Pre-Training Goal-based Models for Sample-Efficient Reinforcement Learning

Bo Liu , Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, Peter Stone
The University of Texas at Austin, Sony AI, Tsinghua University

Accepted as a Poster paper at NeurIPS 2023 Datasets and Benchmarks Track (Accepted, 7/7/7/6/6)
https://openreview.net/forum?id=xzEtNSuDJk

2025.03.12
TaeYoon Kwack
njj05043@g.skku.edu

Table of Contents

Motivation

5 Main Research topics / 6 Questions

LIBERO는 LLDM(Lifelong Learning in Decision-Making)의 **핵심적인 5가지 주제를 분석하기 위해 만들어 졌으며**,
본 연구에서는 **5가지 주제에 대한 해답을 제시할 수 있는 6가지 질문**이 주어진다.
저자의 의도는 LIBERO 벤치마크를 사용하는 연구자들이 이 **5가지 주제에 대한 분석**과 **6가지 질문에 대한 해답**을 제공하기를 바라는 듯 했으나, 대부분의 인용 논문에서 명확하게 지켜지지 않는다.

LLDM 5가지 핵심 주제

1. **Transfer of Different Types of Knowledge**
   객체, 행동 등 다양한 정보의 전이 방법 및 결과 연구
2. **Neural Architecture Design**
   다양한 모달리티의 정보를 입력받고 학습하는 신경망 설계
3. **Lifelong Learning Algorithm Design**
   망각 방지 및 지속적 학습 알고리즘 설계
4. **Robustness to Task Ordering**
   다양한 작업 순서에 강인한 방법론 설계
5. **Usage of Pretrained Models**
   사전학습 모델의 전이 학습 및 활용성 연구

6가지 질문

Q1: How do different architectures/LL algorithms perform under specific distribution shifts?
Q2: To what extent does neural architecture impact knowledge transfer in LLDM, and are there any discernible patterns in the specialized capabilities of each architecture?
Q3: How do existing algorithms from lifelong supervised learning perform on LLDM tasks?
Q4: To what extent does language embedding affect knowledge transfer in LLDM?
Q5: How robust are different LL algorithms to task ordering in LLDM?
Q6: Can supervised pretraining improve downstream lifelong learning performance in LLDM?

Method: Procedural Generation Pipeline

본 연구에서 제시하는 **데이터 생성 방법론**으로 하나의 데이터에서 무수한 데이터를 증강할 수 있다.
LIBERO 벤치마크의 데이터 셋도 같은 방식으로 만들어 졌다.

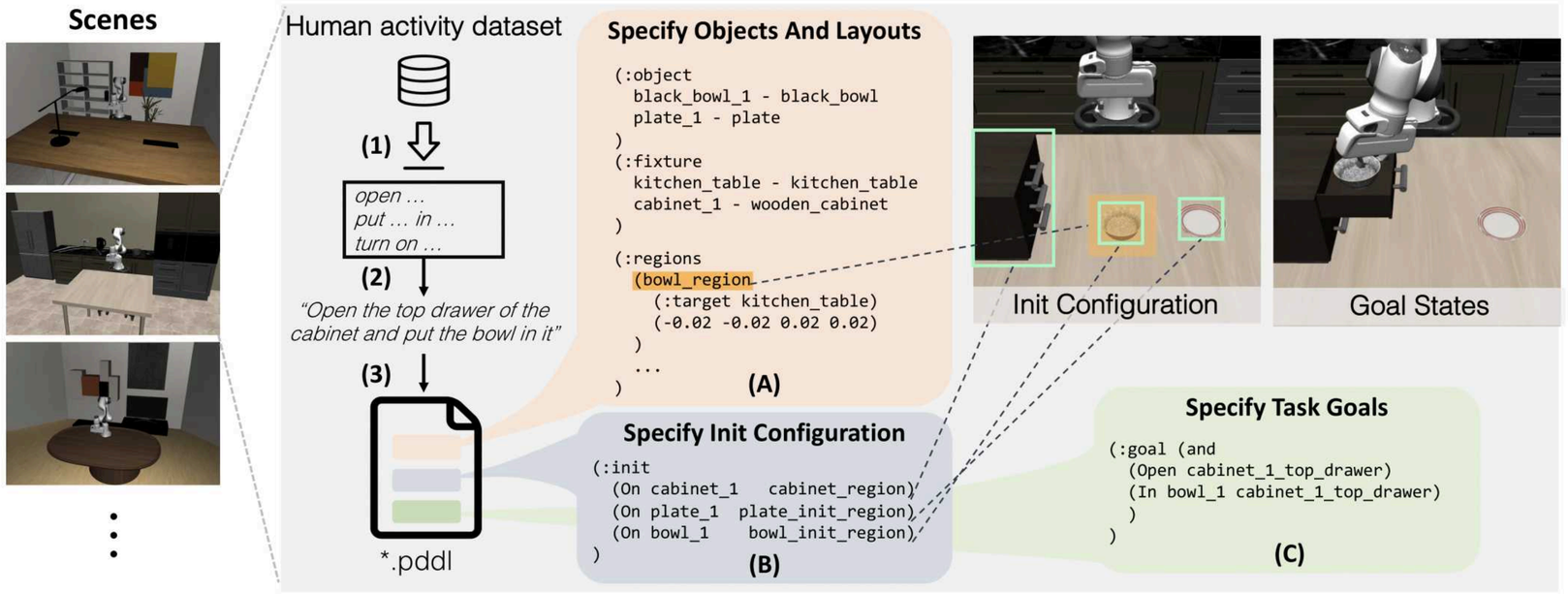

Figure 2: LIBERO's procedural generation pipeline: Extracting behavioral templates from a large-scale human activity dataset **(1)**, Ego4D, for generating task instructions **(2)**; Based on the task description, selecting the scene and generating the PDDL description file **(3)** that specifies the objects and layouts **(A)**, the initial object configurations **(B)**, and the task goal **(C)**.

Ego4D[1] : 사람의 행동을 촬영한 영상 + 캡션으로 이루어진 데이터셋

LIBERO: Dataset

본 연구에서 제시하는 **데이터 생성 방법론**으로 하나의 데이터에서 무수한 데이터를 증강할 수 있다.
LIBERO 벤치마크의 데이터 셋도 같은 방식으로 만들어 졌다.

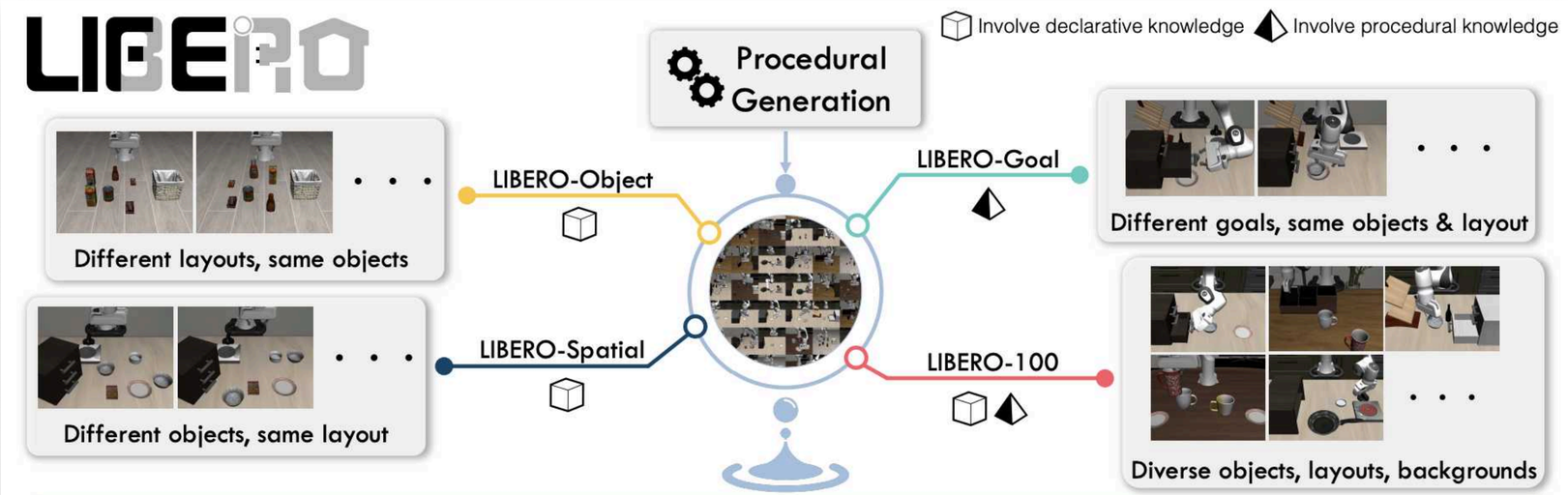

Figure 1: **Top**: LIBERO has four procedurally-generated task suites: LIBERO-SPATIAL, LIBERO-OBJECT, and LIBERO-GOAL have 10 tasks each and require transferring knowledge about spatial relationships, objects, and task goals; LIBERO-100 has 100 tasks and requires the transfer of entangled knowledge.

- **LIBERO-SPATIAL:** 동일한 물체들이 위치를 바꿔가며 등장 → 공간 학습 여부 판단
- **LIBERO-OBJECT:** 같은 공간에서 새로운 객체들이 등장 → 객체 학습 여부 판단
- **LIBERO-GOAL:** 동일한 위치를 유지하는 물체들을 대상으로 다른 Task를 실행 → 목표 학습 여부 판단
- **LIBERO-100:** 복잡한 100개의 작업. 학습용 90개의 단기 과제(LIBERO-90)와 평가용 10개의 장기 과제(LIBERO-LONG)로 구성

LIBERO: Evaluation Metric

본 연구에서는 FWT(forward transfer)[2]를 기반으로 2가지 Metric을 추가로 제안한다.

$$\mathrm{FWT} = \sum_{k\in[K]} \frac{\mathrm{FWT}_k}{K}, \quad \mathrm{FWT}_k = \frac{1}{11} \sum_{e\in\{0\ldots50\}} c_{k,k,e}$$

$$\mathrm{NBT} = \sum_{k\in[K]} \frac{\mathrm{NBT}_k}{K}, \quad \mathrm{NBT}_k = \frac{1}{K-k} \sum_{\tau=k+1}^{K} \left(c_{k,k} - c_{\tau,k}\right) \tag{3}$$

$$\mathrm{AUC} = \sum_{k\in[K]} \frac{\mathrm{AUC}_k}{K}, \quad \mathrm{AUC}_k = \frac{1}{K-k+1}\left(\mathrm{FWT}_k + \sum_{\tau=k+1}^{K} c_{\tau,k}\right)$$

$c_{i,j,e}$: 에이전트가 task j에서의 성공률(success rate)로, 에이전트가 i−1개의 이전 태스크를 학습하고, i번째 태스크를 학습한 후 e 에포크(epoch) 동안의 성능을 의미

- **FWT(Forward Transfer):** 이전에 학습된 Task들을 기반으로 새로운 Task를 학습하는 속도 (높을수록 성능 좋음)
- **NBT(Negative Backward Transfer):** 새로운 태스크를 학습했을 때, 이전에 학습한 Task 망각 정도 (높을수록 망각이 심함)
- **AUC (Area Under Curve):** 성공률 곡선의 면적을 측정하며, 전반적인 학습 성능을 평가 (높을수록 성능 좋음)

LIBERO: Baseline Learning Algorithm

1. **ER (Experience Replay)**[3] → Rehearsal-based Approach (복습 기반 접근)
   설명: 이전에 학습한 Task의 데이터를 replay buffer에 저장 후, 새로운 태스크 학습 시 이 데이터를 섞어서 학습.
   장점: 쉽고 범용적
   단점: 저장 공간이 증가, 재학습 시간 증가

2. **EWC (Elastic Weight Consolidation)**[4] → Regularization-based Approach (정규화 기반 접근)
   설명: 새로운 Task를 학습할 때 신경망 가중치 업데이트를 제한하여 이전 Task의 성능이 저하줄임

$$\mathcal{L}_k^{EWC}(\theta) = \mathcal{L}_K^{BC}(\theta) + \sum_i \frac{\lambda}{2} F_i \left(\theta_i - \theta_{k-1,i}^*\right)^2,$$

   장점: 적은 메모리로 효율적으로 망각을 방지
   단점: Fisher Information Matrix 계산이 복잡하고, 계산 비용이 클 수 있음

3. **PACKNET**[5] → Dynamic Architecture-based Approach (동적 구조 기반 접근)
   설명: 네트워크 학습 후 가장 중요한 25%의 가중치를 고정 / 남은 가중치로 새로운 Task를 학습
   장점: 이전 Task의 성능을 효과적으로 유지
   단점: 학습이 진행될수록 사용할 수 있는 가중치가 점차 감소

4. **SEQL (Sequential Finetuning)** → Lower Bound 알고리즘
   설명: 새로운 Task를 학습할 때 이전 지식 없이 새롭게 학습. 그 어떤 알고리즘도 성능이 더 낮을 수 없다.

5. **MTL (Multitask Learning)** → Upper Bound 알고리즘
   설명: 모든 태스크를 동시에 학습. 그 어떤 알고리즘도 성능이 더 좋을 수 없다.

LIBERO: Baseline Neural Architecture

Neural Architecture	시각정보	언어정보	시계열정보(정책)
ResNet-RNN[7]	ResNet	FiLM[8]	LSTM
ResNet-T[9]	ResNet	Transformer[10]	Transformer(Decoder)
ViT-T[11]	ViT	Transformer	Transformer(Decoder)

명령어 언어정보는 BERT[6] embedding을 사용
action은 GMM을 기반으로 결정(신경망이 각 가우시안 분포의 μ, Σ와 가우시안의 가중치 π를 예측)

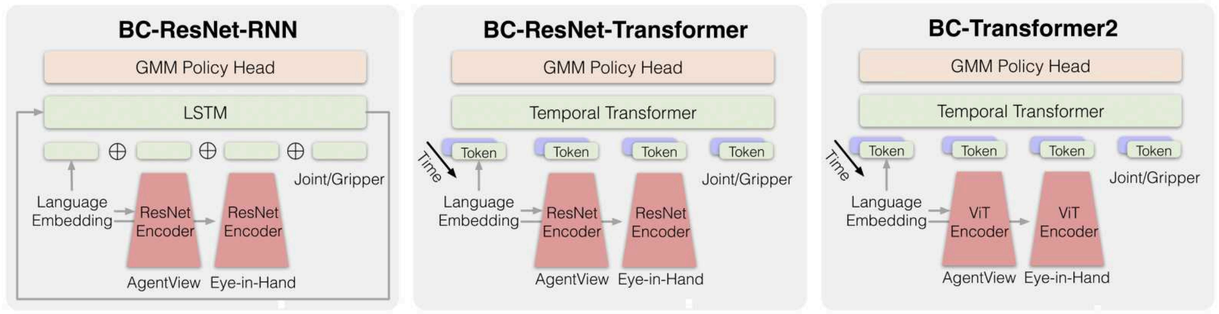

Figure 6: We provide visualizations of the architectures for ResNet-RNN, ResNet-T, and ViT-T, respectively. It is worth noting that each model architecture incorporates language embedding in distinct ways.

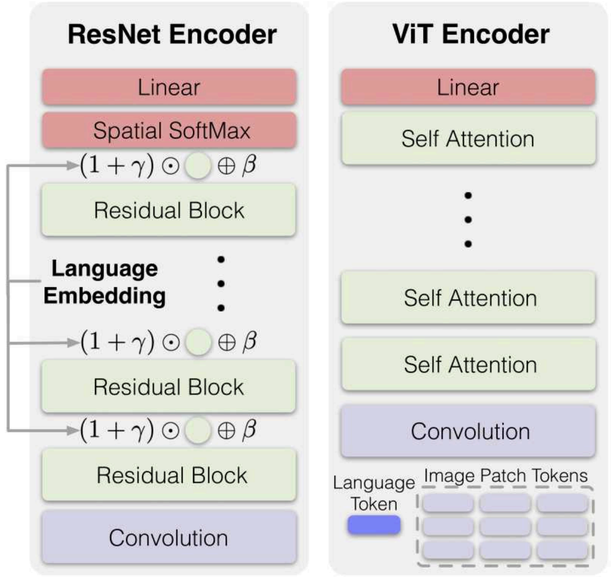

Figure 7: The image encoders: ResNet-based encoder and the vision transformer-based encoder.

Experiment

a. Study on the Policy's Neural Architectures (Q1, Q2)

Policy Arch.	ER			PackNet		
	FWT(↑)	NBT(↓)	AUC(↑)	FWT(↑)	NBT(↓)	AUC(↑)
			LIBERO-Long			
ResNet-RNN	0.16 ± 0.02	**0.16** ± 0.02	0.08 ± 0.01	0.13 ± 0.00	0.21 ± 0.01	0.03 ± 0.00
ResNet-T	**0.48** ± 0.02	0.32 ± 0.04	**0.32** ± 0.01	0.22 ± 0.01	**0.08** ± 0.01	0.25 ± 0.00
ViT-T	0.38 ± 0.05	0.29 ± 0.06	0.25 ± 0.02	**0.36** ± 0.01	0.14 ± 0.01	**0.34** ± 0.01
			LIBERO-Spatial			
ResNet-RNN	0.40 ± 0.02	0.29 ± 0.02	0.29 ± 0.01	0.27 ± 0.03	0.38 ± 0.03	0.06 ± 0.01
ResNet-T	**0.65** ± 0.03	**0.27** ± 0.03	**0.56** ± 0.01	0.55 ± 0.01	**0.07** ± 0.02	**0.63** ± 0.00
ViT-T	0.63 ± 0.01	0.29 ± 0.02	0.50 ± 0.02	**0.57** ± 0.04	0.15 ± 0.00	0.59 ± 0.03
			LIBERO-Object			
ResNet-RNN	0.30 ± 0.01	**0.27** ± 0.05	0.17 ± 0.05	0.29 ± 0.02	0.35 ± 0.02	0.13 ± 0.01
ResNet-T	0.67 ± 0.07	0.43 ± 0.04	0.44 ± 0.06	**0.60** ± 0.07	**0.17** ± 0.05	**0.60** ± 0.05
ViT-T	**0.70** ± 0.02	0.28 ± 0.01	**0.57** ± 0.01	0.58 ± 0.03	0.18 ± 0.02	0.56 ± 0.04
			LIBERO-Goal			
ResNet-RNN	0.41 ± 0.00	0.35 ± 0.01	0.26 ± 0.01	0.32 ± 0.03	0.37 ± 0.04	0.11 ± 0.01
ResNet-T	**0.64** ± 0.01	**0.34** ± 0.02	**0.49** ± 0.02	0.63 ± 0.02	**0.06** ± 0.01	0.75 ± 0.01
ViT-T	0.57 ± 0.00	0.40 ± 0.02	0.38 ± 0.01	**0.69** ± 0.02	0.08 ± 0.01	**0.76** ± 0.02

Table 1: Performance of the three neural architectures using ER and PackNet on the four task suites. Results are averaged over three seeds and we report the mean and standard error. The best performance is **bolded**, and colored in **purple** if the improvement is statistically significant over other neural architectures, when a two-tailed, Student's t-test under equal sample sizes and unequal variance is applied with a p-value of 0.05.

ResNet-T와 ViT-T는 ResNet-RNN보다 전반적으로 우수

- Transformer가 더 효과적으로 시간 정보 (Temporal Information)를 처리

알고리즘별 성능 차이:

- ER에서는 ResNet-T가 대부분의 태스크에서 좋은 성능을 기록 (단, LIBERO-OBJECT에서는 ViT-T가 더 우수)
- PackNet에서는 ViT-T가 LIBERO-LONG에서 강점을 보이고, ResNet-T는 다른 태스크에서 더 뛰어난 성능을 나타냄

Experiment

b. Study on Lifelong Learning Algorithms (Q3)

Lifelong Algo.	FWT(↑)	NBT(↓)	AUC(↑)	FWT(↑)	NBT(↓)	AUC(↑)
	LIBERO-LONG			LIBERO-SPATIAL		
SEQL	**0.54** ± 0.01	0.63 ± 0.01	0.15 ± 0.00	**0.72** ± 0.01	0.81 ± 0.01	0.20 ± 0.01
ER	0.48 ± 0.02	0.32 ± 0.04	**0.32** ± 0.01	0.65 ± 0.03	0.27 ± 0.03	0.56 ± 0.01
EWC	0.13 ± 0.02	0.22 ± 0.03	0.02 ± 0.00	0.23 ± 0.01	0.33 ± 0.01	0.06 ± 0.01
PACKNET	0.22 ± 0.01	**0.08** ± 0.01	0.25 ± 0.00	0.55 ± 0.01	**0.07** ± 0.02	**0.63** ± 0.00
MTL			0.48 ± 0.01			0.83 ± 0.00
	LIBERO-OBJECT			LIBERO-GOAL		
SEQL	**0.78** ± 0.04	0.76 ± 0.04	0.26 ± 0.02	**0.77** ± 0.01	0.82 ± 0.01	0.22 ± 0.00
ER	0.67 ± 0.07	0.43 ± 0.04	0.44 ± 0.06	0.64 ± 0.01	0.34 ± 0.02	0.49 ± 0.02
EWC	0.56 ± 0.03	0.69 ± 0.02	0.16 ± 0.02	0.32 ± 0.02	0.48 ± 0.03	0.06 ± 0.00
PACKNET	0.60 ± 0.07	**0.17** ± 0.05	**0.60** ± 0.05	0.63 ± 0.02	**0.06** ± 0.01	**0.75** ± 0.01
MTL			0.54 ± 0.02			0.80 ± 0.01

Table 2: Performance of three lifelong algorithms and the SEQL and MTL baselines on the four task suites, where the policy is fixed to be RESNET-T. Results are averaged over three seeds and we report the mean and standard error. The best performance is **bolded**, and colored in **purple** if the improvement is statistically significant over other algorithms, when a two-tailed, Student's t-test under equal sample sizes and unequal variance is applied with a p-value of 0.05.

- **Sequential Finetuning (SEQL)**이 예상과 달리 FWT에서 가장 좋은 성능을 보임
  → 지속적 학습 전용 알고리즘이 아님에도 불구하고, FWT에서 뛰어난 학습 능력을 보여줌
- **PackNet**은 LIBERO- X 들에서 우수한 성능을 기록
  → 망각 방지 (Catastrophic Forgetting)에 강점이 있지만, LIBERO-LONG에서는 성능이 떨어짐 (파라미터 고정으로 인한 용량 제한 때문)
- **Experience Replay (ER)**는 모든 Task 에서 비교적 균형 잡힌 성능을 보임
- **Elastic Weight Consolidation (EWC)**는 기대보다 성능이 저조, 강한 정규화가 오히려 학습을 방해하는 결과를 보임

Experiment

c. Study on Language Embeddings as the Task Identifier (Q4)

Embedding Type	Dimension	FWT(↑)	NBT(↓)	AUC(↑)
BERT	768	0.48 ± 0.02	**0.32** ± 0.04	0.32 ± 0.01
CLIP	512	**0.52** ± 0.00	0.34 ± 0.01	**0.35** ± 0.01
GPT-2	768	0.46 ± 0.01	0.34 ± 0.02	0.30 ± 0.01
Task-ID	768	0.50 ± 0.01	0.37 ± 0.01	0.33 ± 0.01

Table 3: Performance of a lifelong learner using four different language embeddings on LIBERO-LONG, where we fix the policy architecture to RESNET-T and the lifelong learning algorithm to ER. The Task-ID embeddings are retrieved by feeding "Task + ID" into a pretrained BERT model. Results are averaged over three seeds and we report the mean and standard error. The best performance is **bolded**. No statistically significant difference is observed among the different language embeddings.

- BERT, CLIP, GPT-2 등 다양한 언어 임베딩을 실험했으나, 성능 차이는 크지 않았음
- 이는 복잡한 언어 표현보다 간단한 Task-ID 임베딩만으로도 지속적 학습에서 태스크 구분에 충분하다는 것을 시사함

Experiment

d. Study on task ordering (Q5)

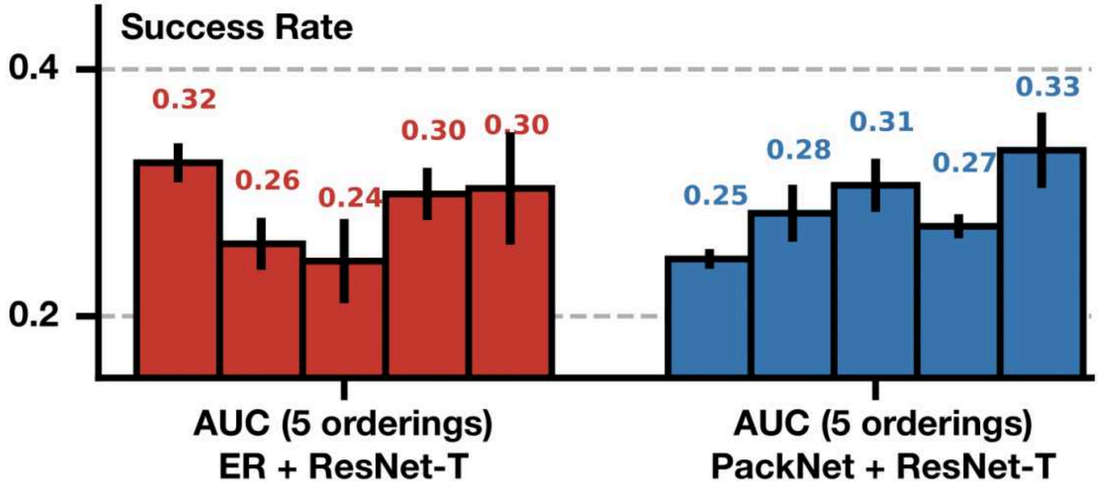

Figure 4: Performance of ER and PACKNET using RESNET-T on five different task orderings. An error bar shows the performance standard deviation for a fixed ordering.

- **PackNet**은 태스크 순서에 매우 민감하게 반응했으며, 특정 순서에서는 성능이 급격히 하락
- **ER**은 상대적으로 태스크 순서 변화에 강한 견고함(Robust)을 보임

Experiment

e. Study on How Pretraining Affects Downstream LLDM (Q6)

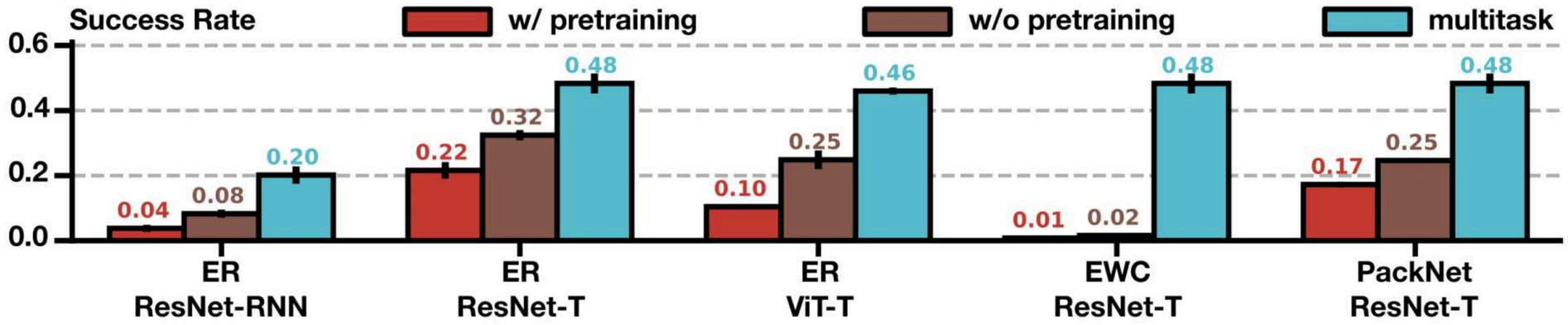

Figure 5: Performance of different combinations of algorithms and architectures without pretraining or with pretraining. The multi-task learning performance is also included for reference.

- 일반적인 지도 학습 기반 사전 학습 (Supervised Pretraining)은 오히려 지속적 학습에서 성능 저하를 유발
- 이는 일반적인 사전 학습이 LLDM (Lifelong Learning in Decision Making)에서 유용한 특성을 충분히 전달하지 못함을 의미 → LLDM에 맞는 더 효과적인 사전 학습 기법이 필요함

IsCiL의 한계로 생각될 수 있음...

Reference

1. Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 18995–19012, 2022
2. Natalia Díaz-Rodríguez, Vincenzo Lomonaco, David Filliat, and Davide Maltoni. Don't forget, there is more than forgetting: new metrics for continual learning. arXiv preprint arXiv:1810.13166, 2018.
3. Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. arXiv preprint arXiv:1902.10486, 2019
4. James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. Proceedings of the national academy of sciences, 114(13):3521–3526, 2017.
5. Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 7765–7773, 2018.
6. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.
7. Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. arXiv preprint arXiv:2108.03298, 2021.
8. Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 32, 2018.
9. Yifeng Zhu, Abhishek Joshi, Peter Stone, and Yuke Zhu. Viola: Imitation learning for visionbased manipulation with object proposal priors. arXiv preprint arXiv:2210.11339, 2022.
10. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. Advances in neural information processing systems, 30, 2017.
11. Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In International Conference on Machine Learning, pages 5583–5594. PMLR, 2021.

Pre-Training Goal-based Models for Sample-Efficient Reinforcement Learning

Bo Liu , Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, Peter Stone
The University of Texas at Austin, Sony AI, Tsinghua University

Accepted as a Poster paper at NeurIPS 2023 Datasets and Benchmarks Track (Accepted, 7/7/7/6/6)
https://openreview.net/forum?id=xzEtNSuDJk

2025.03.12
TaeYoon Kwack
njj05043@g.skku.edu

Appendix

a. Ego4D Example

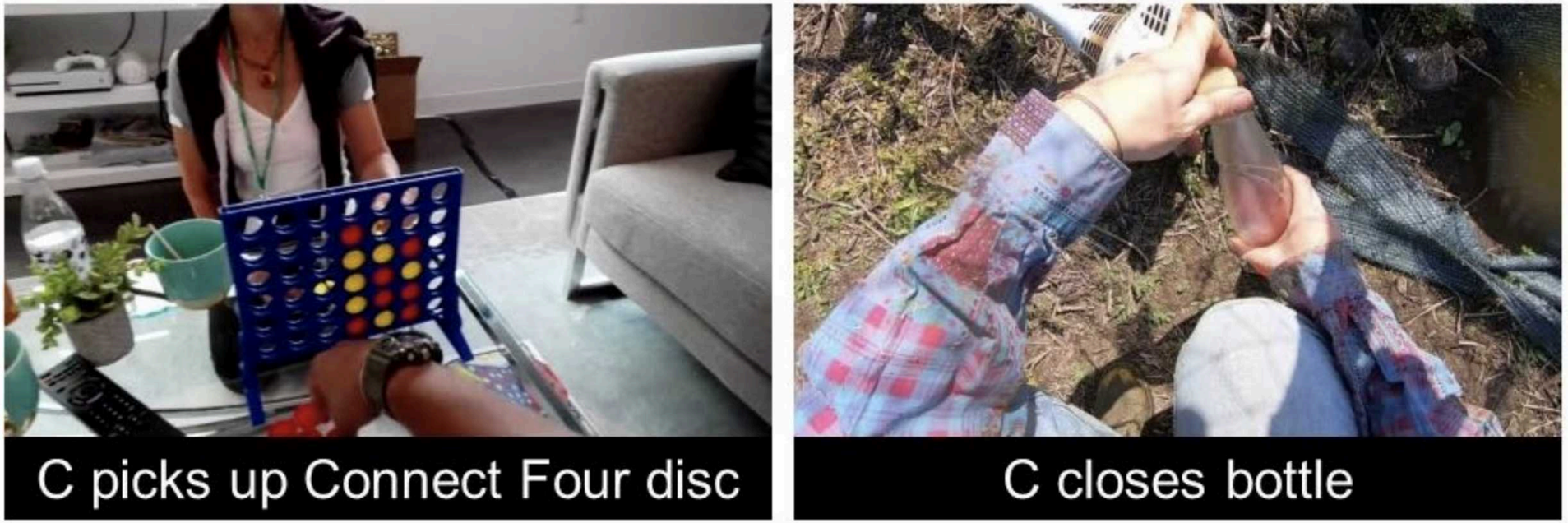

Figure 5. Example narrations. “C” refers to camera wearer.

Appendix

b. Baseline Neural Architecture Configuration

Variable	Value
resnet_image_embed_size	64
text_embed_size	32
rnn_hidden_size	1024
rnn_layer_num	2
rnn_dropout	0.0

Table 5: Hyper parameters of RESNET-RNN.

Variable	Value
extra_info_hidden_size	128
img_embed_size	64
transformer_num_layers	4
transformer_num_heads	6
transformer_head_output_size	64
transformer_mlp_hidden_size	256
transformer_dropout	0.1
transformer_max_seq_len	10

Table 6: Hyper parameters of RESNET-T.

Variable	Value
extra_info_hidden_size	128
img_embed_size	128
spatial_transformer_num_layers	7
spatial_transformer_num_heads	8
spatial_transformer_head_output_size	120
spatial_transformer_mlp_hidden_size	256
spatial_transformer_dropout	0.1
spatial_down_sample_embed_size	64
temporal_transformer_input_size	null
temporal_transformer_num_layers	4
temporal_transformer_num_heads	6
temporal_transformer_head_output_size	64
temporal_transformer_mlp_hidden_size	256
temporal_transformer_dropout	0.1
temporal_transformer_max_seq_len	10

Table 7: Hyper parameters of VIT-T.

Appendix

c. LIBERO-SPATIAL

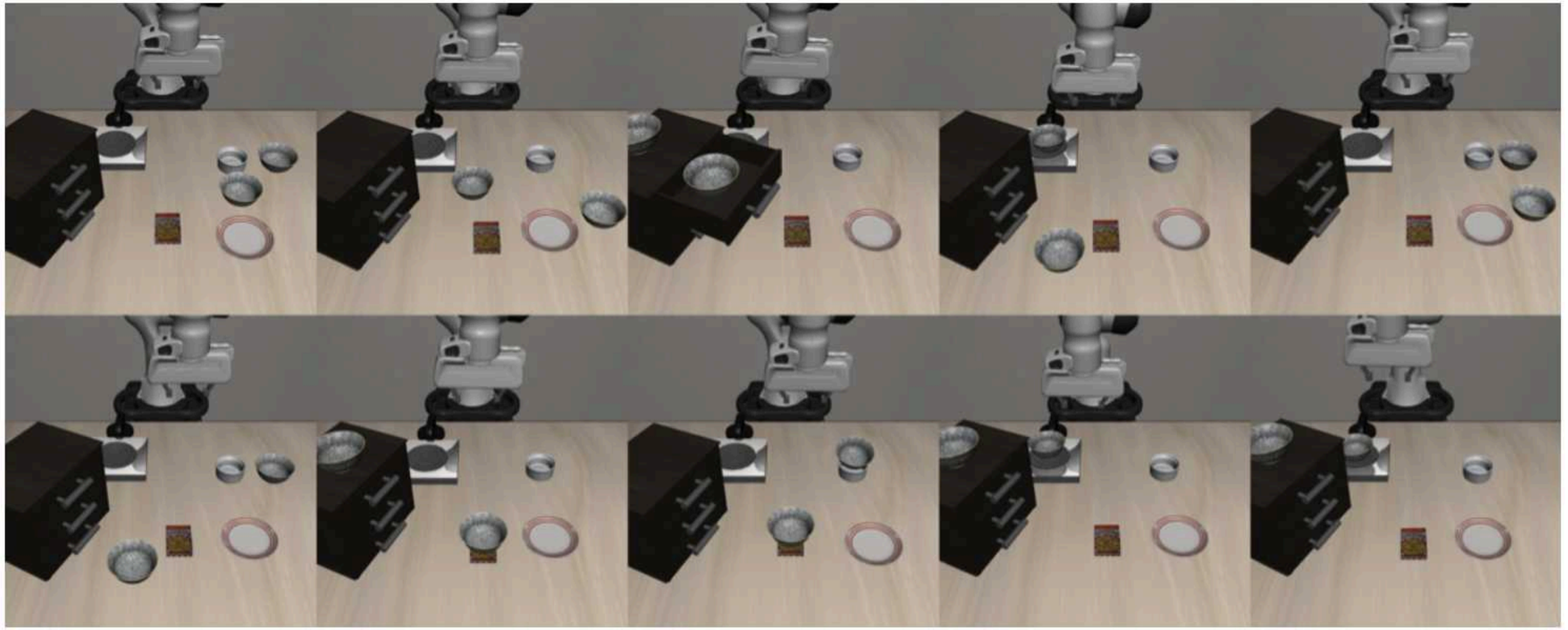

Figure 8: LIBERO-SPATIAL

Appendix

d. LIBERO-OBJECT

Figure 9: LIBERO-OBJECT

Appendix

e. LIBERO-GOAL

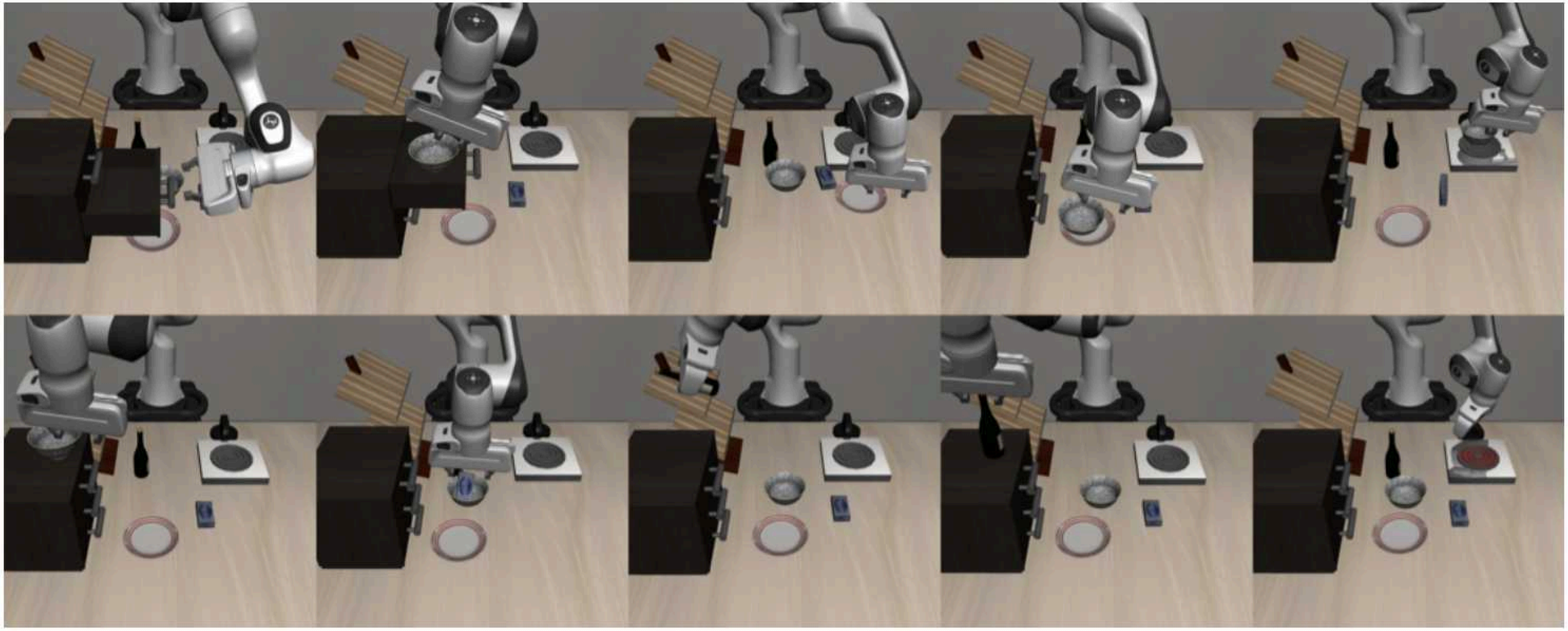

Figure 10: LIBERO-GOAL

Appendix

f. LIBERO-100

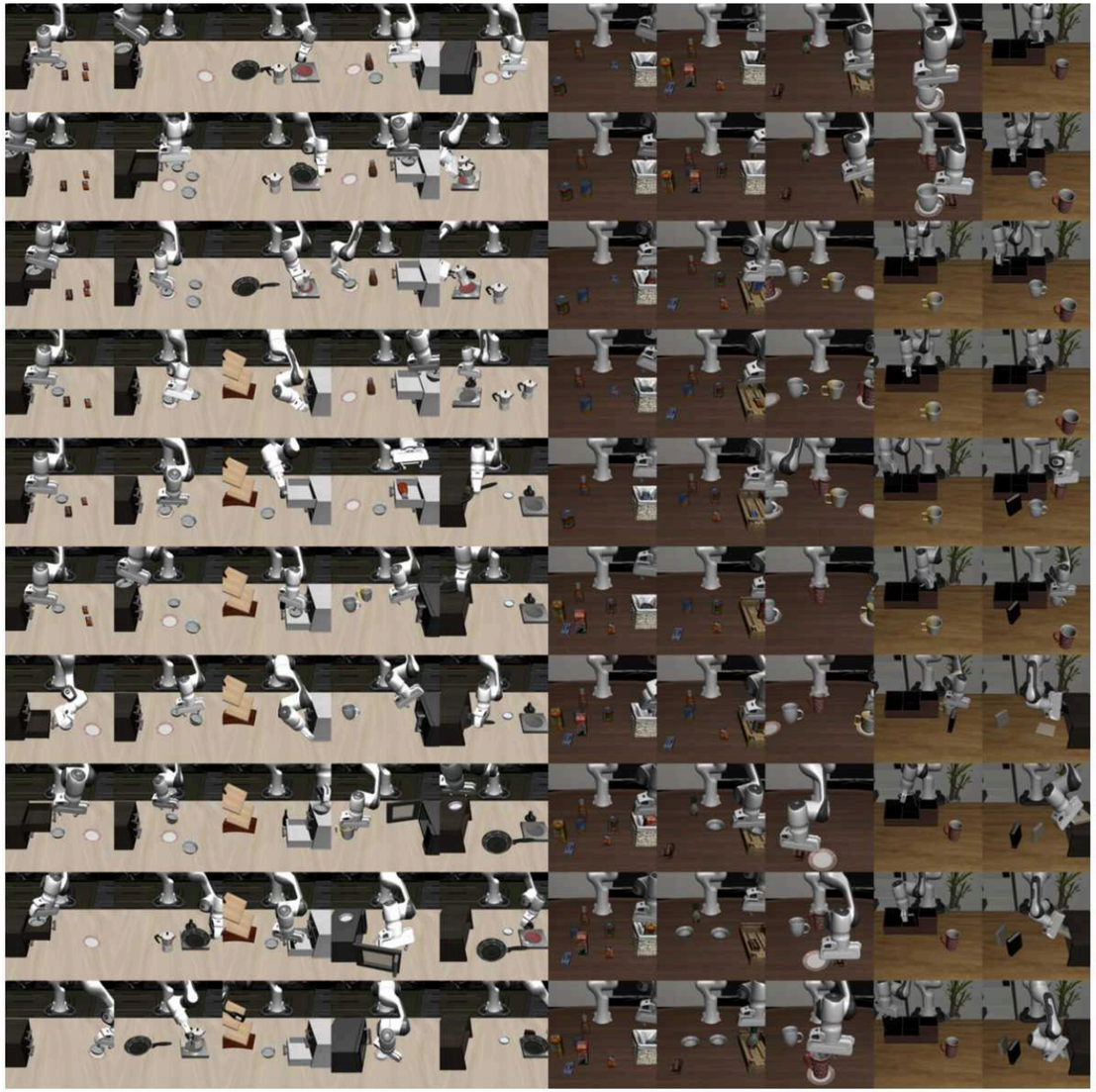

Figure 11: LIBERO-100