7B-RL-2508 [32] and Cosmos-Reason1-7B [33] with 7B parameters, and Kimi-VL-A3B [34] with 16B total parameters (3B active parameters), all of which incorporate thinking mechanisms that enable enhanced temporal reasoning through explicit reasoning steps. All selected models follow similar architectural paradigms with integrated vision and language encoders, enabling direct comparison of their temporal reasoning capabilities.

For comparison with proprietary models, we have also evaluated `gpt-4o` [35], `gemini-2.5-flash-lite-preview-06-17`, and `gemini-2.5-pro` [36] based on their context capabilities and previous performance. Since closed-source models are updated regularly, we initially tested their performance on unshuffled trajectories. We observed similar behavior to the versions evaluated in [20], confirming that these models tend to over-rely on temporal ordering cues in the provided context. Therefore, we evaluate all subsequent results using shuffled frames.

3.2 Benchmarking Open Source VLMs for GVL

Table 1 presents results for all models in the initial benchmark release under zero-shot and two-shot conditioning. The results show that the VLM scale improves temporal score quality. Both the largest Qwen and Gemma versions achieve similar scores with significant improvements over their smaller counterparts, while among reasoning models, MiMo-VL-7B-RL-2508 shows strong performance and GLM-4.1V-9B-Thinking demonstrates solid results, though Kimi-VL-A3B falls short despite good performance on other vision benchmarks [34, 31].

Model	**Size**	**nyu_door**		**berkeley_mvp**		**cmu_stretch**		**nyu_franka**	
		0-shot	2-shot	0-shot	2-shot	0-shot	2-shot	0-shot	2-shot
Open-source models									
Gemma-3-4b-it	4B	0.0213	0.0521	-0.0176	-0.0352	-0.0461	0.0304	-0.0430	-0.0177
Gemma-3-12b-it	12B	0.5206	0.4304	0.1805	0.1260	0.0045	0.0458	-0.0427	0.0477
Gemma-3-27b-it	27B	0.6372	0.8219	0.1427	0.1575	0.0963	0.1419	0.0226	0.0950
Kimi-VL-A3B	16B	0.2545	0.1605	0.0528	0.0148	-0.0059	-0.0089	-0.0122	0.0417
GLM-4.1V-9B-Thinking	9B	0.6420	0.6540	0.4276	0.3424	0.1628	0.0867	0.1025	0.1392
Qwen2.5-VL-3B-Instruct	3B	-0.0014	0.0097	-0.0112	-0.0232	0.0005	-0.0152	-0.0159	-0.0094
Qwen2.5-VL-7B-Instruct	7B	0.0843	0.1444	0.0500	0.0710	-0.0495	0.0061	0.0181	0.0167
Qwen2.5-VL-32B-Instruct	32B	0.5296	0.6092	0.2491	0.2426	0.0345	0.1192	0.0196	0.1370
MiMo-VL-7B-RL-2508	9B	0.5314	0.5977	0.4391	0.4736	0.2340	0.1798	-0.0544	0.1413
Cosmos-Reason1-7B	7B	0.1703	0.0359	0.0264	0.0208	-0.0429	-0.0233	0.0148	0.0376
Closed-source models									
gpt-4o	–	0.720	0.870	0.410	0.420	0.200	0.200	0.527	0.290
Gemini-2.5-Flash-lite	–	0.8119	0.8491	0.4767	0.6298	0.1500	0.3866	0.1609	0.2679
Gemini-2.5-Pro	–	**0.9158**	**0.9654**	**0.5626**	**0.6806**	**0.3348**	**0.4427**	**0.4065**	**0.4099**

Table 1: VOC scores across different datasets and model sizes in a zero-shot and two-shot context conditioning. VOC is averaged over 50 episodes. We can clearly see that VOC scores improve with model size, observable for both the Gemma family and Qwen models, demonstrating the effect of model scaling on temporal progress prediction.

Moreover, open-source counterparts reach only approximately 60–70% of the performance of proprietary models' upper bound scores[2]. This is a substantial gap compared to the smaller performance differences typically observed in text-only models. This finding demonstrates the importance of comprehensive VLM evaluation suites focused on robotics tasks [37] and highlights the much-needed progress in vision-language tasks requiring spatial reasoning.

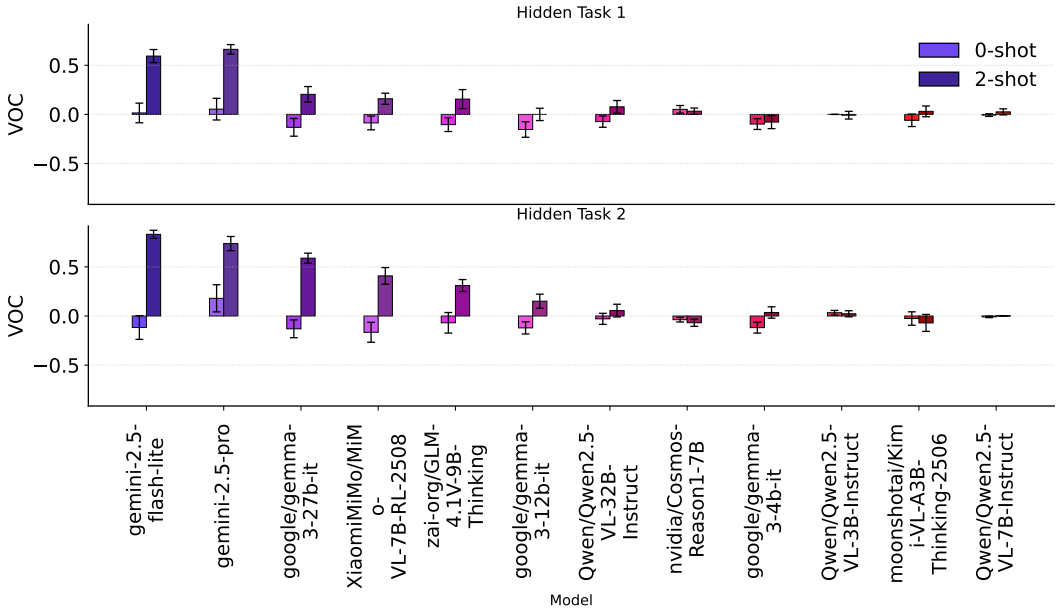

Figure 3: In hidden tasks 1 and 2, zero-shot VOC clusters performed at or below chance levels, indicating poor cold-start grounding capabilities. While two-shot prompting generally improved VOC scores, many remained weak (approximately 0.1–0.3), with only a minority achieving moderate performance (≥ 0.4) and very few reaching strong performance levels (≥ 0.7). This suggests that these tasks remain challenging overall, and while few-shot prompting provides some benefit, it is often insufficient on its own to achieve robust performance.

Motivated by practical applications, we developed two additional evaluation datasets involving last-mile electronic assembly—a multi-step process requiring sub-millimeter precision. Both datasets address the same task: one features human execution while the other uses two 7-DOF robotic arms (see Figure 3). To prevent data contamination, we withhold all task-related data and conduct evaluations for each new benchmark submission. These challenging datasets serve as a stress test for future model capabilities and will become increasingly relevant as VLMs improve their fine-grained spatial reasoning abilities.

3.3 OpenGVL Benchmark Space

To further promote temporal progress scoring as a benchmark for VLM evaluation and data curation, we have created a Hugging Face Space enabling community contributions of new models and datasets for evaluation (see Figure 4). The OpenGVL Benchmark and interactive evaluation interface are publicly available at link. We also provide the complete codebase with all experimental results at link.

4 Data curation in the wild

To demonstrate the potential of OpenGVL for data curation, we analyzed various datasets recently shared on the Hugging Face LeRobot datasets hub. With over 13,000 datasets already published,

[2]The upper bound scores themselves are only a proxy to the (unobservable) ground truth.

automatic data curation and filtering have become essential for leveraging these datasets during the pre-training phase.

We show how OpenGVL can easily identify different dataset issues ranging from unclear task definitions and instructions to occluded sensors and failed/out-of-distribution examples that can disrupt training, as observed previously [20].

VLMs have already been employed to detect incorrect task instructions, one of the most pressing challenges since these often include ambiguous placeholders [6]. For example, SmolVLA sampled representative frames and provided them to the VLM alongside the original instructions. The VLM was prompted to produce a short, action-oriented sentence summarizing the behavior. We demonstrate that OpenGVL adds new dimensions to data curation by enabling effective filtering of problematic episodes or even entire datasets with ill-posed setups. We have identified three common issues with publicly shared data: **(1)** task definition problems, **(2)** labeling ambiguity, and **(3)** failed/out-of-distribution examples. In the following sections, we provide a detailed analysis of each category. We emphasize that we do not critique any specific submissions but rather highlight these challenges to improve future data collection efforts.

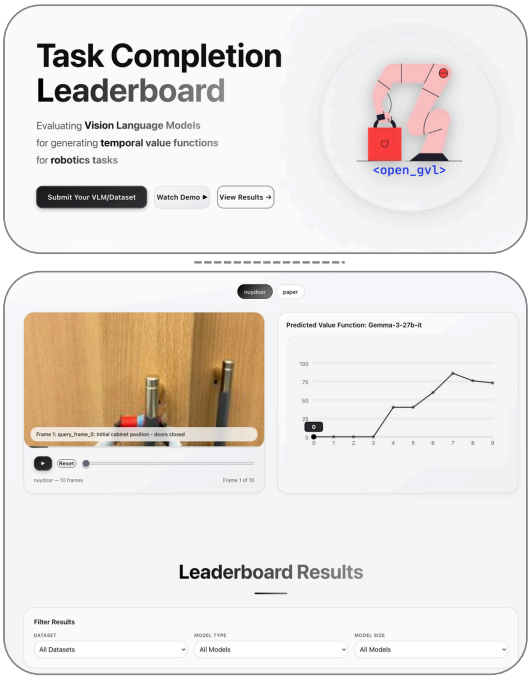

Figure 4: OpenGVL Benchmark Space and interactive analysis of different models and datasets.

4.1 Task definition

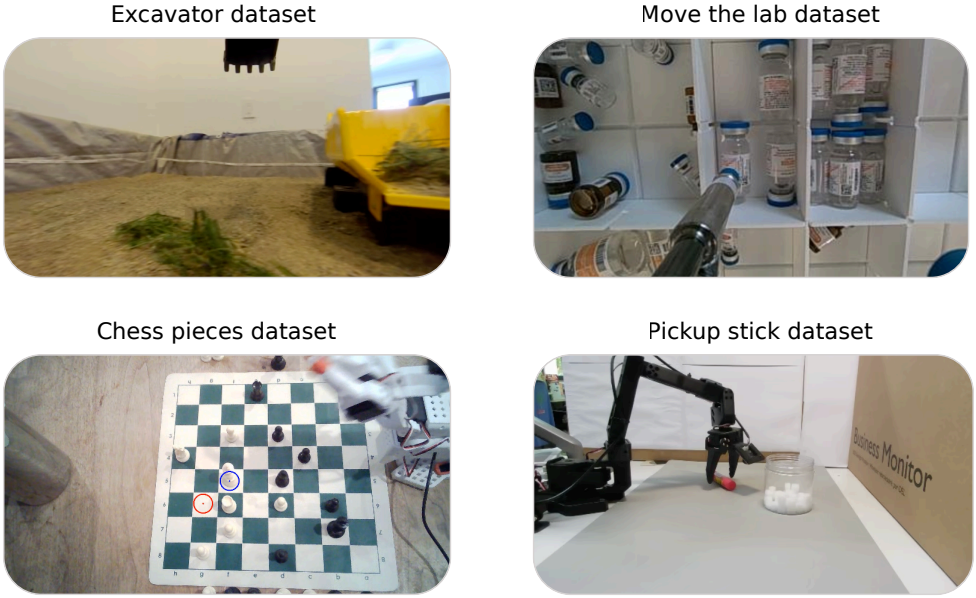

Figure 5: Example of different datasets published by the community and analyzed in Section 4.

The most common type of problem relates to the definition of the task itself. An example of an unclearly defined task can be shown with the dataset Mahimana/excavator_toy_v3_dig_dump_v3_51, where the instruction is "Dig grass and dump in dump truck". Due to ambiguity in both the instruction and task definition, task completion often decreases when it should consistently increase throughout all episodes. This occurs because it is difficult to define progress when there is no clear definition of what constitutes "a dump" and how much material needs to be excavated. This issue is easily detected by checking the VOC accuracy.

Another interesting example can be seen in dopaul/1500_chess_moves—a large dataset of moving chess pieces from point A (red circle) to point B (blue circle) (see Figure 5). Despite the dataset size, training a performant model poses severe challenges, even though this could be viewed as a relatively simple pick-and-place problem. Analyzing the VOC scores from the dataset shows that the VLM does not understand the task definition well given the current instructions. Furthermore, the camera positioned at the arm angle is often completely obscured by lighting, providing no useful sensor information. This suggests the need for different task instructions or improved visual markers.

4.2 Labeling ambiguity

Another common issue stems from labeling ambiguity and unclear instructions. For example, in the dataset willx0909/pickplace_joint, the VOC score is very low due to highly unclear task instructions ("take out a vial and put it into another pocket"). The movement between pockets can be accomplished in multiple ways and between multiple different pockets, and the model struggles to identify the proper temporal relationship. Without clear task boundaries and success criteria, the VLM cannot establish consistent progress patterns across episodes. This type of data can have a deteriorating effect on training foundation VLA models.

4.3 OOD/Failed examples

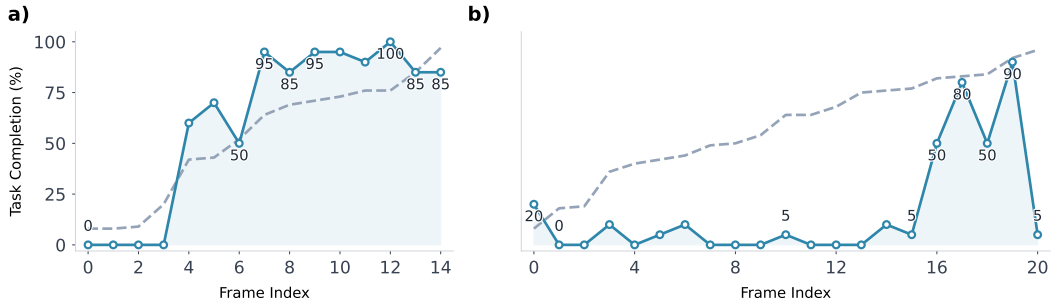

Figure 6: Two examples of task progress trajectories from the Rorschach4153/so101_60_new dataset evaluated by Qwen2.5-VL-32B-Instruct. a) A standard trajectory across all other collected episodes. b) A trajectory from episode 93 that shows a wrong example.

Other common issues can be observed at the individual trajectory level, where some episodes differ significantly from the standard collected data. Comparison between individual scores can easily identify such examples. In the dataset Rorschach4153/so101_60_new, although the overall VOC score is high, it is straightforward to identify patterns of rising and falling task completions that can quickly detect examples falling outside the expected pattern. These outlier trajectories often represent execution failures, sensor malfunctions, or fundamentally different task interpretations that would confuse model training. See Figure 6 for a comparison between a standard and an OOD trajectory. It is worth noting that episode 93 is the only one out of 150 episodes that differs significantly from the rest.

5 Conclusions

In this work, we have presented OpenGVL—an open-source benchmark for evaluating VLMs on temporal task progress prediction for robotics applications. OpenGVL enables rapid validation of different VLMs on the GVL task and facilitates comparisons across models. Additionally, we have demonstrated how open-source VLMs can also be repurposed as a data curation tool that identifies issues at both macro and micro levels within collected datasets. Through qualitative examples, we show how different issues in open-source datasets can be easily detected, paving the way for creating large-scale robotics datasets in the wild. In future work, we plan to investigate how visual goal or failure conditioning could improve prediction quality. The rank correlation metrics could be enhanced by incorporating additional submetrics or explicit Chain-of-Thought processes.

5.1 Limitations

Several aspects of our evaluation could be extended in future work. We tested all models using a temperature setting of 1.0, and it would be valuable to examine how VOC scores vary across different temperature parameters. Additionally, we used a single system prompt template throughout our experiments. As vision-language models are expected to be robust to prompt variations, investigating VOC score sensitivity to different system prompt formulations would strengthen the evaluation framework. Finally, we sampled trajectories uniformly from expert demonstrations. Exploring how VOC performance changes with alternative sampling strategies—such as importance sampling or stratified sampling—could provide deeper insights into model capabilities and evaluation robustness.

Acknowledgments

We gratefully acknowledge the PLGrid (HPC Center: ACK Cyfronet AGH) for providing computer facilities and support within computational grant no. PLG/2025/018552.

References

[1] A. Koch. Low cost arm. 2024. URL https://github.com/AlexanderKoch-Koch/low_cost_robot.

[2] R. S. Team. So-arm 100. 2025. URL https://github.com/TheRobotStudio/SO-ARM100.

[3] C. C. Christoph, M. Eberlein, F. Katsimalis, A. Roberti, A. Sympetheros, M. R. Vogt, D. Liconti, C. Yang, B. G. Cangan, R. J. Hinchet, and R. K. Katzschmann. Orca: An open-source, reliable, cost-effective, anthropomorphic robotic hand for uninterrupted dexterous task learning, 2025. URL https://arxiv.org/abs/2504.04259.

[4] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, S. Jakubczak, T. Jones, L. Ke, S. Levine, A. Li-Bell, M. Mothukuri, S. Nair, K. Pertsch, L. X. Shi, J. Tanner, Q. Vuong, A. Walling, H. Wang, and U. Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2024. URL https://arxiv.org/abs/2410.24164.

[5] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[6] M. Shukor, D. Aubakirova, F. Capuano, P. Kooijmans, S. Palma, A. Zouitine, M. Aractingi, C. Pascal, M. Russi, A. Marafioti, et al. Smolvla: A vision-language-action model for affordable and efficient robotics. *arXiv preprint arXiv:2506.01844*, 2025.

[7] R. Cadene, S. Alibert, A. Soare, Q. Gallouedec, A. Zouitine, S. Palma, P. Kooijmans, M. Aractingi, M. Shukor, D. Aubakirova, M. Russi, F. Capuano, C. Pascal, J. Choghari, J. Moss, and T. Wolf. Lerobot: State-of-the-art machine learning for real-world robotics in pytorch. https://github.com/huggingface/lerobot, 2024.

[8] K.-S. Labs. Rl training library for humanoid locomotion and manipulation. 2024. URL github.com/kscalelabs/ksim.

[9] Y. Park, J. S. Bhatia, L. Ankile, and P. Agrawal. Dexhub and dart: Towards internet scale robot data collection, 2024. URL https://arxiv.org/abs/2411.02214.

[10] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots, 2024. URL https://arxiv.org/abs/2402.10329.

[11] O. Team. Open x-embodiment: Robotic learning datasets and rt-x models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

[12] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

[13] Q. Bu, J. Cai, L. Chen, X. Cui, Y. Ding, S. Feng, S. Gao, X. He, X. Huang, S. Jiang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025.

[14] L. Gao, S. Biderman, S. Black, L. Golding, T. Hoppe, C. Foster, J. Phang, H. He, A. Thite, N. Nabeshima, S. Presser, and C. Leahy. The pile: An 800gb dataset of diverse text for language modeling, 2020. URL https://arxiv.org/abs/2101.00027.

[15] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2019. URL https://arxiv.org/abs/1910.10683.

[16] A. S. Chen, S. Nair, and C. Finn. Learning generalizable robotic reward functions from" in-the-wild" human videos. *arXiv preprint arXiv:2103.16817*, 2021.

[17] L. Guan, Y. Zhou, D. Liu, Y. Zha, H. B. Amor, and S. Kambhampati. "task success" is not enough: Investigating the use of video-language models as behavior critics for catching undesirable agent behaviors. *arXiv:2402.04210*, 2024.

[18] Y. J. Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, and A. Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022.

[19] S. Cabi, S. G. Colmenarejo, A. Novikov, K. Konyushkova, S. Reed, R. Jeong, K. Zolna, Y. Aytar, D. Budden, M. Vecerik, O. Sushkov, D. Barker, J. Scholz, M. Denil, N. de Freitas, and Z. Wang. Scaling data-driven robotics with reward sketching and batch reinforcement learning, 2020. URL https://arxiv.org/abs/1909.12200.

[20] Y. J. Ma, J. Hejna, C. Fu, D. Shah, J. Liang, Z. Xu, S. Kirmani, P. Xu, D. Driess, T. Xiao, et al. Vision language models are in-context value learners. In *The Thirteenth International Conference on Learning Representations*, 2024.

[21] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023. URL https://arxiv.org/abs/1910.10683.

[22] Y. Du, K. Konyushkova, M. Denil, A. Raju, J. Landon, F. Hill, N. de Freitas, and S. Cabi. Vision-language models as success detectors. *arXiv preprint arXiv:2303.07280*, 2023.

[23] J. Rocamonde, V. Montesinos, E. Nava, E. Perez, and D. Lindner. Vision-language models are zero-shot reward models for reinforcement learning. *arXiv preprint arXiv:2310.12921*, 2023.

[24] J. Yang, M. S. Mark, B. Vu, A. Sharma, J. Bohg, and C. Finn. Robot fine-tuning made easy: Pre-training rewards and policies for autonomous real-world reinforcement learning, 2023. URL https://arxiv.org/abs/2310.15145.

[25] M. Alakuijala, R. McLean, I. Woungang, N. Farsad, S. Kaski, P. Marttinen, and K. Yuan. Video-language critic: Transferable reward functions for language-conditioned robotics, 2024. URL https://arxiv.org/abs/2405.19988.

[26] J. Zhang, Y. Luo, A. Anwar, S. A. Sontakke, J. J. Lim, J. Thomason, E. Biyik, and J. Zhang. Rewind: Language-guided rewards teach robot policies without new demonstrations, 2025. URL https://arxiv.org/abs/2505.10911.

[27] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell. Real-world robot learning with masked visual pre-training. In *CoRL*, 2022.

[28] S. Bahl, R. Mendonca, L. Chen, U. Jain, and D. Pathak. Affordances from human videos as a versatile representation for robotics, 2023. URL https://arxiv.org/abs/2304.08488.

[29] G. Team, A. Kamath, J. Ferret, S. Pathak, N. Vieillard, R. Merhej, S. Perrin, T. Matejovicova, A. Ramé, M. Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

[30] Qwen, :, A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, H. Lin, J. Yang, J. Tu, J. Zhang, J. Yang, J. Yang, J. Zhou, J. Lin, K. Dang, K. Lu, K. Bao, K. Yang, L. Yu, M. Li, M. Xue, P. Zhang, Q. Zhu, R. Men, R. Lin, T. Li, T. Tang, T. Xia, X. Ren, X. Ren, Y. Fan, Y. Su, Y. Zhang, Y. Wan, Y. Liu, Z. Cui, Z. Zhang, and Z. Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

[31] G. Team. Glm-4.5v and glm-4.1v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning, 2025. URL https://arxiv.org/abs/2507.01006.

[32] C. Team, Z. Yue, Z. Lin, Y. Song, W. Wang, S. Ren, S. Gu, S. Li, P. Li, L. Zhao, L. Li, K. Bao, H. Tian, H. Zhang, G. Wang, D. Zhu, Cici, C. He, B. Ye, B. Shen, Z. Zhang, Z. Jiang, Z. Zheng, Z. Song, Z. Luo, Y. Yu, Y. Wang, Y. Tian, Y. Tu, Y. Yan, Y. Huang, X. Wang, X. Xu, X. Song, X. Zhang, X. Yong, X. Zhang, X. Deng, W. Yang, W. Ma, W. Lv, W. Zhuang, W. Liu, S. Deng, S. Liu, S. Chen, S. Yu, S. Liu, S. Wang, R. Ma, Q. Wang, P. Wang, N. Chen, M. Zhu, K. Zhou, K. Zhou, K. Fang, J. Shi, J. Dong, J. Xiao, J. Xu, H. Liu, H. Xu, H. Qu, H. Zhao, H. Lv, G. Wang, D. Zhang, D. Zhang, D. Zhang, C. Ma, C. Liu, C. Cai, and B. Xia. Mimo-vl technical report, 2025. URL https://arxiv.org/abs/2506.03569.

[33] NVIDIA, :, A. Azzolini, J. Bai, H. Brandon, J. Cao, P. Chattopadhyay, H. Chen, J. Chu, Y. Cui, J. Diamond, Y. Ding, L. Feng, F. Ferroni, R. Govindaraju, J. Gu, S. Gururani, I. E. Hanafi, Z. Hao, J. Huffman, J. Jin, B. Johnson, R. Khan, G. Kurian, E. Lantz, N. Lee, Z. Li, X. Li, M. Liao, T.-Y. Lin, Y.-C. Lin, M.-Y. Liu, X. Lu, A. Luo, A. Mathau, Y. Ni, L. Pavao, W. Ping, D. W. Romero, M. Smelyanskiy, S. Song, L. Tchapmi, A. Z. Wang, B. Wang, H. Wang, F. Wei, J. Xu, Y. Xu, D. Yang, X. Yang, Z. Yang, J. Zhang, X. Zeng, and Z. Zhang. Cosmos-reason1: From physical common sense to embodied reasoning, 2025. URL https://arxiv.org/abs/2503.15558.

[34] K. Team. Kimi-vl technical report, 2025. URL https://arxiv.org/abs/2504.07491.

[35] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[36] G. Team. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. 2025. URL https://storage.googleapis.com/deepmind-media/gemini/gemini_v2_5_report.pdf.

[37] G. R. Team, S. Abeyruwan, J. Ainslie, J.-B. Alayrac, M. G. Arenas, T. Armstrong, A. Balakrishna, R. Baruch, M. Bauza, M. Blokzijl, et al. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025.

A Prompt

The full prompt provided to the VLM for GVL predictions. The same prompt is used for all models and all datasets reported in Table 1.

```
You are an expert roboticist tasked to predict task completion percentages for
    frames of a robot for the task of {instruction}. The task completion
    percentages are between 0 and 100, where 100 corresponds to full task
    completion. We provide several examples of the robot performing the task at
    various stages and their corresponding task completion percentages. Note that
    these frames are in random order, so please pay attention to the individual
    frames when reasoning about task completion percentage.

Initial robot scene:
[Image: eval_episode.starting_frame]
In the initial robot scene, the task completion percentage is 0.

Frame 1:
[Image: context_episode.frames[0]]
Task Completion Percentage: {task_completion:.1f}%

Frame 2:
[Image: context_episode.frames[1]]
Task Completion Percentage: {task_completion:.1f}%

...
(repeated for all context frames)

Now, for the task of {eval_episode.instruction}, output the task completion
    percentage for the following frames that are presented in random order.
For each frame, format your response as follows:
Frame {i}: Description:{}, Task Completion Percentages: {}%

Be rigorous, precise and remember that the task completion percentage is the
    percentage of the task that has been completed.

Remember that the frames are presented in random order.

Frame N:
[Image: eval_episode.frames[0]]
...
Frame N+eval_num:
[Image: eval_episode.frames[eval_num-1]]
```

B Detailed evaluation results

Table 2: Experimental Results (Part 1 of 4)

Model	Dataset	Ctx	VOC Mean	VOC Std	Std Err	Mism.	Empty
gemini-2.5-flash-lite	lerobot/berkeley_mvp	2	0.6298	0.3077	0.0435	0	0
		0	0.4767	0.4505	0.0637	2	0
	lerobot/cmu_stretch	2	0.3866	0.4428	0.0626	0	0
		0	0.1500	0.3966	0.0561	1	3
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.8491	0.2645	0.0374	0	1
		0	0.8119	0.2145	0.0303	0	0
	lerobot/nyu_franka_play_dataset	2	0.2679	0.5169	0.0731	0	0
		0	0.1609	0.5131	0.0726	0	5
	Hidden Task 1	2	0.5936	0.2577	0.0665	0	0
		0	0.0149	0.3870	0.0999	0	0
	Hidden Task 2	2	0.8311	0.1590	0.0411	0	0
		0	−0.1172	0.4669	0.1206	0	0
gemini-2.5-pro	lerobot/berkeley_mvp	2	0.6806	0.4119	0.0582	0	0
		0	0.5626	0.4757	0.0673	0	0
	lerobot/cmu_stretch	2	0.4427	0.4631	0.0655	0	0
		0	0.3348	0.4267	0.0603	0	0
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.9654	0.0335	0.0047	0	0
		0	0.9158	0.1489	0.0211	0	0
	lerobot/nyu_franka_play_dataset	2	0.4099	0.4335	0.0613	0	0
		0	0.4065	0.5030	0.0711	0	0
	Hidden Task 1	2	0.6621	0.1871	0.0483	0	0
		0	0.0534	0.4264	0.1101	0	0
	Hidden Task 2	2	0.7371	0.2789	0.0720	0	0
		0	0.1800	0.5344	0.1380	0	0
google/gemma-3-12b-it	lerobot/berkeley_mvp	2	0.1260	0.3927	0.0555	0	0
		0	0.1805	0.3955	0.0559	0	0
	lerobot/cmu_stretch	2	0.0458	0.2990	0.0423	0	0
		0	0.0045	0.2664	0.0377	0	0
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.4304	0.3593	0.0508	0	0
		0	0.5206	0.3690	0.0522	0	0
	lerobot/nyu_franka_play_dataset	2	0.0477	0.2786	0.0394	0	0
		0	−0.0427	0.3160	0.0447	1	0
	Hidden Task 1	2	0.0000	0.2420	0.0625	1	0
		0	−0.1535	0.3056	0.0789	0	0
	Hidden Task 2	2	0.1507	0.2767	0.0714	1	0
		0	−0.1209	0.2399	0.0619	0	0

Table 3: Experimental Results (Part 2 of 4)

Model	Dataset	Ctx	VOC Mean	VOC Std	Std Err	Mism.	Empty
google/gemma-3-27b-it	lerobot/berkeley_mvp	2	0.1575	0.5174	0.0732	0	0
		0	0.1427	0.5369	0.0759	0	0
	lerobot/cmu_stretch	2	0.1419	0.3146	0.0445	0	0
		0	0.0963	0.3869	0.0547	0	0
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.8219	0.1810	0.0256	0	0
		0	0.6372	0.4286	0.0606	0	0
	lerobot/nyu_franka_play_dataset	2	0.0950	0.4236	0.0599	0	0
		0	0.0226	0.3325	0.0470	0	0
	Hidden Task 1	2	0.2042	0.3064	0.0791	0	0
		0	−0.1317	0.3486	0.0900	0	0
	Hidden Task 2	2	0.5886	0.1969	0.0508	0	0
		0	−0.1303	0.3512	0.0907	0	0
google/gemma-3-4b-it	lerobot/berkeley_mvp	2	−0.0352	0.2310	0.0327	8	0
		0	−0.0176	0.2634	0.0373	1	0
	lerobot/cmu_stretch	2	0.0304	0.2146	0.0303	6	0
		0	−0.0461	0.2459	0.0348	2	0
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.0521	0.2600	0.0368	7	0
		0	0.0213	0.3010	0.0426	0	0
	lerobot/nyu_franka_play_dataset	2	−0.0177	0.2775	0.0392	4	0
		0	−0.0430	0.2334	0.0330	6	0
	Hidden Task 1	2	−0.0781	0.2588	0.0668	1	0
		0	−0.0983	0.2127	0.0549	4	0
	Hidden Task 2	2	0.0359	0.2247	0.0580	2	0
		0	−0.1186	0.2149	0.0555	3	0
moonshotai/Kimi-VL-A3B-Thinking-2506	lerobot/berkeley_mvp	2	0.0148	0.2171	0.0307	16	0
		0	0.0528	0.3598	0.0509	5	0
	lerobot/cmu_stretch	2	−0.0089	0.1920	0.0272	14	0
		0	−0.0059	0.1732	0.0245	5	0
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.1605	0.2757	0.0390	5	0
		0	0.2545	0.3887	0.0550	4	0
	lerobot/nyu_franka_play_dataset	2	0.0417	0.3045	0.0431	0	0
		0	−0.0122	0.2721	0.0385	6	0
	Hidden Task 1	2	0.0310	0.2111	0.0545	2	0
		0	−0.0600	0.2458	0.0635	3	0
	Hidden Task 2	2	−0.0704	0.3339	0.0862	0	0
		0	−0.0255	0.2640	0.0682	2	0

Table 4: Experimental Results (Part 3 of 4)

Model	Dataset	Ctx	VOC Mean	VOC Std	Std Err	Mism.	Empty
nvidia/Cosmos-Reason1-7B	lerobot/berkeley_mvp	2	0.0208	0.1821	0.0258	18	1
		0	0.0264	0.2294	0.0324	19	6
	lerobot/cmu_stretch	2	−0.0233	0.1659	0.0235	22	2
		0	−0.0429	0.2288	0.0324	11	3
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.0359	0.2183	0.0309	18	1
		0	0.1703	0.3069	0.0434	16	1
	lerobot/nyu_franka_play_dataset	2	0.0376	0.2139	0.0303	18	2
		0	0.0148	0.2672	0.0378	10	6
	Hidden Task 1	2	0.0329	0.1230	0.0318	14	0
		0	0.0516	0.1507	0.0389	8	1
	Hidden Task 2	2	−0.0685	0.1413	0.0365	10	0
		0	−0.0375	0.0902	0.0233	7	2
Qwen/Qwen2.5-VL-32B-Instruct	lerobot/berkeley_mvp	2	0.2426	0.4168	0.0589	2	0
		0	0.2491	0.3730	0.0528	15	0
	lerobot/cmu_stretch	2	0.1192	0.2552	0.0361	8	0
		0	0.0345	0.2372	0.0336	17	0
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.6092	0.3823	0.0541	9	0
		0	0.5296	0.3440	0.0486	9	0
	lerobot/nyu_franka_play_dataset	2	0.1370	0.3306	0.0468	2	0
		0	0.0196	0.1889	0.0267	23	0
	Hidden Task 1	2	0.0773	0.2479	0.0640	0	0
		0	−0.0732	0.2231	0.0576	2	0
	Hidden Task 2	2	0.0550	0.2507	0.0647	3	0
		0	−0.0290	0.2183	0.0564	4	0
Qwen/Qwen2.5-VL-3B-Instruct	lerobot/berkeley_mvp	2	−0.0232	0.1262	0.0178	31	2
		0	−0.0112	0.1928	0.0273	27	2
	lerobot/cmu_stretch	2	−0.0152	0.1506	0.0213	29	2
		0	0.0005	0.1005	0.0142	38	1
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.0097	0.1603	0.0227	21	1
		0	−0.0014	0.1148	0.0162	38	0
	lerobot/nyu_franka_play_dataset	2	−0.0094	0.2023	0.0286	18	4
		0	−0.0159	0.0792	0.0112	38	1
	Hidden Task 1	2	−0.0080	0.1518	0.0392	9	2
		0	0.0000	0.0000	0.0000	0	15
	Hidden Task 2	2	0.0228	0.1194	0.0308	10	1
		0	0.0336	0.0896	0.0231	13	0

Table 5: Experimental Results (Part 4 of 4)

Model	Dataset	Ctx	VOC Mean	VOC Std	Std Err	Mism.	Empty
Qwen/Qwen2.5-VL-7B-Instruct	lerobot/berkeley_mvp	2	0.0710	0.2894	0.0409	4	0
		0	0.0500	0.2450	0.0346	10	1
	lerobot/cmu_stretch	2	0.0061	0.2888	0.0408	1	0
		0	−0.0495	0.2316	0.0328	14	1
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.1444	0.2602	0.0368	7	0
		0	0.0843	0.2913	0.0412	12	0
	lerobot/nyu_franka_play_dataset	2	0.0167	0.2594	0.0367	1	0
		0	0.0181	0.2848	0.0403	9	1
	Hidden Task 1	2	0.0259	0.1193	0.0308	7	2
		0	−0.0071	0.0516	0.0133	12	1
	Hidden Task 2	2	0.0019	0.0071	0.0018	9	5
		0	−0.0084	0.0313	0.0081	4	8
XiaomiMiMo/MiMo-VL-7B-RL-2508	lerobot/berkeley_mvp	2	0.4736	0.4028	0.0570	5	0
		0	0.4391	0.4151	0.0587	4	0
	lerobot/cmu_stretch	2	0.1798	0.3495	0.0494	3	0
		0	0.2340	0.3984	0.0563	3	0
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.5977	0.3594	0.0508	7	0
		0	0.5314	0.3889	0.0550	6	0
	lerobot/nyu_franka_play_dataset	2	0.1413	0.4122	0.0583	5	0
		0	−0.0544	0.2882	0.0408	4	0
	Hidden Task 1	2	0.1593	0.2185	0.0564	4	0
		0	−0.0866	0.2726	0.0704	0	0
	Hidden Task 2	2	0.4083	0.3275	0.0846	1	0
		0	−0.1659	0.3936	0.1016	0	0
zai-org/GLM-4.1V-9B-Thinking	lerobot/berkeley_mvp	2	0.3424	0.4266	0.0603	0	0
		0	0.4276	0.5050	0.0714	1	0
	lerobot/cmu_stretch	2	0.0867	0.3595	0.0508	0	0
		0	0.1628	0.3592	0.0508	1	0
	lerobot/nyu_door_opening_surprising_effectiveness	2	0.6540	0.2667	0.0377	0	0
		0	0.6420	0.3068	0.0434	0	0
	lerobot/nyu_franka_play_dataset	2	0.1392	0.3743	0.0529	0	0
		0	0.1025	0.3835	0.0542	0	0
	Hidden Task 1	2	0.1559	0.3746	0.0967	0	0
		0	−0.1038	0.2725	0.0703	0	0
	Hidden Task 2	2	0.3101	0.2359	0.0609	0	0
		0	−0.0692	0.4053	0.1047	0	0