STRAP: Robot Sub-Trajectory Retrieval for Augmented Policy Learning

Anonymous Author(s)
Affiliation
Address
`email`

Abstract:

Robot learning is experiencing a surge in the size, diversity, and complexity of pre-collected datasets, paralleling trends in NLP and computer vision. Many methods treat these datasets as multi-task expert data to train generalist policies. However, while generalist policies improve average performance, they often underperform on individual tasks due to negative transfer, compared to specialist policies. In this work, we advocate for training policies during deployment by non-parametrically retrieving and training models on relevant data at test time, rather than relying on zero-shot pre-trained policies. We show that many robotics tasks share many low-level behaviors and that retrieval at the *"sub"-trajectory* granularity enables significantly improved data utilization, generalization, and robustness in adapting policies to novel problems. In contrast, existing retrieval methods tend to under-utilize the data and miss out on shared cross-task content. Our proposed method, `STRAP`, uses vision foundation models and dynamic time warping to retrieve sub-sequences from large training corpora. `STRAP` outperforms prior retrieval algorithms in both simulated and real-world experiments, scaling to larger datasets and learning robust control policies from minimal real-world demonstrations.

Keywords: DTW, few-shot imitation learning, retrieval, foundation models

1 Introduction

Robot learning has increasingly shifted from manual controller design to data-driven approaches [1, 2]. Especially, end-to-end imitation learning with, *e.g.*, diffusion models [3, 4] and transformers [5], have shown impressive success. However, collecting large amounts of in-domain data remains expensive and impractical, especially in dynamic environments like homes and offices. Multi-task policy learning attempts to generalize across tasks by training on diverse datasets. While this has led to successes in certain domains [6, 7], generalist policies often suffer from negative transfer, resulting in sub-optimal performance on individual tasks. This issue is exacerbated in unseen environments, where zero-shot generalization is difficult, and task-specific fine-tuning is costly.

Non-parametric data retrieval has been explored as a way to mitigate the need for large fine-tuning datasets. Prior work on retrieval-based methods includes ”replaying” past experiences by retrieving based on off-the-shelf models [8, 9, 10], training encoders on the offline dataset [11], or leveraging abstract representation [12, 13, 14]. The key assumption of these methods is that the offline data consists of expert demonstrations collected in the test environment or that intermediate representations can bridge the environment gap, limiting the usage of large multi-task datasets collected in various domains. Retrieval for policy learning tries to mitigate these issues by learning policies from the retrieved data [15, 16, 17]. However, requiring encoders trained on the offline dataset makes them not scale well to the increasing size of the available data while retrieving individual states underutilizes data sharing between tasks in multi-task datasets [18, 19].

Submitted to the 8th Conference on Robot Learning (CoRL 2024). Do not distribute.

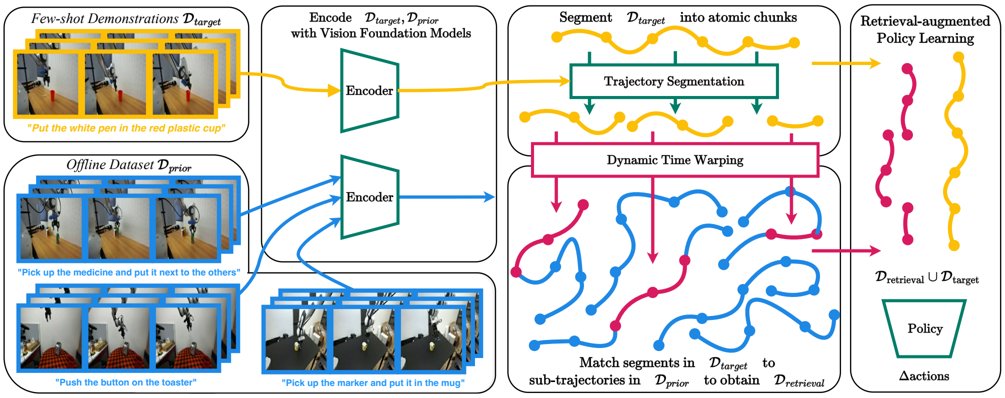

Figure 1: **Overview of** `STRAP`**:** 1) demonstrations $\mathcal{D}_{\text{target}}$ and offline datasets $\mathcal{D}_{\text{prior}}$ are encoded into a shared embedding space using a vision foundation model, 2) automatic trajectory segmentation generates sub-trajectories which 3) S-DTW matches to corresponding sub-trajectories in $\mathcal{D}_{\text{prior}}$ creating $\mathcal{D}_{\text{retrieval}}$, 4) training a policy on $\mathcal{D}_{\text{target}} \cup \mathcal{D}_{\text{retrieval}}$ results in better performance and robustness.

We introduce **S**ub-sequence **T**rajectory **R**etrieval for **A**ugmented **P**olicy Learning (`STRAP`), a novel retrieval method that leverages sub-trajectory similarity, improving test-time generalization by using components of diverse tasks from pre-collected data. Our approach incorporates time-invariant alignment techniques like dynamic time warping [20], enabling the comparison of sub-trajectories of different lengths, further increasing flexibility across tasks and domains. We demonstrate significant gains for few-shot learning on the LIBERO [21] benchmark in simulation, and a challenging Pen-in-Cup task in the real world. Our key insights are as follows:

1. *Vision foundation models* offer powerful out-of-the-box representations for trajectory retrieval. They sufficiently encode scene semantics and offer visual robustness in contrast to brittle in-domain feature extractors from prior work.
2. *Sub-trajectory retrieval* can enable maximal re-use of prior data while capturing temporal information about tasks and dynamics.
3. Performing retrieval via *subsequence dynamic time warping* can find optimal sub-trajectory matches in offline datasets that are agnostic to segment length task horizon or fluctuations in demonstration frequency.

2 `STRAP`: Sub-sequence Robot Trajectory Retrieval for Augmented Policy Training

Retrieval-augmented Policy Learning: We consider a few-shot learning setting where we're given a target dataset $\mathcal{D}_{\text{target}}$ of expert trajectories collected in the test environment and task. This dataset only contains a small set of trajectories, often insufficient to solve the task and limiting generalization. We posit that generalization can be accomplished by non-parametrically *retrieving* data from an offline dataset $\mathcal{D}_{\text{prior}}$ to augment the target dataset $\mathcal{D}_{\text{target}}$. $\mathcal{D}_{\text{prior}}$ can contain data from different environments, scenes, levels of expertise, tasks, or embodiments. Notably, the set of tasks in the offline dataset does *not* need to overlap with the set of tasks in the target dataset but for the scope of this work we assume expert-level trajectories and shared embodiment.

Sub-trajectories for Retrieval: To make the best use of the offline dataset $\mathcal{D}_{\text{prior}}$, while capturing temporal task-specific dynamics, we expand the notion of retrieval from being able to retrieve entire trajectories or single states to retrieving variable-length sub-trajectories. In doing so, retrieval can capture the temporal dynamics of the task, while still being able to share data between seemingly different tasks. Most long-horizon problems observed in robotics datasets [21, 19, 18] naturally contain multiple such sub-trajectories, *e.g.*, picking and placing, or opening and closing. Since $\mathcal{D}_{\text{prior}}$ is usually much larger than $\mathcal{D}_{\text{target}}$, we only require segmenting the $\mathcal{D}_{\text{target}}$ into sub-trajectories and utilize dynamic time warping (DTW) to find corresponding matches in $\mathcal{D}_{\text{prior}}$. While this segmentation can be done manually, we propose an automatic technique for sub-trajectory segmentation in Appendix A.3 that yields promising empirical results.

Vision Foundation Models for Measuring Similarity: Given the segmented sub-trajectories from $\mathcal{D}_{\text{target}}$ and our DTW based matching algorithm, we must define a measure of similarity that allows us to retrieve *relevant* sub-trajectory data from $\mathcal{D}_{\text{prior}}$. While prior work has suggested objectives to train such similarity metrics through representation learning [15, 17, 13], these methods are often trained purely in-domain, making them particularly sensitive to visual appearance, distractors, and irrelevant spurious features. In this work, we will adopt the insight that vision(-language) foundation models [22, 23] offer off-the-shelf solutions to measuring the semantic and visual similarities between sub-trajectories. Their rich representations are robust to the aforementioned variations and naturally capture a notion of object-ness and semantic correspondence. Denoting a vision foundation model as $\mathcal{F}(\cdot)$, we can compute the pairwise distance of two camera views o_i and o_j with an L2 norm in embedding space, *i.e.*, $||\mathcal{F}(o_i) - \mathcal{F}(o_j)||_2$.

Efficient Sub-trajectory Retrieval with S-DTW: In contrast to single states or full trajectories, sub-trajectories may have variable lengths and temporal positioning within a trajectory caused by varying tasks, platforms, or demonstrators. We employ subsequence dynamic time warping (S-DTW), a variant of DTW, to match the target sub-trajectories to appropriate segments in $\mathcal{D}_{\text{prior}}$ (*c.f.* Eq. 23). Since S-DTW doesn't require the start and end points to line up it scales naturally with these challenges and allows for retrieval from diverse, multi-task datasets. To construct our retrieval dataset $\mathcal{D}_{\text{retrieval}}$, we select the K matches with the lowest cost uniformly across the sub-trajectories in $\mathcal{D}_{\text{target}}$, *i.e.*, the same number of matches for each initial sub-trajectory until K matches are retrieved. The training dataset then contains a union of the target dataset $\mathcal{D}_{\text{target}}$ and the retrieved dataset $\mathcal{D}_{\text{retrieval}}$, $\mathcal{D}_{\text{target}} \cup \mathcal{D}_{\text{retrieval}}$. This significantly larger, retrieval-augmented dataset can then be used to learn policies via imitation learning, leading to robust, generalizable policies.

`STRAP`**– Sub-sequence Trajectory Retrieval for Augmented Policy Learning:** We outline the full retrieval and policy-augmented training process in Eq. 1. **1) Encode $\mathcal{D}_{\text{target}}$ and $\mathcal{D}_{\text{prior}}$:** We encode image observations in $\mathcal{D}_{\text{target}}$ and $\mathcal{D}_{\text{prior}}$ using a vision foundation model, *e.g.*, DINOv2 [22] or CLIP [23]. *2) Segment $\mathcal{D}_{target}$ into sub-trajectories:* To best leverage the multi-task trajectories in $\mathcal{D}_{\text{prior}}$, we segment the demonstrations in $\mathcal{D}_{\text{target}}$ into atomic chunks based on a low-level motion heuristic. **3) S-DTW matching of $\mathcal{D}_{\text{target}}$ to $\mathcal{D}_{\text{prior}}$:** We utilize S-DTW to generate matches between chunks in $\mathcal{D}_{\text{target}}$ and $\mathcal{D}_{\text{prior}}$, and construct $\mathcal{D}_{\text{retrieval}}$ by selecting the top K matches uniformly across all chunks. **4) Augmented-policy learning:** Combining $\mathcal{D}_{\text{retrieval}}$ with $\mathcal{D}_{\text{target}}$ forms our dataset for learning a policy. We use language-conditioned behavior cloning (BC) to learn a visuomotor policy similar to Haldar et al. [5], Nasiriany et al. [24]. We choose a transformer-based [25] architecture feeding in a history of the last h observations $s_{t-h:t}$ and predicting a chunk of h future actions using a Gaussian mixture model action head. We sample batches from the union of $\mathcal{D}_{\text{target}}$ and $\mathcal{D}_{\text{retrieval}}$, as in $\mathcal{B} \sim \mathcal{D}_{\text{target}} \cup \mathcal{D}_{\text{retrieval}}$. As proposed by Haldar et al. [5] we compute the mean-squared error multi-step action loss and add an L2 regularization term over the model weights.

3 Experiments and Results

Task Definition: We demonstrate the efficacy of `STRAP` in simulation on the LIBERO benchmark [21], and on a Pen-in-Cup manipulation task with a real world robot arm. (*c.f.* Eq. 10).

- **LIBERO:** We evaluate on 10 long-horizon tasks (Tab. 1 and **??**) (LIBERO-10) which include diverse objects, layouts, and backgrounds. Each task comes with 50 demonstrations from which we select 5 random demonstrations ($\mathcal{D}_{\text{target}}$) in a few-shot imitation learning setting and retrieve data from all LIBERO-90 tasks, which amounts to 4500 total offline demonstrations ($\mathcal{D}_{\text{prior}}$).
- **Franka-Pen-in-Cup:** To demonstrate the efficacy of `STRAP` in a real-world setting, we solve a Pen-In-Cup task using the Franka Emika Panda robot. $\mathcal{D}_{\text{target}}$ contains 3 on-task demonstrations, and $\mathcal{D}_{\text{prior}}$ consists of 100 demonstrations across 10 tasks in the same tabletop environment collected on the DROID [19] hardware setup.

Table 1: **Baselines:** Performance of baselines, ablations and variations of `STRAP` on the LIBERO 10 tasks (Eq. 10). DINOv2 and CLIP features perform similarly, making `STRAP` flexible in the encoder choice. **Bold** indicates best and <u>underline</u> runner-up results.

Task	Stove-Pot	Bowl-Cabinet	Soup-Cheese	Mug-Mug	Book-Caddy
BC	$77.33\% \pm 4.35$	$71.33\% \pm 5.68$	$27.33\% \pm 2.18$	$38.00\% \pm 5.66$	$75.33\% \pm 1.44$
MT	$0.00\% \pm 0.00$	$0.00\% \pm 0.00$	$0.00\% \pm 0.00$	$0.00\% \pm 0.00$	$\mathbf{88.00\% \pm 1.89}$
BR [15]	$80.0\% \pm 1.63$	$72.0\% \pm 7.72$	$26.0\% \pm 5.25$	$40.0\% \pm 8.64$	$16.0\% \pm 1.89$
FR [17]	$76.0\% \pm 6.60$	$54.67\% \pm 11.98$	$24.67\% \pm 8.55$	$29.33\% \pm 1.44$	$52.0\% \pm 5.89$
D-S	$70.67\% \pm 7.85$	$65.33\% \pm 1.96$	$18.0\% \pm 3.40$	$16.0\% \pm 0.94$	$57.33\% \pm 2.88$
D-T	$78.67\% \pm 2.72$	$75.33\% \pm 2.72$	$37.33\% \pm 6.62$	$\mathbf{63.33\% \pm 3.57}$	$79.00\% \pm 4.95$
`STRAP` (CLIP)	$\mathbf{86.00\% \pm 4.10}$	$\underline{90.67\% \pm 2.18}$	$\underline{42.00\% \pm 0.94}$	$54.67\% \pm 3.31$	$83.33\% \pm 3.03$
`STRAP` (DINOv2)	$\underline{85.33\% \pm 2.18}$	$\mathbf{91.33\% \pm 2.18}$	$\mathbf{42.67\% \pm 7.20}$	$\underline{57.33\% \pm 7.68}$	$\underline{85.33\% \pm 2.81}$

Baselines and Ablation: We compare `STRAP` to Behavior Cloning (BC), Multi-task Policy (MT), BehaviorRetrieval (BR), FlowRetrieval (FR) and ablate DINOv2 features in a state-based (D-S) and full-trajectory (D-T) retrieval setting. We refer the reader to Appendix A.1 for implementation details and Appendix A.5 for extensive ablations.

Does *sub-trajectory retrieval* improve performance in few-shot imitation learning? `STRAP` outperforms the retrieval baselines BR and FR on average by $+12.20\%$ and $+12.47\%$ across all 10 tasks (Tab. 1). These results demonstrate the policy's robustness to varying object poses. BC represents a strong baseline on the LIBERO task as the benchmark's difficulty comes from pose variations during evaluation. By memorizing the demonstrations, BC achieves high success rates, outperforming BR and FR by $+4.53\%$ and $+4.80\%$ across all 10 tasks. The multi-task baseline trained on LIBERO-90 struggles to generalize to unseen language instructions, failing on 9/10 tasks, only succeeding on the one with an almost exact match in LIBERO-90 (*c.f.* Tab. 1). To prove that the robustness benefits are not unique to the LIBERO benchmark we perform a real-world evaluation in Tab. sec. 3. While BC and `STRAP` solve the Franka-Pen-in-Cup demonstrated in $\mathcal{D}_{\text{target}}$ (*base*), BC lacks robustness to out-of-distribution (*OOD*) scenarios. The policy replays the trajectories observed in $\mathcal{D}_{\text{target}}$. `STRAP` retrieves relevant sub-trajectories from $\mathcal{D}_{\text{prior}}$, *e.g.*, the robot putting the screwdriver in the cup or picking up pens in various poses. Augmented policy learning then distills this knowledge into a policy, resulting in generalization to an OOD scenario. To investigate the efficacy of sub-trajectories, we compare sub-trajectory retrieval with (`STRAP`) to retrieving full trajectories (D-T) – both using S-DTW – in Tab. 1. We find sub-trajectory retrieval to improve performance by $+4.17\%$ across all 10 tasks. We hypothesize that full trajectories can contain segments irrelevant to the task, effectively hurting performance and matching accuracy.

Pen-in-Cup	*base*		*OOD*	
	Pick	Place	Pick	Place
BC	100%	100%	0%	0%
`STRAP`	100%	90%	**100%**	**100%**

Table 2: **Real-world results:** Franka-Pen-in-Cup task

How effective are the representations from *vision-foundation models* for retrieval? We ablate the choice of foundation model representation in `STRAP` by comparing CLIP, trained through supervised learning on image-text pairs, with DINOv2, trained in a self-supervised fashion on unlabeled images. We don't find any representation to significantly outperform the other with DINOv2 separated from CLIP by only $+0.73\%$ across all 10 tasks. To show the efficacy of vision-foundation models for retrieval, we replace the in-domain feature extractors from prior work (BR, FR) trained on $\mathcal{D}_{\text{prior}}$ with an off-the-shelf DINOv2 encoder model (D-S). Tab. 1 shows the choice of representation to depend on the task with no method outperforming the others on all tasks. Since D-S has no notion of dynamics and task semantics due to single-state retrieval, BR and FR outperform it by $+5.00\%$ and $+4.73\%$, respectively. We highlight that vision foundation models are not trained on $\mathcal{D}_{\text{prior}}$ and scale much better with increasing amounts of trajectory data and on unseen tasks.

Conclusion We introduce `STRAP` as an innovative approach for leveraging visual foundation models in few-shot robotics manipulation, eliminating the need to train on the entire retrieval dataset and allowing it to scale with minimal compute overhead. By focusing on sub-trajectory retrieval using S-DTW, `STRAP` improves data utilization and captures dynamics more effectively.

References

[1] J. Francis, N. Kitamura, F. Labelle, X. Lu, I. Navarro, and J. Oh. Core challenges in embodied vision-language planning. *Journal of Artificial Intelligence Research*, 74:459–515, 2022.

[2] Y. Hu, Q. Xie, V. Jain, J. Francis, J. Patrikar, N. Keetha, S. Kim, Y. Xie, T. Zhang, Z. Zhao, et al. Toward general-purpose robots via foundation models: A survey and meta-analysis. *arXiv preprint arXiv:2312.08782*, 2023.

[3] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*, 2023.

[4] L. Wang, J. Zhao, Y. Du, E. H. Adelson, and R. Tedrake. Poco: Policy composition from and for heterogeneous robot learning. *CoRR*, abs/2402.02511, 2024. doi:10.48550/ARXIV.2402.02511. URL `https://doi.org/10.48550/arXiv.2402.02511`.

[5] S. Haldar, Z. Peng, and L. Pinto. Baku: An efficient transformer for multi-task policy learning. *arXiv preprint arXiv:2406.07539*, 2024.

[6] S. E. Reed, K. Zolna, E. Parisotto, S. G. Colmenarejo, A. Novikov, G. Barth-Maron, M. Gimenez, Y. Sulsky, J. Kay, J. T. Springenberg, T. Eccles, J. Bruce, A. Razavi, A. Edwards, N. Heess, Y. Chen, R. Hadsell, O. Vinyals, M. Bordbar, and N. de Freitas. A generalist agent. *Trans. Mach. Learn. Res.*, 2022, 2022. URL `https://openreview.net/forum?id=1ikK0kHjvj`.

[7] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth, N. J. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, K. Lee, S. Levine, Y. Lu, U. Malla, D. Manjunath, I. Mordatch, O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch, J. Quiambao, K. Rao, M. S. Ryoo, G. Salazar, P. R. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. T. Tran, V. Vanhoucke, S. Vega, Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich. RT-1: robotics transformer for real-world control at scale. In K. E. Bekris, K. Hauser, S. L. Herbert, and J. Yu, editors, *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023. doi:10.15607/RSS.2023.XIX.025. URL `https://doi.org/10.15607/RSS.2023.XIX.025`.

[8] N. Di Palo and E. Johns. Dinobot: Robot manipulation via retrieval and alignment with vision foundation models. *arXiv preprint arXiv:2402.13181*, 2024.

[9] F. Malato, F. Leopold, A. Melnik, and V. Hautamäki. Zero-shot imitation policy via search in demonstration dataset. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7590–7594. IEEE, 2024.

[10] Y. Zhang, W. Yang, and J. Pajarinen. Demobot: Deformable mobile manipulation with vision-based sub-goal retrieval. *arXiv preprint arXiv:2408.15919*, 2024.

[11] J. Pari, N. M. M. Shafiullah, S. P. Arunachalam, and L. Pinto. The surprising effectiveness of representation learning for visual imitation. In *18th Robotics: Science and Systems, RSS 2022*. MIT Press Journals, 2022.

[12] J. Sheikh, A. Melnik, G. C. Nandi, and R. Haschke. Language-conditioned semantic search-based policy for robotic manipulation tasks. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*.

[13] Y. Kuang, J. Ye, H. Geng, J. Mao, C. Deng, L. Guibas, H. Wang, and Y. Wang. Ram: Retrieval-based affordance transfer for generalizable zero-shot robotic manipulation. In *8th Annual Conference on Robot Learning*.

[14] G. Papagiannis, N. Di Palo, P. Vitiello, and E. Johns. R+ x: Retrieval and execution from everyday human videos. In *RSS 2024 Workshop: Data Generation for Robotics*.

[15] M. Du, S. Nair, D. Sadigh, and C. Finn. Behavior retrieval: Few-shot imitation learning by querying unlabeled datasets. *arXiv preprint arXiv:2304.08742*, 2023.

[16] S. Nasiriany, T. Gao, A. Mandlekar, and Y. Zhu. Learning and retrieval from prior data for skill-based imitation learning. In *Conference on Robot Learning*, 2022.

[17] L.-H. Lin, Y. Cui, A. Xie, T. Hua, and D. Sadigh. Flowretrieval: Flow-guided data retrieval for few-shot imitation learning. In *8th Annual Conference on Robot Learning*, 2024.

[18] O. X.-E. Collaboration, A. O'Neill, A. Rehman, A. Gupta, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, A. Tung, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Gupta, A. Wang, A. Kolobov, A. Singh, A. Garg, A. Kembhavi, A. Xie, A. Brohan, A. Raffin, A. Sharma, A. Yavary, A. Jain, A. Balakrishna, A. Wahid, B. Burgess-Limerick, B. Kim, B. Schölkopf, B. Wulfe, B. Ichter, C. Lu, C. Xu, C. Le, C. Finn, C. Wang, C. Xu, C. Chi, C. Huang, C. Chan, C. Agia, C. Pan, C. Fu, C. Devin, D. Xu, D. Morton, D. Driess, D. Chen, D. Pathak, D. Shah, D. Büchler, D. Jayaraman, D. Kalashnikov, D. Sadigh, E. Johns, E. Foster, F. Liu, F. Ceola, F. Xia, F. Zhao, F. V. Frujeri, F. Stulp, G. Zhou, G. S. Sukhatme, G. Salhotra, G. Yan, G. Feng, G. Schiavi, G. Berseth, G. Kahn, G. Yang, G. Wang, H. Su, H.-S. Fang, H. Shi, H. Bao, H. B. Amor, H. I. Christensen, H. Furuta, H. Bharadhwaj, H. Walke, H. Fang, H. Ha, I. Mordatch, I. Radosavovic, I. Leal, J. Liang, J. Abou-Chakra, J. Kim, J. Drake, J. Peters, J. Schneider, J. Hsu, J. Vakil, J. Bohg, J. Bingham, J. Wu, J. Gao, J. Hu, J. Wu, J. Wu, J. Sun, J. Luo, J. Gu, J. Tan, J. Oh, J. Wu, J. Lu, J. Yang, J. Malik, J. Silvério, J. Hejna, J. Booher, J. Tompson, J. Yang, J. Salvador, J. J. Lim, J. Han, K. Wang, K. Rao, K. Pertsch, K. Hausman, K. Go, K. Gopalakrishnan, K. Goldberg, K. Byrne, K. Oslund, K. Kawaharazuka, K. Black, K. Lin, K. Zhang, K. Ehsani, K. Lekkala, K. Ellis, K. Rana, K. Srinivasan, K. Fang, K. P. Singh, K.-H. Zeng, K. Hatch, K. Hsu, L. Itti, L. Y. Chen, L. Pinto, L. Fei-Fei, L. Tan, L. J. Fan, L. Ott, L. Lee, L. Weihs, M. Chen, M. Lepert, M. Memmel, M. Tomizuka, M. Itkina, M. G. Castro, M. Spero, M. Du, M. Ahn, M. C. Yip, M. Zhang, M. Ding, M. Heo, M. K. Srirama, M. Sharma, M. J. Kim, N. Kanazawa, N. Hansen, N. Heess, N. J. Joshi, N. Suenderhauf, N. Liu, N. D. Palo, N. M. M. Shafiullah, O. Mees, O. Kroemer, O. Bastani, P. R. Sanketi, P. T. Miller, P. Yin, P. Wohlhart, P. Xu, P. D. Fagan, P. Mitrano, P. Sermanet, P. Abbeel, P. Sundaresan, Q. Chen, Q. Vuong, R. Rafailov, R. Tian, R. Doshi, R. Mart'in-Mart'in, R. Baijal, R. Scalise, R. Hendrix, R. Lin, R. Qian, R. Zhang, R. Mendonca, R. Shah, R. Hoque, R. Julian, S. Bustamante, S. Kirmani, S. Levine, S. Lin, S. Moore, S. Bahl, S. Dass, S. Sonawani, S. Tulsiani, S. Song, S. Xu, S. Haldar, S. Karamcheti, S. Adebola, S. Guist, S. Nasiriany, S. Schaal, S. Welker, S. Tian, S. Ramamoorthy, S. Dasari, S. Belkhale, S. Park, S. Nair, S. Mirchandani, T. Osa, T. Gupta, T. Harada, T. Matsushima, T. Xiao, T. Kollar, T. Yu, T. Ding, T. Davchev, T. Z. Zhao, T. Armstrong, T. Darrell, T. Chung, V. Jain, V. Kumar, V. Vanhoucke, W. Zhan, W. Zhou, W. Burgard, X. Chen, X. Chen, X. Wang, X. Zhu, X. Geng, X. Liu, X. Liangwei, X. Li, Y. Pang, Y. Lu, Y. J. Ma, Y. Kim, Y. Chebotar, Y. Zhou, Y. Zhu, Y. Wu, Y. Xu, Y. Wang, Y. Bisk, Y. Dou, Y. Cho, Y. Lee, Y. Cui, Y. Cao, Y.-H. Wu, Y. Tang, Y. Zhu, Y. Zhang, Y. Jiang, Y. Li, Y. Li, Y. Iwasawa, Y. Matsuo, Z. Ma, Z. Xu, Z. J. Cui, Z. Zhang, Z. Fu, and Z. Lin. Open X-Embodiment: Robotic learning datasets and RT-X models. `https://arxiv.org/abs/2310.08864`, 2023.

[19] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, P. D. Fagan, J. Hejna, M. Itkina, M. Lepert, Y. J. Ma, P. T. Miller, J. Wu, S. Belkhale, S. Dass, H. Ha, A. Jain, A. Lee, Y. Lee, M. Memmel, S. Park, I. Radosavovic, K. Wang, A. Zhan, K. Black, C. Chi, K. B. Hatch, S. Lin, J. Lu, J. Mercat, A. Rehman, P. R. Sanketi, A. Sharma, C. Simpson, Q. Vuong, H. R. Walke, B. Wulfe, T. Xiao, J. H. Yang, A. Yavary, T. Z. Zhao, C. Agia, R. Baijal, M. G. Castro, D. Chen, Q. Chen, T. Chung, J. Drake, E. P. Foster, J. Gao, D. A. Herrera, M. Heo, K. Hsu, J. Hu, D. Jackson, C. Le, Y. Li, K. Lin, R. Lin, Z. Ma, A. Maddukuri, S. Mirchandani, D. Morton, T. Nguyen, A. O'Neill, R. Scalise, D. Seale, V. Son, S. Tian, E. Tran, A. E. Wang, Y. Wu, A. Xie, J. Yang, P. Yin, Y. Zhang, O. Bastani, G. Berseth, J. Bohg, K. Goldberg, A. Gupta, A. Gupta, D. Jayaraman, J. J. Lim, J. Malik, R. Martín-Martín, S. Ramamoorthy, D. Sadigh, S. Song, J. Wu,

M. C. Yip, Y. Zhu, T. Kollar, S. Levine, and C. Finn. Droid: A large-scale in-the-wild robot manipulation dataset. 2024.

[20] T. Giorgino. Computing and visualizing dynamic time warping alignments in R: The dtw package. *Journal of Statistical Software*, 31(7):1–24, 2009. doi:10.18637/jss.v031.i07.

[21] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36, 2024.

[22] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. HAZIZA, F. Massa, A. El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*.

[23] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[24] S. Nasiriany, A. Maddukuri, L. Zhang, A. Parikh, A. Lo, A. Joshi, A. Mandlekar, and Y. Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. *arXiv preprint arXiv:2406.02523*, 2024.

[25] A. Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[26] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, and D. Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8121–8130, 2022.

[27] G. Team, R. Anil, S. Borgeaud, Y. Wu, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[28] Y. Jiang, E. Z. Liu, B. Eysenbach, J. Z. Kolter, and C. Finn. Learning options via compression. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL `http://papers.nips.cc/paper_files/paper/2022/hash/8567a53e58a9fa4823af356c76ed943c-Abstract-Conference.html`.

A Appendix

Figure 10: **Simulation and real-world tasks:** $\mathcal{D}_{\text{target}}$ tasks from LIBERO-10 and real-world Franka-Pen-in-Cup (top) and retrieval dataset $\mathcal{D}_{\text{prior}}$ (bottom).

A.1 Simulation Experiments

Table 3: **Baselines (sim):** Performance of different methods on LIBERO-10 tasks in simulation

Method	Mug-Microwave	Moka-Moka	Soup-Sauce	Cream-Cheese-Butter	Mug-Pudding
BC	$28.00\% \pm 0.94$	$0.00\% \pm 0.00$	$\mathbf{17.33\% \pm 4.46}$	$26.67\% \pm 4.25$	$18.00\% \pm 2.49$
MT	$0.00\% \pm 0.00$	$0.00\% \pm 0.00$	$0.00\% \pm 0.00$	$0.00\% \pm 0.00$	$0.00\% \pm 0.00$
BR [15]	$28.67\% \pm 3.93$	$0.0\% \pm 0.0$	$13.33\% \pm 3.81$	$\underline{32.0\% \pm 4.32}$	$\mathbf{26.0\% \pm 1.89}$
FR [17]	$27.33\% \pm 1.44$	$0.0\% \pm 0.0$	$11.33\% \pm 3.03$	$\mathbf{41.33\% \pm 5.52}$	$14.67\% \pm 1.09$
D-S	$30.0\% \pm 3.4$	$0.0\% \pm 0.0$	$4.67\% \pm 0.54$	$16.0\% \pm 5.66$	$6.0\% \pm 0.94$
D-T	$34.67\% \pm 1.96$	$0.0\% \pm 0.0$	$4.67\% \pm 1.09$	$27.33\% \pm 4.46$	$14.0\% \pm 3.4$
STRAP (CLIP)	$\mathbf{30.00\% \pm 2.49}$	$0.00\% \pm 0.00$	$8.67\% \pm 6.28$	$29.33\% \pm 10.51$	$\underline{24.00\% \pm 4.32}$
STRAP (DINO)	$\underline{29.33\% \pm 2.72}$	$0.00\% \pm 0.00$	$\underline{16.67\% \pm 1.97}$	$29.33\% \pm 11.34$	$18.67\% \pm 1.44$

Task Description The tasks descriptions for Tab. 1 are as follows: *Stove-Moka* combines knob-turning and Pick&Place, *Bowl-Cabinet* combines Pick&Place with cabinet closing, *Soup-Cheese* and *Mug-Mug* both contain two consecutive Pick&Place tasks, and *Book-Caddy* involves Pick&Place and insertion.

Remaining results on LIBERO-10 Tab. 3 shows the results for the remaining LIBERO-10 task not reported in the main sections. Both FR and BR outperform STRAP on the Cream-Cheese-Butter task. We hypothesize that our chunking heuristic generates sub-optimal sub-trajectories (too long) causing them to contain multiple different semantic tasks, leading to worse matches in our retrieval datasets and eventually in decreasing downstream performance.

Hyperparameters for sim results: We use the agent view (exocentric) observations for the retrieval and train policies on both agent view and in-hand observations. All results are reported over 3 training and evaluation seeds (1234, 42, 4325). We fixed both the number of segments retrieved to 100, the camera viewpoint to the agent view image for retrieval, and the number of expert demonstrations to 5. Our transformer policy was trained over all input images for 300 epochs with batch size 32 and an epoch every 200 gradient steps.

Baseline implementation details:

- **Behavior Cloning** (BC) behavior cloning using a transformer-based policy trained on $\mathcal{D}_{\text{target}}$;
- **Multi-task Policy** (MT) transformer-based policy trained on $\mathcal{D}_{\text{prior}}$;
- **BR** (BehaviorRetrieval) [15] prior work that trains a VAE on state-action pairs for retrieval and uses cosine similarity to retrieve single state-action pairs;
- **FR** (FlowRetrieval) [17] same setup as BR but VAE is trained on pre-computed optical flow from GMFlow [26];

- **D-S** (DINO state) same as BR and FR but uses off-the-shelf DINOv2 [22] features instead of training a VAE;
- **D-T** (DINO trajectory) retrieves *full* trajectories (rather than sub-trajectories) with S-DTW and DINOv2 features;

Following Lin et al. [17], we retrieve single-state action pairs for the state-based retrieval baselines (BR, FR, D-S) and pad them by also retrieving the states from $t-h$ to $t+h-1$ to make the samples compatible with our transformer-based policy. We refer the reader to Appendix A.5 for extensive ablation.

A.2 Real-world Experiments

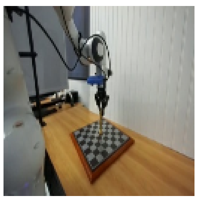

Figure 11: chess

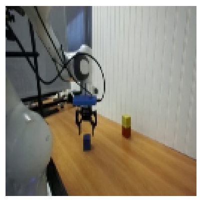

Figure 12: cube_stacking

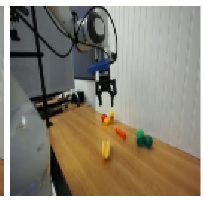

Figure 13: hotdog

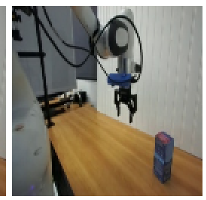

Figure 14: knock_over_box

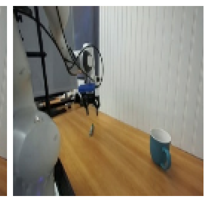

Figure 15: marker_in_mug

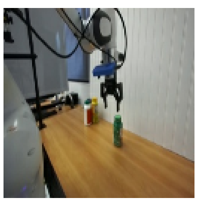

Figure 16: medicine_pnp

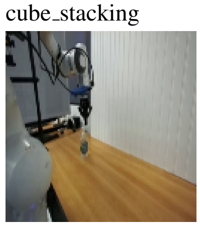

Figure 17: dispense_soap

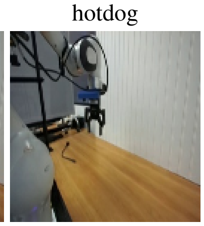

Figure 18: pull_cable_right

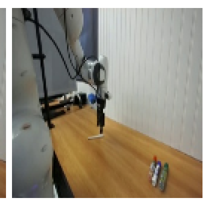

Figure 19: pen_next_to_pens

Figure 20: screwdriver

Figure 21: **Real-world tasks** in $\mathcal{D}_{\text{prior}}$

Table 4: **Task/language instructions** for the real-world dataset $\mathcal{D}_{\text{prior}}$

Environment Name	Language Instruction
chess	Move the king to the top right of the chess board
cube_stacking	Stack the blue cube on top of the tower
hotdog	Put the hotdog in the bun
knock_over_box	Knock over the box
marker_in_mug	Put the marker in the mug
medicine_pnp	Pick up the medicine box on the right and put it next to the other medicine boxes
dispense_soap	Press the soap dispenser
pull_cable_right	Pull the cable to the right
pen_next_to_pens	Put the pen next to the markers
screwdriver	Pick up the screwdriver and put it in the cup

Hyperparameters for real results: For task details please refer to Appendix A.2. For retrieval, we average the embeddings per time-step across the left, right, and in-hand camera observations while training the policies on all three image observations.

A.3 Automatic Sub-trajectory Segmentation

We propose a simple proprioception-based segmentation technique that optimizes for changes in the robot's end-effector motion indicating the transition between two chunks. For example, a

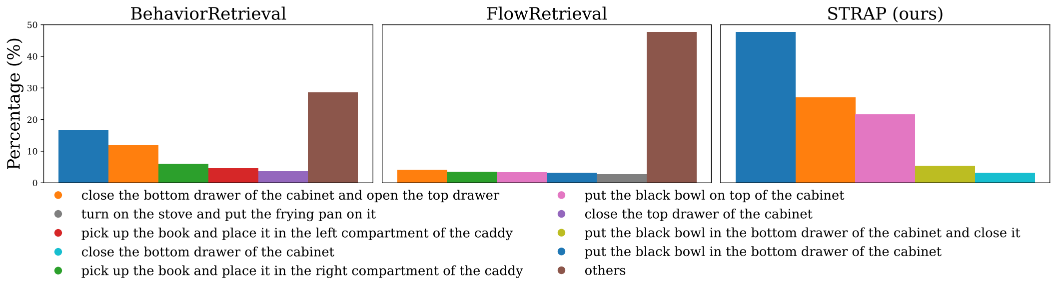

Figure 22: **Tasks distribution** in $\mathcal{D}_{\text{retrieval}}$ for different retrieval methods with target task *"put the black bowl in the bottom drawer of the cabinet and close it"*.

Pick&Place task can be split into picking and placing separated by a short pause when grasping the object. Let x_t be a vector describing the end-effector position at timestep t. We define "transition states" where the absolute velocity drops below a threshold: $\|\dot{x}\| < \epsilon$ [1]. We empirically find that this proprioception-driven segmentation can perform reasonable temporal segmentation of target trajectories into sub-components. This procedure can certainly be improved further via techniques in action recognition using vision-foundation models [27], or information-theoretic segmentation methods [28].

A.4 Qualitative Analysis of Retrieval

What types of matches are identified by *S-DTW*? To understand what data STRAP retrieves, we visualize the distribution over tasks as a function of $\mathcal{D}_{\text{retrieval}}$ proportion in Figure 22. The figure visualizes the top five tasks retrieved and accumulates the rest into the "others" category. It becomes clear that STRAP retrieves semantically relevant data – each task shares at least one sub-task with the target task. For example, *"put the black bowl in the bottom drawer of the cabinet"*, *"close the bottom drawer of the cabinet ..."* (Eq. 23). Furthermore, STRAP's retrieval is sparse, only selecting data from 5/90 semantically relevant tasks and ignoring irrelevant ones. We observe that DINOv2 features are surprisingly agnostic to different environment textures, retrieving data from the same task but in a different environment (*c.f.* Eq. 22, *"put the black bowl in the bottom drawer of the cabinet and close it"*). Furthermore, DINOv2 is robust to object poses retrieving sub-trajectories that "close the drawer" with the bowl either on the table or in the drawer (*c.f.* Eq. 24, *"close the bottom drawer of the cabinet and open the top drawer"*). Trained on optical flow, FR has no notion of visual appearance, failing to retrieve most of the semantically relevant data.

What Sub-trajectories are identified by S-DTW?

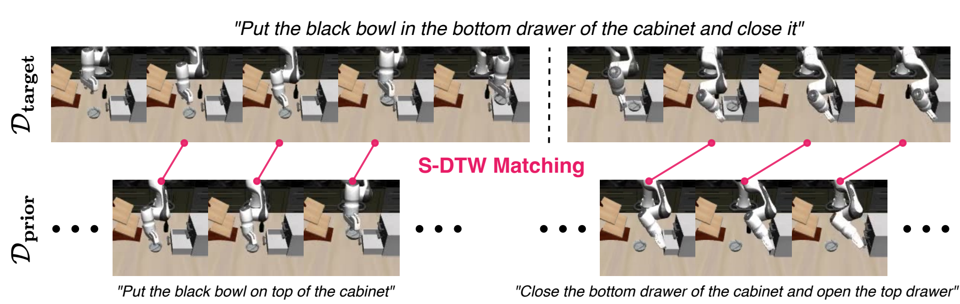

Figure 23: **Sub-trajectory matching:** S-DTW matches the sub-trajectories of $\mathcal{D}_{\text{target}}$ (top) to the relevant segments in $\mathcal{D}_{\text{prior}}$. A feature of S-DTW is that the start and end of the trajectories do not have to align, finding optimal matches for each pairing.

[1] For trajectories involving "stop-motion", this heuristic returns many short chunks as the end-effector idles, waiting for the gripper to close. To ensure a minimum length, we merge neighboring chunks until all are ≥ 20.

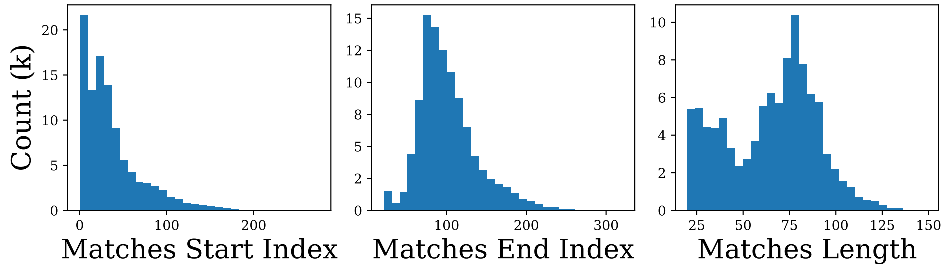

Figure 24: **Match distribution** $\mathcal{D}_{\text{prior}}$ for STRAP with target task: *"put the black bowl in the bottom drawer of the cabinet and close it"*. S-DTW finds the best matches regardless of start and end points or trajectory length. This results in a distribution over start and end points as well as a variety of trajectory lengths retrieved.

A.5 Ablations

Table 5: **Ablations - Retrieval Method:** We explore different approaches for trajectory-based retrieval. Besides the heuristic reported in the main paper, we experiment with a sliding window approach that segments a trajectory into sub-trajectories of equal length (here: 30). We use S-DTW for both sliding window sub-trajectories and full trajectories.

Method	Stove-Moka	Bowl-Cabenet	Mug-Microwave	Moka-Moka	Soup-Cream-Cheese
Sub-traj (sliding window)	76.0% ± 4.71	**75.33% ± 2.72**	26.0% ± 1.89	0.0% ± 0.0	**37.33% ± 6.62**
Full traj	**78.67% ± 2.72**	68.67% ± 1.44	**34.67% ± 1.96**	0.0% ± 0.0	28.67% ± 3.81
Method	Soup-Sauce	Cream-Cheese-Butter	Mug-Mug	Mug-Pudding	Book-Caddy
Sub-traj (sliding window)	**40.00% ± 0.94**	**27.33% ± 2.18**	**63.33% ± 3.57**	**30.00% ± 3.40**	**79.0% ± 4.95**
Full traj	4.67% ± 1.09	**27.33% ± 4.46**	43.33% ± 1.09	14.0% ± 3.4	68.0% ± 5.66

Table 6: **Ablations - Retrieval Seeds:** We run STRAP on different retrieval seeds on a subset of LIBERO-10 tasks. We report results over all possible combinations of 3 training and 3 retrieval seeds

Method	Stove-Moka	Mug-Cabinet	Book-Caddy
BC Baseline	93.11% ± 1.57	83.11% ± 2.69	93.11% ± 1.57
STRAP	**98.0% ± 1.04**	**88.67% ± 2.11**	**98.0% ± 1.04**

Table 7: **Ablations - amount data retrieved:** We explore the effect of increasing the size of $\mathcal{D}_{\text{retrieval}}$. We evaluate performance on LIBERO-10 tasks in simulation on 2 different retrieval and 3 training seeds. We randomly sample 10 demos from $\mathcal{D}_{\text{target}}$ and retrieve 1500 segments. This demonstrates STRAP's robustness over multiple seeds, as well as scalability to more data even leading to performance gains

Task	Stove-Pot	Bowl-Cabinet	Soup-Cheese	Mug-Mug	Book-Caddy
BC	86.33% ± 2.18	76.0% ± 3.97	41.67% ± 3.72	59.0% ± 2.25	92.67% ± 1.81
STRAP (DINO)	**88.67% ± 3.42**	**95.67% ± 1.19**	**45.67% ± 7.41**	**67.67% ± 1.59**	**93.71% ± 1.87**
Method	Mug-Microwave	Pots-On-Stove	Soup-Sauce	Cream cheese-Butter	Mug-Pudding
BC	**47.67% ± 4.75**	0.00% ± 0.00	23.0% ± 3.42	57.33% ± 0.77	32.0% ± 1.33
STRAP (DINO)	31.33% ± 3.73	0.00% ± 0.00	**45.0% ± 5.09**	**58.67% ± 9.58**	**38.33% ± 3.38**

Table 8: **Ablations - Diffusion Policies:** Performance on LIBERO-10 tasks using diffusion policies without language conditioning for BR and FR. These experiments replicate the training setup for BR and FR. Both methods fall short of the baselines reported in the rest of the paper.

Task	Stove-Pot	Bowl-Cabinet	Soup-Cheese	Mug-Mug	Book-Caddy
Diffusion Behavior Retrieval	$36.67\% \pm 1.44$	$68.0\% \pm 2.49$	$34.0\% \pm 2.49$	$55.33\% \pm 1.44$	$42.0\% \pm 1.63$
Diffusion Flow Retrieval	$68.67\% \pm 2.37$	$56.0\% \pm 4.32$	$18.0\% \pm 3.4$	$56.0\% \pm 3.4$	$35.33\% \pm 6.28$
Method	**Mug-Microwave**	**Pots-On-Stove**	**Soup-Sauce**	**Cream cheese-Butter**	**Mug-Pudding**
Diffusion Behavior Retrieval	$30.67\% \pm 0.54$	$0.00\% \pm 0.00$	$10.67\% \pm 1.96$	$24.0\% \pm 0.94$	$9.33\% \pm 1.44$
Diffusion Flow Retrieval	$32.67\% \pm 3.31$	$68.0\% \pm 2.49$	$6.0\% \pm 0.0$	$35.33\% \pm 0.54$	$8.0\% \pm 1.89$