# CONTINUAL ADAPTATION OF MULTITASK WORLD MODEL WITH EMBEDDING RETRIEVAL

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Recent advances in world models have enabled generalist agents to learn diverse continuous control tasks within a single model. However, joint training on all desired tasks is not always feasible, as limited resources constrain access to collected data and simulation environments at a given time. An important goal is therefore to adapt pretrained multitask agents to newly accessible tasks while retaining their existing capabilities. We propose MARR, a framework for multitask world model adaptation with capability retention. Given a short probing trajectory, our framework retrieves a task-relevant dynamics representation from the pretrained world model to initialize adaptation, enabling the agent to learn multiple new tasks efficiently. We adapt world-model weights through a factorized parameterization that learns new combinations of fixed pretrained bases, while exemplar-based distillation helps retain prior knowledge and skills. Experiments across diverse continuous control tasks show that our approach learns new tasks while retaining performance on previously learned tasks and outperforms a range of existing baselines. These results support multitask adaptation as a promising approach to broadening the capabilities of a multitask model.

## 1 INTRODUCTION

Building general-purpose models for continuous control has become an increasingly important goal in robot learning. Substantial progress has been made along two complementary directions: imitation learning and reinforcement learning (RL). Imitation-learning approaches train large control policies on broad collections of expert trajectories spanning multiple tasks and embodiments (Reed et al., 2022; Brohan et al., 2022; 2023; Octo Model Team et al., 2024; Kim et al., 2024). Their performance depends on the quality and coverage of the collected demonstrations, while obtaining diverse, high-quality demonstrations can be costly (Mandlekar et al., 2022; Khazatsky et al., 2024). In parallel, RL learns control policies through interaction and reward feedback. Model-free multitask methods, including extensions of soft actor-critic (SAC) and proximal policy optimization (PPO), learn shared policies across tasks (Yu et al., 2019; Joshi et al., 2025), but can require substantial environment interaction and suffer from interference between task objectives (Yu et al., 2020). Model-based methods can improve data efficiency by planning with learned dynamics, but face challenges from model inaccuracies and the computational cost of long-horizon planning (Hansen et al., 2022). Recent world-model methods integrate learned latent dynamics with policy and value learning, combining model-based prediction with model-free components. The Dreamer family optimizes behavior through imagined rollouts in a learned latent model, whereas the TD-MPC family integrates latent world models with temporal-difference learning and model-predictive control (Hafner et al., 2020; 2025; Hansen et al., 2022; 2024). Recent multitask extensions further demonstrate broad multitask learning and favorable scaling with model and data size (Hansen et al., 2024; 2025). Together, these efforts point toward foundation models for continuous control: a single trained agent can support a broad range of tasks.

Despite this progress, existing generalist agents are predominantly trained with a fixed number of tasks defined in advance. Once training is complete, extending the model to a new task is typically treated as a separate transfer-learning problem. Fine-tuning is a common way to adapt a trained model to a held-out task, but it may damage performance on previously learned tasks (McCloskey & Cohen, 1989; French, 1999). Moreover, prior work on learning new tasks while preserving existing capabilities has largely considered tasks that arrive one at a time (Khetarpal et al., 2022; Wołczyk



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

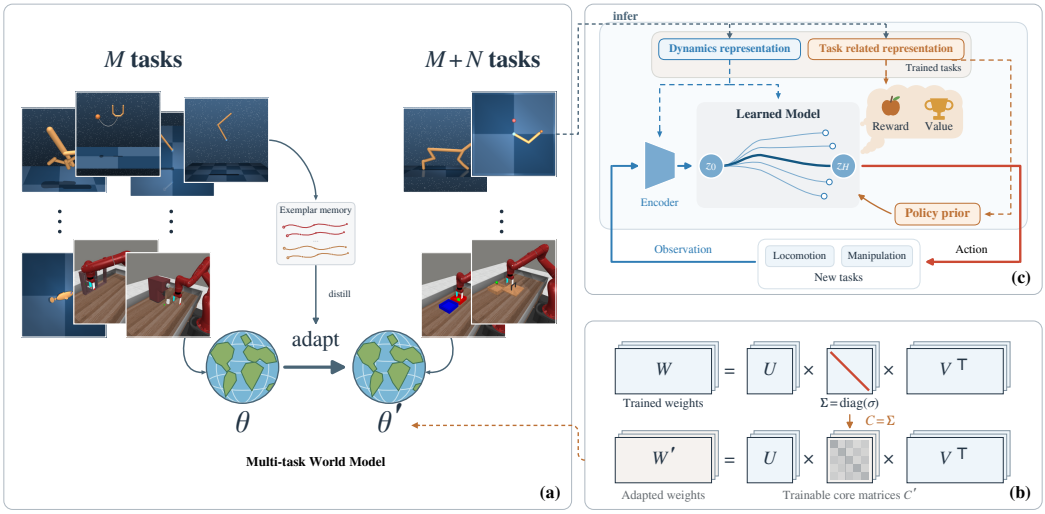


Figure 1: **Overview of our framework.** (a) A trained multitask world model learns new tasks while retaining prior capabilities through exemplar-based distillation. (b) Adaptation learns core weight matrices within fixed SVD bases. (c) Separate retrieval of dynamics and task-related representations transfers knowledge for prediction and control.

et al., 2021), leaving simultaneous adaptation to multiple new tasks with retention of previously learned capabilities comparatively underexplored.

Advances in model scaling have enabled larger control models to learn diverse tasks jointly (Hansen et al., 2024). In parallel, research on replay, regularization, architecture design, optimization, and representation learning has developed complementary mechanisms for retaining previously learned capabilities (Rolnick et al., 2019; Kirkpatrick et al., 2017; Rusu et al., 2016; Lopez-Paz & Ranzato, 2017; Cha et al., 2021; Wang et al., 2024). Building on these advances, we propose MARR, a framework for multitask world model adaptation with capability retention. Our main contributions are as follows: (1) By briefly exploring each new task, MARR retrieves dynamics-related and task-related representations from a trained multitask world model, providing an effective initialization for adaptation. It then jointly learns multiple new tasks by optimizing core weight matrices within fixed SVD bases, while exemplar-based distillation helps retain prior knowledge and skills. (2) We conduct an exploratory empirical study of MARR for simultaneous adaptation to multiple continuous control tasks across DMControl (Tassa et al., 2018) and Meta-World (Yu et al., 2019). These results suggest that multitask adaptation with capability retention is a promising direction for advancing generalist agents.

## 2 Related Work

**Multitask RL.** Multitask reinforcement learning aims to share representations and parameters across tasks within a single agent. Model-free approaches improve task sharing through gradient conflict mitigation, modular policies, and context-conditioned representations (Yu et al., 2020; Yang et al., 2020; Sodhani et al., 2021). Model-based methods learn environment dynamics and use predicted trajectories for planning (Chua et al., 2018). Within this family, multitask world-model agents such as TD-MPC2 and Newt combine model-based prediction and planning with model-free policy and value learning, supporting shared control across diverse tasks and embodiments (Hansen et al., 2022; 2024; 2025). MARR studies how to reuse the knowledge acquired through joint training, adapting a single world-model agent to multiple newly available tasks while retaining its earlier capabilities.

**Knowledge Retention.** Following the taxonomy of Wang et al. (2024), existing approaches can be organized into replay (Shin et al., 2017; Rolnick et al., 2019), regularization (Kirkpatrick et al., 2017; Zenke et al., 2017; Aljundi et al., 2018), architecture design (Rusu et al., 2016; Mallya &

Lazebnik, 2018), optimization (Lopez-Paz & Ranzato, 2017), and representation learning (Cha et al., 2021). Replay and distillation are also used to retain policy behavior and world-model knowledge in RL (Wołczyk et al., 2022; Kessler et al., 2023; Yang et al., 2024). MARR combines exemplar-based component distillation with weight adaptation within fixed SVD bases to retain both predictive and control capabilities.

**Adaptation.** Adaptation can reuse prior knowledge through parameter optimization or inference from interaction context. Optimization-based approaches update policy or dynamics parameters on new experience, with meta-learning providing initializations for rapid updates (Finn et al., 2017; Kaushik et al., 2020). Context-based approaches infer task variables or dynamics parameters to condition control (Rakelly et al., 2019; Xian et al., 2021; Zhou et al., 2025). These mechanisms can be combined: context can select an initialization before further optimization. MARR starts from a jointly trained multitask world model and uses brief random exploration to retrieve dynamics-related and task-related embeddings separately, without a dedicated meta-training objective or a learned context-to-task encoder. It then jointly adapts to multiple new tasks while retaining earlier capabilities.

# 3 Multitask Adaptation with Capability Retention

## 3.1 Problem Formulation

An agent learns a control policy through interaction with an environment. We model each task as a discounted Markov decision process $\mathcal{M}_i = (\mathcal{S}_i, \mathcal{A}_i, p_i, r_i, \gamma_i)$, where $\mathcal{S}_i$ and $\mathcal{A}_i$ are the state and action spaces, $p_i(s' \mid s, a)$ is the transition distribution, $r_i(s, a)$ is the reward function, and $\gamma_i$ is the discount factor. The learning objective is to find a policy $\mu(a \mid s, i)$ that maximizes the expected discounted return $J_i(\mu) = \mathbb{E}_\mu[\sum_{t=0}^{\infty} \gamma_i^t r_i(s_t, a_t)]$.

We study multitask adaptation with capability retention. Consider a world model $f(\theta, E)$ initially trained on $\mathcal{T}_{\text{pre}} = \{\mathcal{M}_1, \ldots, \mathcal{M}_N\}$ and a set of newly available tasks $\mathcal{T}_{\text{new}} = \{\mathcal{M}_{N+1}, \ldots, \mathcal{M}_{N+n}\}$. Without assuming access to pre-collected training datasets or expert demonstrations, we begin adaptation to each new task $\mathcal{M}_i \in \mathcal{T}_{\text{new}}$ with only a short interaction context $\mathcal{C}_i$ consisting of randomly sampled trajectories. We seek to use this limited context to identify reusable knowledge in the trained model, providing an effective initialization that accelerates subsequent adaptation. Our objective is to learn the new tasks while preserving performance on previously learned tasks, yielding a single agent that performs well across $\mathcal{T} = \mathcal{T}_{\text{pre}} \cup \mathcal{T}_{\text{new}} = \{\mathcal{M}_1, \ldots, \mathcal{M}_{N+n}\}$.

## 3.2 Embedding Retrieval Through Multi-task World Models

Following TD-MPC2 (Hansen et al., 2024), we consider a world model comprising an encoder $h_\theta$, a latent dynamics model $d_\theta$, a reward predictor $R_\theta$, an actor $\pi_\theta$, and an action-value function $Q_\theta$. We group these into dynamics-related components $(h_\theta, d_\theta)$ and task-related components $(R_\theta, \pi_\theta, Q_\theta)$. Generally, the agent encodes states into a latent space and plans using predicted rewards and values along latent dynamics rollouts, while the actor provides a policy prior to guide action sampling during planning. In the multitask setting considered here, network parameters $\theta$ are shared across tasks, and each component is conditioned on a task embedding $e_j$. The embedding table $E = [e_1, \ldots, e_N]^\top$, together with the shared parameters, defines the task-conditioned model $f(\theta, E)$.

In a trained multitask model, each task embedding can be regarded as a learned representation that, together with the shared model weights, determines task-specific dynamics predictions, policy behavior, and reward and value estimates. We seek to reuse these representations to initialize a new task from a short interaction context obtained from trajectories sampled during random exploration. To provide an effective initialization, we use these observations to infer task similarities and retrieve suitable embeddings separately for the dynamics-related components $(h_\theta, d_\theta)$ and the task-related components $(R_\theta, \pi_\theta, Q_\theta)$.

We score candidate embeddings using the latent consistency, reward prediction, and temporal-difference losses from TD-MPC2's learning objective (Hansen et al., 2024), denoted by $\mathcal{L}_{\text{dyn}}$, $\mathcal{L}_R$, and $\mathcal{L}_Q$, respectively. Let $\mathcal{E} = \{e_1, \ldots, e_N\}$ be the set of embeddings for previously learned tasks.

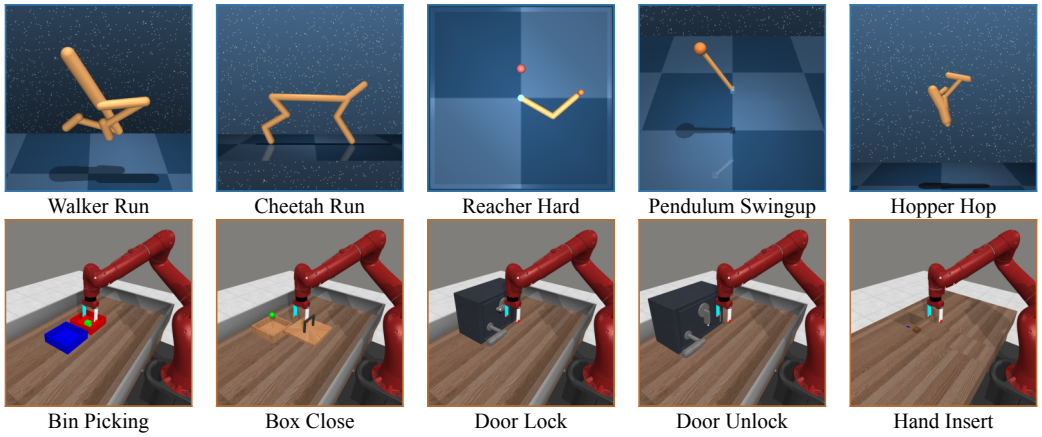


Figure 2: **The ten new tasks.** Five DeepMind Control tasks (top) and five Meta-World tasks (bottom).

We first use $\mathcal{C}_i$ to retrieve a previously learned task embedding for the dynamics-related components and denote it by $e^{\text{dyn}}$. Holding $e^{\text{dyn}}$ fixed, we then retrieve $e^{\text{ctrl}}$ for the task-related components:

$$e^{\text{dyn}} \leftarrow \arg\min_{e_j \in \mathcal{E}} \mathcal{L}_{\text{dyn}}(e_j; \mathcal{C}_i),$$
$$e^{\text{ctrl}} \leftarrow \arg\min_{e_j \in \mathcal{E}} \Big[\beta_R \mathcal{L}_R(e_j; \mathcal{C}_i) + \beta_Q \mathcal{L}_Q(e_j; \mathcal{C}_i)\Big].$$

Here, $\beta_R$ and $\beta_Q$ are the corresponding loss weights in TD-MPC2's learning objective.

### 3.3 Exemplar-Based Component Distillation

Adapting the shared model to new tasks can alter the representations, predictions, and policy behavior learned for previous tasks. To retain these capabilities, we keep the $K$ trajectories with the highest episodic returns from each previously learned task in an exemplar memory $\mathcal{B}_{\text{pre}}$. The original model, together with its task embeddings, provides supervision through its outputs on these exemplars, without further interaction with the previous task environments.

We introduce modular distillation losses to measure differences between the outputs of the adapted and original models. We define $\mathcal{L}_h^{\text{dist}}$ and $\mathcal{L}_d^{\text{dist}}$ for discrepancies in encoder representations and one-step dynamics predictions, and $\mathcal{L}_R^{\text{dist}}$ and $\mathcal{L}_Q^{\text{dist}}$ for discrepancies in reward and value predictions. The policy distillation loss $\mathcal{L}_\pi^{\text{dist}}$ measures the KL divergence between their policy priors. The policy and model objectives, with model-component weights $\beta_{\text{dyn}}, \beta_R, \beta_Q$, are

$$\mathcal{L}_\pi^{\text{dist}} = \mathbb{E}_{\mathcal{B}_{\text{pre}}}\left[D_{\text{KL}}(\bar{\pi}(\cdot \mid \bar{z}) \,\|\, \pi_\theta(\cdot \mid \bar{z}))\right],$$
$$\mathcal{L}_{\text{model}}^{\text{dist}} = \frac{\beta_{\text{dyn}}}{2}\left(\mathcal{L}_h^{\text{dist}} + \mathcal{L}_d^{\text{dist}}\right) + \beta_R \mathcal{L}_R^{\text{dist}} + \beta_Q \mathcal{L}_Q^{\text{dist}}.$$

Here, $\bar{\pi}$ denotes the original policy prior, and $\bar{z}$ is the original model's latent encoding of an exemplar observation.

### 3.4 Joint Multitask Adaptation

For each new task, the two embeddings retrieved in Sec. 3.2 initialize separate trainable embeddings for the dynamics-related and task-related components. The agent then gathers additional experience from the $n$ new tasks and stores it in a replay buffer $\mathcal{B}_{\text{new}}$. Collection and sampling are balanced across tasks so that each contributes equally to new-task learning.

We use TD-MPC2's model and policy learning objectives, evaluated with each new task's two embeddings. The model objective combines latent consistency, reward prediction, and temporal-difference losses, while the policy objective improves the policy prior using value estimates and

entropy regularization. Their averages over task-balanced batches from $\mathcal{B}_{\text{new}}$ are denoted by $\mathcal{L}^{\text{new}}_{\text{model}}$ and $\mathcal{L}^{\text{new}}_{\text{policy}}$, respectively.

We combine new-task learning with capability retention through

$$\mathcal{L}_{\text{model}} = \mathcal{L}^{\text{new}}_{\text{model}} + \lambda_{\text{model}}\mathcal{L}^{\text{dist}}_{\text{model}},$$
$$\mathcal{L}_{\text{policy}} = \mathcal{L}^{\text{new}}_{\text{policy}} + \lambda_{\text{policy}}\mathcal{L}^{\text{dist}}_{\pi}.$$

Here, $\lambda_{\text{model}}$ and $\lambda_{\text{policy}}$ are hyperparameters. Based on these objectives, we train a single agent to acquire multiple new tasks simultaneously while retaining its previous capabilities.

### 3.5 Weight Adaptation with Fixed SVD Bases

To optimize the objectives in Sec. 3.4, we follow prior SVD-based adaptation methods (Lingam et al., 2024; Ge et al., 2026) and apply singular value decomposition to selected linear weight matrices of the shared model parameterized by $\theta$. For each matrix $W$, we can select $r$ singular directions and write its rank-$r$ approximation and adapted form $W'$ as

$$W \approx U\Sigma V^\top, \qquad W' = UCV^\top, \qquad C_{\text{init}} = \Sigma.$$

Here, $U$ and $V$ contain the selected singular vectors, and $\Sigma$ is the diagonal matrix of corresponding singular values. We keep $U$ and $V$ fixed throughout adaptation. The full core matrix $C$ is trainable and initialized to $\Sigma$. Learning both diagonal and off-diagonal entries of $C$ allows the model to adjust and combine these pretrained directions. We optimize the core matrices under the objectives in Sec. 3.4, combining new-task learning with exemplar-based distillation. Implementation details are provided in Appendix A.

## 4 Experiments

We evaluate our framework on diverse continuous control tasks from the DeepMind Control Suite (Tassa et al., 2018) and Meta-World (Yu et al., 2019), spanning classical control, locomotion, and robotic manipulation. We start from TD-MPC2 models (Hansen et al., 2024) trained on 70 tasks, using released checkpoints where available, and adapt them to 10 new tasks simultaneously. Figure 2 visualizes these ten new tasks. The original 70 tasks are presented in Appendix B. Implementation details are provided in Appendix A.

We seek to answer the following questions through experimentation:

- Starting from a multitask world model trained on 70 tasks, can the representations retrieved in Sec. 3.2 provide an effective initialization for learning the 10 new tasks?
- Can our method accelerate learning of the 10 new tasks while largely preserving performance on the 70 previously learned tasks?
- How does each component introduced in Sec. 3 influence performance?
- How does our method scale with increasing model size?

### 4.1 Baselines

We compare MARR, which builds on multitask TD-MPC2 (Hansen et al., 2024), with direct fine-tuning and three additional baselines based on multitask Soft Actor-Critic (Multi-SAC) (Haarnoja et al., 2018). All baselines start from models trained on the 70 original tasks using model-based or model-free RL and adapt to the 10 new tasks simultaneously. Table 1 shows that TD-MPC2 starts with lower average performance on the original 70 tasks than Multi-SAC. We expect MARR to achieve better new-task adaptation despite this weaker trained model.

Table 1: **Performance on the original 70 tasks.** Mean normalized scores before adaptation, evaluated over 10 episodes per task.

| Model | Parameters | 70 tasks |
|---|---|---|
| Multi-SAC | ≈5M | 56.16 |
| TD-MPC2 | 5M | 47.99 |

(1) **Multi-SAC with ClonEx** adapts ClonEx-SAC (Wołczyk et al., 2022) to our multitask setting. It selects a previously learned policy based on its return on each new task to guide initial exploration, and applies behavioral cloning on old-task exemplars to mitigate forgetting.

(2) **Multi-SAC with Retrieval and EWC** initializes each new task's actor and critic embeddings from a previously learned task selected by minimizing the SAC temporal-difference loss on the probing context. During adaptation, Elastic Weight Consolidation (EWC) (Kirkpatrick et al., 2017) penalizes changes to parameters important for the original tasks.

(3) **Fine-tuning Multi-SAC(Multi-SAC FT)** randomly initializes the new task embeddings and fine-tunes the model on the 10 new tasks using the standard SAC objectives.

(4) **Fine-tuning TD-MPC2 with Retrieval (TD-MPC2 FT with retrieval)** starts from the same TD-MPC2 model as MARR. Following the task-embedding initialization strategy used in TD-MPC2's fine-tuning experiments (Hansen et al., 2024), it retrieves the embedding of a semantically similar previously learned task for each new task and uses the same embedding to condition all model components. It then fine-tunes the model on the 10 new tasks using the original TD-MPC2 learning objectives.

### 4.2 RESULTS

**Zero shot performance.** MARR provides a stronger initialization for adaptation. Before gradient updates, it already produces more effective behavior on the new tasks than the baseline initialization procedures. Table 2 further shows that the selected embeddings often come from tasks with related embodiments or behaviors. For dynamics-related components, shared embodiments may lead to overlapping state distributions, allowing the encoder to map new-task observations to familiar latent regions and reuse learned dynamics. For task-related components, similar reward structures and control behaviors may support reuse of reward and value estimates to evaluate imagined trajectories, while the retrieved policy prior guides action sampling during planning. Appendix C reports the retrieval results for all three seeds.

Table 2: **Retrieved original tasks' embeddings for the ten new tasks.** Each entry identifies the original task whose embedding is retrieved for a new task, separately for Dynamics-related and Task-related components.

| | **Meta-World** | | | | | **DeepMind Control** | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **New task** | Bin Picking | Box Close | Door Lock | Door Unlock | Hand Insert | Walker Run | Cheetah Run | Reacher Hard | Pendulum Swingup | Hopper Hop |
| Dynamics-related | Coffee Pull | Pick Place Wall | Basketball | Pick Out of Hole | Coffee Push | Walker Walk Backwards | Cheetah Run Front | Reacher Easy | Pendulum Spin | Hopper Stand |
| Task-related | Handle Press | Faucet Open | Drawer Open | Window Close | Push Back | Walker Run Backwards | Cheetah Run Front | Reacher Easy | Pendulum Spin | Hopper Hop Backwards |

**Few shot Adaptation.** During joint adaptation, MARR maintains the highest task-averaged performance on the 10 new tasks among the evaluated methods, suggesting more efficient adaptation. TD-MPC2 FT with retrieval achieves the second-best performance, consistent with the potential benefits of embedding retrieval for world-model adaptation. The Multi-SAC baselines make limited progress within this budget. For task-specific performance, MARR demonstrates sample-efficient learning on most new tasks, as shown in Figure 4. Among the model-free baselines, Multi-SAC with ClonEx also shows some learning capability, although its gains are limited to a subset of tasks. All evaluated methods fail to learn effective behavior on Hopper Hop. This difficulty may reflect the weak Hopper skills acquired during the original 70-task training: both original models perform poorly on the related Hopper Stand and Hopper Hop Backwards tasks, as shown in Table 3, indicating that little useful behavior is available for transfer.

Table 3: **Hopper tasks' performance.** Normalized scores before adaptation, averaged over 10 episodes per task.

| Task | TD-MPC2 | Multi-SAC |
|---|---|---|
| Stand | 0.47 | 0.12 |
| Hop Backwards | 0.22 | 0.00 |

**Retention on original tasks.** We assess retention by comparing the mean normalized score on the original 70 tasks before and after adaptation, reporting the percentage of pre-adaptation performance retained. At 0.6M total adaptation interactions, the ClonEx- and EWC-based baselines retain a

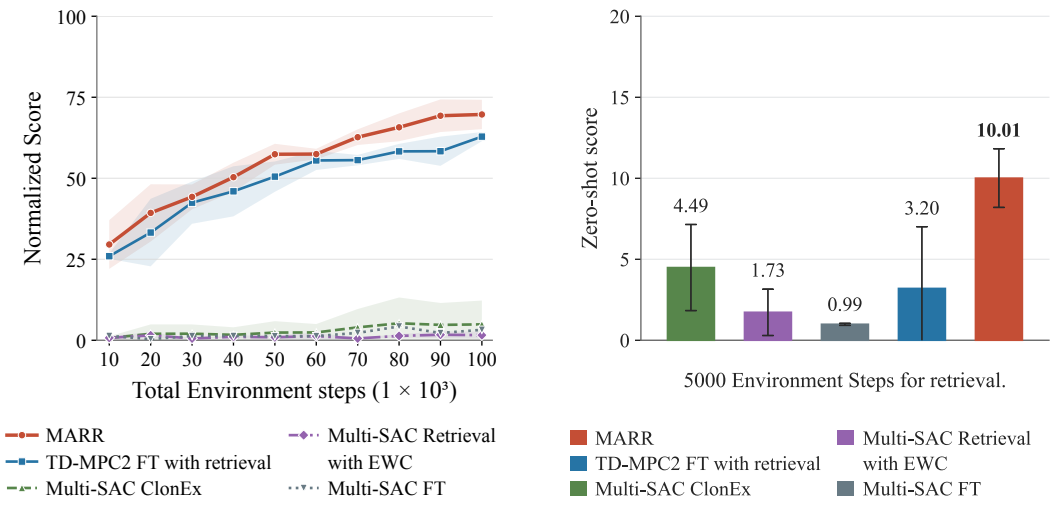


Figure 3: **Adaptation and zero shot performance on the 10 new tasks. Left:** Learning curves with 100k environment steps per task. Normalized scores are averaged over the 10 tasks. **Right:** Zero shot performance on the 10 new tasks, with scores averaged over the 10 tasks. We report mean and standard deviation over three seeds

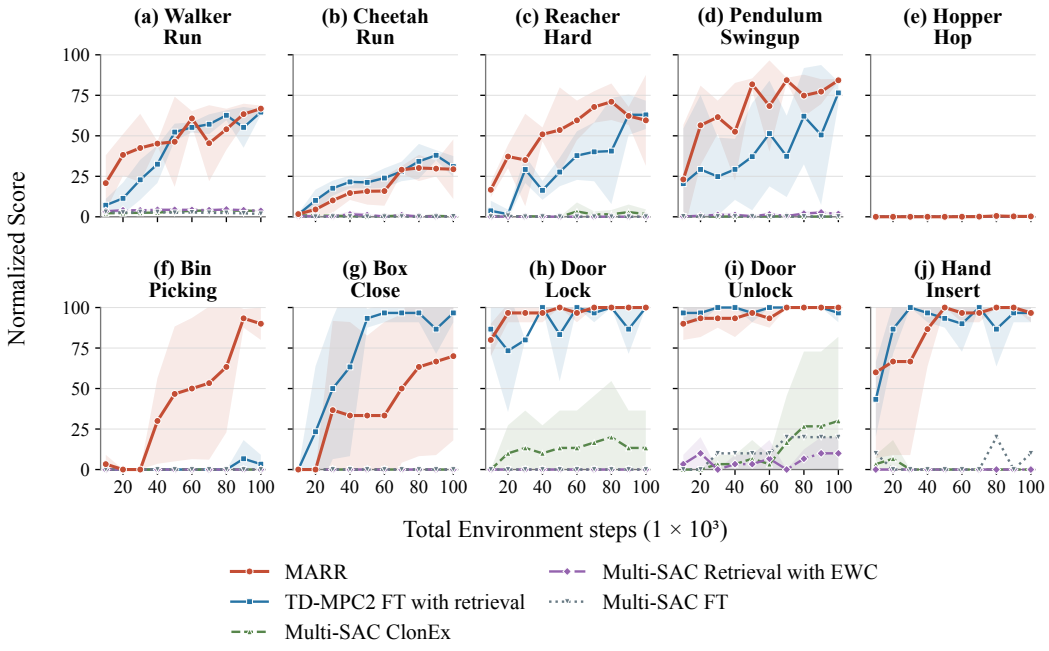


Figure 4: **Learning curves on the 10 new tasks.** Each task is allowed at most 100k environment interactions. We report mean and standard deviation over three seeds, except for the single-seed Multi-SAC FT learning curves.

high proportion of their original-task performance, as shown in Table 4. However, Figures 3 and 4 show that their task-averaged progress on the new tasks remains limited. Strong retention alone therefore does not ensure a favorable balance between acquiring new skills and preserving existing knowledge. At the matched budget, MARR retains most of its original-task performance while achieving stronger new-task performance, providing a more favorable trade-off in our experiments.

Table 4: **Retention on the original 70 tasks.** Percentage of the pre-adaptation mean score retained after 0.6M total adaptation interactions, using 5M models and 10 evaluation episodes per task. We report mean $\pm$ sample standard deviation over three adaptation seeds, except for single-seed Multi-SAC Fine-tuning.

| Method | Retention (%) |
|---|---|
| **MARR (Ours)** | $\mathbf{93.50 \pm 3.93}$ |
| TD-MPC2 Fine-tuning with retrieval | $34.56 \pm 4.46$ |
| Multi-SAC with ClonEx | $90.98 \pm 0.87$ |
| Multi-SAC with Retrieval and EWC | $88.75 \pm 2.91$ |
| Multi-SAC Fine-tuning | 10.63 |

**Scalability.** We increase the model size from 5M to 19M and evaluate the adapted agents across all 80 tasks under the same 1M interaction budget. As shown in Table 5, the larger model achieves higher overall performance after adaptation with capability retention. This result suggests that our method can benefit from increased model capacity, providing preliminary evidence of its scalability.

Table 5: **Model scalability.** All-80 normalized scores.

| 5M | 19M |
|---|---|
| $45.10 \pm 1.80$ | $49.68 \pm 1.32$ |

mean $\pm$ std over 3 seeds

**Ablation.** To investigate the contributions of our method's main mechanisms to adaptation and retention, we evaluate variants without exemplar-based distillation, SVD-based parameterization, or embedding retrieval. Table 6 reports performance on both the 70 original tasks and the 10 new tasks. The results indicate that exemplar-based distillation helps preserve original-task performance when adapting coefficients within fixed SVD bases. Embedding retrieval also benefits new-task learning, yielding higher average performance on the new tasks.

Table 6: **Perfomance for ablations** Normalized scores after 1M total adaptation interactions (100k per new task) with 5M models. We report mean $\pm$ sample standard deviation over three training seeds from separate final-checkpoint evaluations with 10 episodes per task. The no-SVD variant also differs in embedding parameterization and distillation weights.

| Method | Original 70 tasks | New 10 tasks |
|---|---|---|
| MARR (Ours) | $\mathbf{43.45 \pm 2.41}$ | $\mathbf{73.41 \pm 6.88}$ |
| w/o Exemplar Distillation | $17.11 \pm 1.14$ | $68.73 \pm 3.97$ |
| w/o Retrieval | $38.02 \pm 2.31$ | $67.35 \pm 6.60$ |
| w/o SVD Decomposition | $37.13 \pm 0.51$ | $68.91 \pm 5.48$ |

## 5 Conclusion

We presented MARR, a framework for adapting trained multitask world models to multiple new continuous control tasks while retaining prior capabilities. Through brief exploration of each new task, MARR retrieves dynamics-related and task-related representations to provide an effective initialization for adaptation. Adaptation combines learning core weight matrices within fixed SVD bases with exemplar-based distillation to balance new-task learning and knowledge retention. Our exploratory experiments on DMControl and Meta-World show improved new-task learning over a range of adaptation baselines while retaining most of the original-task performance. These findings suggest that reusing and preserving knowledge in existing multitask world models offers a promising route to advancing generalist agents as new tasks become accessible.

## AI USE STATEMENT

We used OpenAI Codex to assist with refining the problem formulation and experimental design, implementing and debugging methods, baselines, and evaluation scripts. It also assisted with literature search, translating and revising manuscript text. The authors direct the research and take responsibility for the final manuscript, implementations, experimental results, and scientific claims, including material produced with AI assistance.

## REFERENCES

Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *European conference on computer vision*, pp. 144–161. Springer, 2018.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent Vanhoucke, Steve Vega, Quan Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning (CoRL)*, 2023.

Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2L: Contrastive continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9516–9525, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Cha_Co2L_Contrastive_Continual_Learning_ICCV_2021_paper.html.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/3de568f8597b94bda53149c7d7f5958c-Abstract.html.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1126–1135, 2017. URL https://proceedings.mlr.press/v70/finn17a.html.

Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999.

Jingze Ge, Xue Geng, Yun Liu, Wanqi Dong, Wang Zhe Mark, Min Wu, Ngai-Man Cheung, Bharadwaj Veeravalli, and Xulei Yang. CERSA: Cumulative energy-retaining subspace adaptation for memory-efficient fine-tuning, 2026. URL https://arxiv.org/abs/2605.08174.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *ICML*, 2018. URL https://arxiv.org/abs/1801.01290.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *ICLR*, 2020.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025. doi: 10.1038/s41586-025-08744-2.

Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. In *ICML*, 2022.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *ICLR*, 2024.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Learning massively multitask world models for continuous control. *arXiv preprint arXiv:2511.19584*, 2025.

Viraj Joshi, Zifan Xu, Bo Liu, Peter Stone, and Amy Zhang. Benchmarking massively parallelized multi-task reinforcement learning for robotics tasks. *Reinforcement Learning Journal*, 6:1293–1317, 2025. URL https://rlj.cs.umass.edu/2025/papers/Paper140.html.

Rituraj Kaushik, Timothée Anne, and Jean-Baptiste Mouret. Fast online adaptation in robotics through meta-learning embeddings of simulated priors. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. URL https://arxiv.org/abs/2003.04663.

Samuel Kessler, Mateusz Ostaszewski, Michał Paweł Bortkiewicz, Mateusz Żarski, Maciej Wolczyk, Jack Parker-Holder, Stephen J. Roberts, and Piotr Miłoś. The effectiveness of world models for continual reinforcement learning. In *Proceedings of the 2nd Conference on Lifelong Learning Agents*, volume 232 of *Proceedings of Machine Learning Research*, pp. 184–204, 2023. URL https://proceedings.mlr.press/v232/kessler23a.html.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, et al. DROID: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024. URL https://arxiv.org/abs/2403.12945.

Khimya Khetarpal, Matthew Riemer, Irina Rish, and Doina Precup. Towards continual reinforcement learning: A review and perspective. *Journal of Artificial Intelligence Research*, 75:1401–1476, 2022.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 2017. doi: 10.1073/pnas.1611835114. URL https://arxiv.org/abs/1612.00796.

Vijay Lingam, Atula Tejaswi, Aditya Vavre, Aneesh Shetty, Gautham Krishna Gudur, Joydeep Ghosh, Alex Dimakis, Eunsol Choi, Aleksandar Bojchevski, and Sujay Sanghavi. SVFT: Parameter-efficient fine-tuning with singular vectors. In *Advances in Neural Information Processing Systems*, volume 37, pp. 41425–41446, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/48c368f105e8145b945227b73255635a-Abstract-Conference.html.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/f87522788a2be2d171666752f97ddebb-Abstract.html.

Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7765–7773. IEEE, 2018.

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. In *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pp. 1678–1690, 2022. URL https://proceedings.mlr.press/v164/mandlekar22a.html.

Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of Learning and Motivation*, 24:109–165, 1989.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Lawrence Yunliang Chen, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy. In *Robotics: Science and Systems (RSS)*, 2024.

Kate Rakelly, Aurick Zhou, Deirdre Quillen, Chelsea Finn, and Sergey Levine. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5331–5340, 2019. URL https://proceedings.mlr.press/v97/rakelly19a.html.

Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley Edwards, Nicolas Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar, and Nando de Freitas. A generalist agent. *TMLR*, 2022.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in neural information processing systems*, 32, 2019.

Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30, 2017.

Shagun Sodhani, Amy Zhang, and Joelle Pineau. Multi-task reinforcement learning with context-based representations. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9767–9779, 2021. URL https://proceedings.mlr.press/v139/sodhani21a.html.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, Timothy Lillicrap, and Martin Riedmiller. DeepMind Control Suite. *arXiv preprint arXiv:1801.00690*, 2018.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5362–5383, 2024. doi: 10.1109/TPAMI.2024.3367329. URL https://arxiv.org/abs/2302.00487.

Maciej Wołczyk, Michał Zając, Razvan Pascanu, Łukasz Kuciński, and Piotr Miłoś. Continual world: A robotic benchmark for continual reinforcement learning. In *NeurIPS*, 2021.

Maciej Wołczyk, Michał Zając, Razvan Pascanu, Łukasz Kuciński, and Piotr Miłoś. Disentangling transfer in continual reinforcement learning. In *NeurIPS*, 2022. URL https://arxiv.org/abs/2209.13900.

Zhou Xian, Shamit Lal, Hsiao-Yu Fish Tung, Emmanouil Antonios Platanios, and Katerina Fragkiadaki. HyperDynamics: Meta-learning object and agent dynamics with hypernetworks. In *ICLR*, 2021. URL https://openreview.net/forum?id=pHXfe1cOmA.

Luke Yang, Levin Kuhlmann, and Gideon Kowadlo. Augmenting replay in world models for continual reinforcement learning. *arXiv preprint arXiv:2401.16650*, 2024. URL https://arxiv.org/abs/2401.16650.

Ruihan Yang, Huazhe Xu, Yi Wu, and Xiaolong Wang. Multi-task reinforcement learning with soft modularization. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/32cfdce9631d8c7906e8e9d6e68b514b-Abstract.html.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-World: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2019.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/3fe78a8acf5fda99de95303940a2420c-Abstract.html.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pp. 3987–3995. Pmlr, 2017.

Hongtu Zhou, Ruiling Yang, Yakun Zhu, Haoqi Zhao, Hai Zhang, Di Zhang, Junqiao Zhao, Chen Ye, and Changjun Jiang. CERTAIN: Context uncertainty-aware one-shot adaptation for context-based offline meta reinforcement learning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79529–79547, 2025. URL https://proceedings.mlr.press/v267/zhou25ai.html.

# APPENDIX CONTENTS

A Algorithm and Implementation Details . . . 13
A.1 Algorithm Overview . . . 13
A.2 Algorithm Implementations . . . 14
A.3 SVD Basis Counts . . . 14
B Original Task Suite . . . 15
B.1 DeepMind Control Tasks . . . 15
B.2 Meta-World Tasks . . . 16
C Embedding Retrieval and Task Representations . . . 18
D Task-wise Performance Before and After Adaptation . . . 20

# A ALGORITHM AND IMPLEMENTATION DETAILS

## A.1 ALGORITHM OVERVIEW

Algorithm 1 summarizes the TD-MPC2 instantiation of MARR.

**Algorithm 1 MARR: retrieval, adaptation, and retention**

**Input:** Original model $\bar{f} = f(\theta_0, E_0)$; exemplar memory $\mathcal{B}_{\text{pre}}$; new tasks $\mathcal{T}_{\text{new}}$; candidate sets $\mathcal{E}_i \subseteq \mathcal{E}$; interaction budget.

Initialize the adapted model from $\bar{f}$; keep $\bar{f}$ and $E_0$ fixed.
For each adapted weight $W = U\Sigma V^\top$, set $W' = UCV^\top$, $C \leftarrow \Sigma$.
Train dense cores and task embeddings; freeze bases and other parameters.
Cache the original component outputs on $\mathcal{B}_{\text{pre}}$ as fixed targets.
Initialize new-task replay $\mathcal{B}_{\text{new}} \leftarrow \emptyset$.
**for each** new task $\mathcal{M}_i \in \mathcal{T}_{\text{new}}$ **do**
  Collect random-action context $\mathcal{C}_i$ within the interaction budget.
  $e_i^{\text{dyn}} \leftarrow \arg\min_{e_j \in \mathcal{E}_i} \mathcal{L}_{\text{dyn}}(e_j; \mathcal{C}_i)$.
  Hold $e_i^{\text{dyn}}$ fixed for $(h, d)$ while scoring control candidates.
  $e_i^{\text{ctrl}} \leftarrow \arg\min_{e_j \in \mathcal{E}_i} [\beta_R \mathcal{L}_R(e_j; \mathcal{C}_i) + \beta_Q \mathcal{L}_Q(e_j; \mathcal{C}_i)]$.
  Copy these into trainable rows for $(h, d)$ and $(R, Q, \pi)$, respectively.
  Add $\mathcal{C}_i$ to $\mathcal{B}_{\text{new}}$.
**end for**
**for each** scheduled learning update **do**
  When scheduled, collect equally from all new tasks using the current planner; append this experience to $\mathcal{B}_{\text{new}}$ within the remaining budget.
  Sample task-balanced new-task and exemplar batches from their buffers.
  Compute TD-MPC2 losses $\mathcal{L}^{\text{new}}_{\text{model}}$, $\mathcal{L}^{\text{new}}_{\text{policy}}$ on the new-task batch.
  Compute $\mathcal{L}^{\text{dist}}_{\text{model}}$ and $\mathcal{L}^{\text{dist}}_{\pi}$ from fixed exemplar targets (Sec. 3.3).
  $\mathcal{L}_{\text{model}} \leftarrow \mathcal{L}^{\text{new}}_{\text{model}} + \lambda_{\text{model}} \mathcal{L}^{\text{dist}}_{\text{model}}$.
  $\mathcal{L}_{\text{policy}} \leftarrow \mathcal{L}^{\text{new}}_{\text{policy}} + \lambda_{\text{policy}} \mathcal{L}^{\text{dist}}_{\pi}$.
  Update world-model cores and all old/new embeddings using $\mathcal{L}_{\text{model}}$.
  Update policy cores using $\mathcal{L}_{\text{policy}}$; update the target value network.
**end for**
**return** The adapted model for $\mathcal{T}_{\text{pre}} \cup \mathcal{T}_{\text{new}}$.

The three-seed 5M adaptation experiments restrict $\mathcal{E}_i$ to original tasks with the same environment family and action dimension (e.g., Walker tasks); Meta-World tasks are grouped together.

### A.2 Algorithm Implementations

MARR collects 500 random-action transitions per new task as exploration. Each candidate embedding is scored on the same context as described in Sec. 3.2; context collection counts toward the adaptation budget. Multi-SAC with ClonEx receives the same budget of 500 environment interactions per task in the initialization comparison. It evaluates the 70 policies learned on the original tasks and selects the highest-return policy as the initial behavior policy for the new task. The memory $\mathcal{B}_{\text{pre}}$ retains the four highest-return trajectories from each original task. These trajectories, paired with the corresponding outputs of each component of the original model, form the exemplars. For distillation, we balance exemplar sampling across tasks. For each adapted weight matrix $W_\ell \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, we retain all $r_\ell = \min(d_{\text{out}}, d_{\text{in}})$ singular directions, preserving all singular-value energy. We freeze the left and right bases and optimize each dense $r_\ell \times r_\ell$ core, initialized to the diagonal singular-value matrix. Further details are provided in Table 7.

Table 7: **Interaction and exemplar counts.** Totals cover the ten new tasks for interaction budgets and the 70 original tasks for memory. Domain-specific rows cover the five new tasks in that domain.

| Quantity | Per task | Total |
|---|---|---|
| *MARR retrieval* | | |
| DMC random-action trajectories | $1 \times 500$ steps | 5 trajectories |
| Meta-World random-action trajectories | $5 \times 100$ steps | 25 trajectories |
| Collected context | 500 interactions | 5,000; 30 trajectories |
| *ClonEx initialization comparison* | | |
| Selection budget cap | 500 interactions | 5,000 interactions |
| Actual policy trials | $70 \times 7$ steps | 4,900 interactions |
| *ClonEx adaptation curves* | | |
| Full-episode trials, DMC | $70 \times 500$ steps | 175,000 interactions |
| Full-episode trials, Meta-World | $70 \times 100$ steps | 35,000 interactions |
| Subsequent best-policy exploration | 10,000 steps | 100,000 interactions |
| Updates begin at | 1,000 interactions | — |
| *Exemplar memory* | | |
| Retained trajectory records | $4 \times 100$ transitions | 280 records |
| Retained transitions | 400 | 28,000 |
| ClonEx pre-transition observations | 400 | 28,000 |
| MARR distillation window | 3 transitions | — |

### A.3 SVD Basis Counts

Table 8: SVD basis counts for the 5M model. Retained singular directions are listed in layer order. The Q row repeats the three listed ranks for each of five heads; its matrix and direction counts include all five. Each retained direction corresponds to one left–right singular-vector pair.

| Component | Matrices | Directions per matrix | Total pairs |
|---|---|---|---|
| Encoder | 2 | $(135, 256)$ | 391 |
| Dynamics | 3 | $(512, 512, 512)$ | 1,536 |
| Reward predictor | 3 | $(512, 512, 101)$ | 1,125 |
| Policy prior | 3 | $(512, 512, 12)$ | 1,036 |
| Q ensemble (5 heads) | 15 | $(512, 512, 101)$ per head | 5,625 |
| Total | 26 | — | 9,713 |

The dense $r_\ell \times r_\ell$ cores contain 4,601,559 trainable coefficients in total; task embeddings are not SVD-factorized.

# B ORIGINAL TASK SUITE

The original task set contains 25 DeepMind Control tasks and 45 Meta-World tasks. Figures 5–7 show all 70 tasks and their names. These tasks are disjoint from the ten new tasks in Figure 2; together, the two sets contain 30 DeepMind Control tasks and 50 Meta-World tasks.

## B.1 DEEPMIND CONTROL TASKS

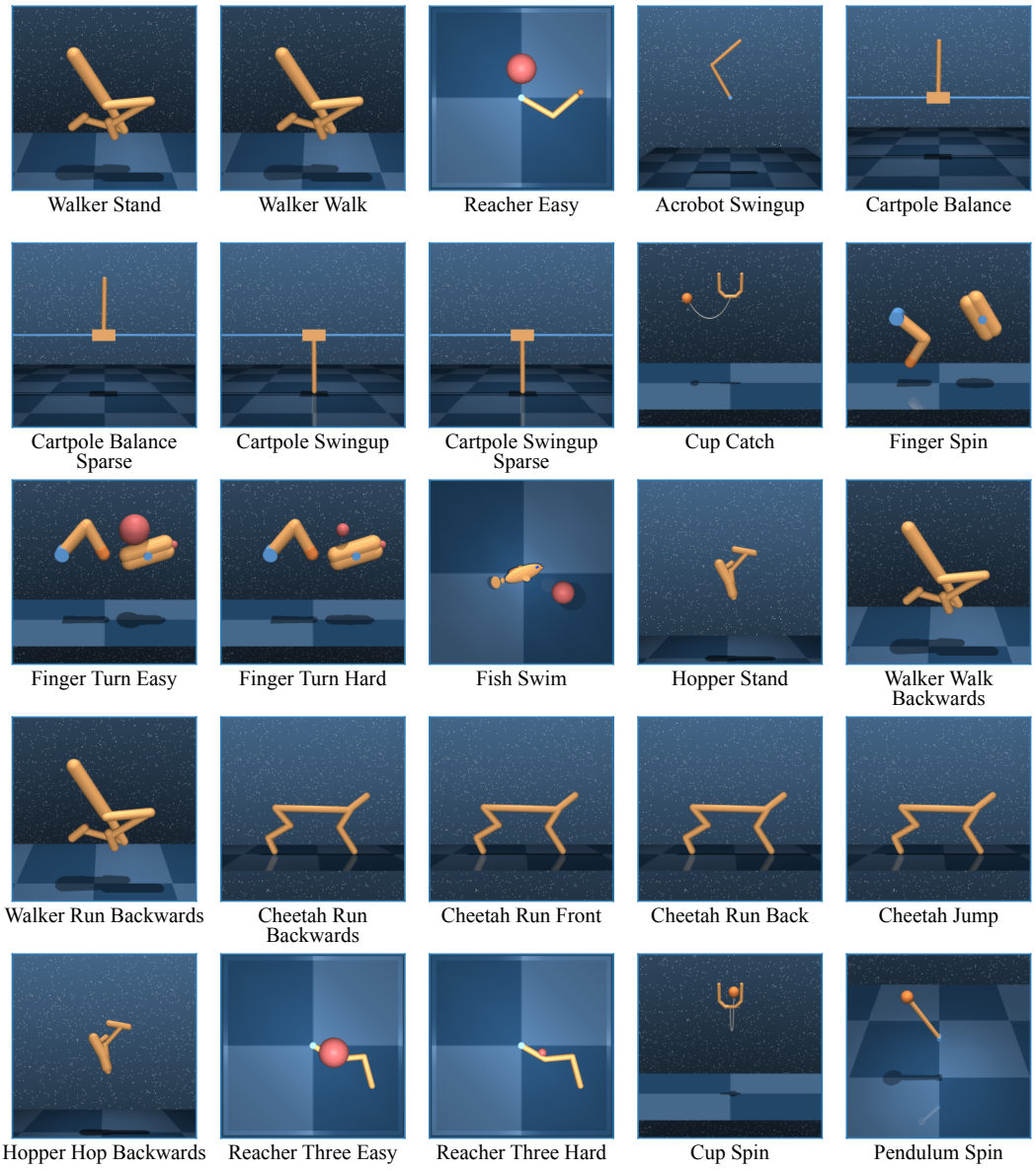


Figure 5: **Original DeepMind Control tasks.** The 25 tasks included in the original 70-task training set.

### B.2 META-WORLD TASKS

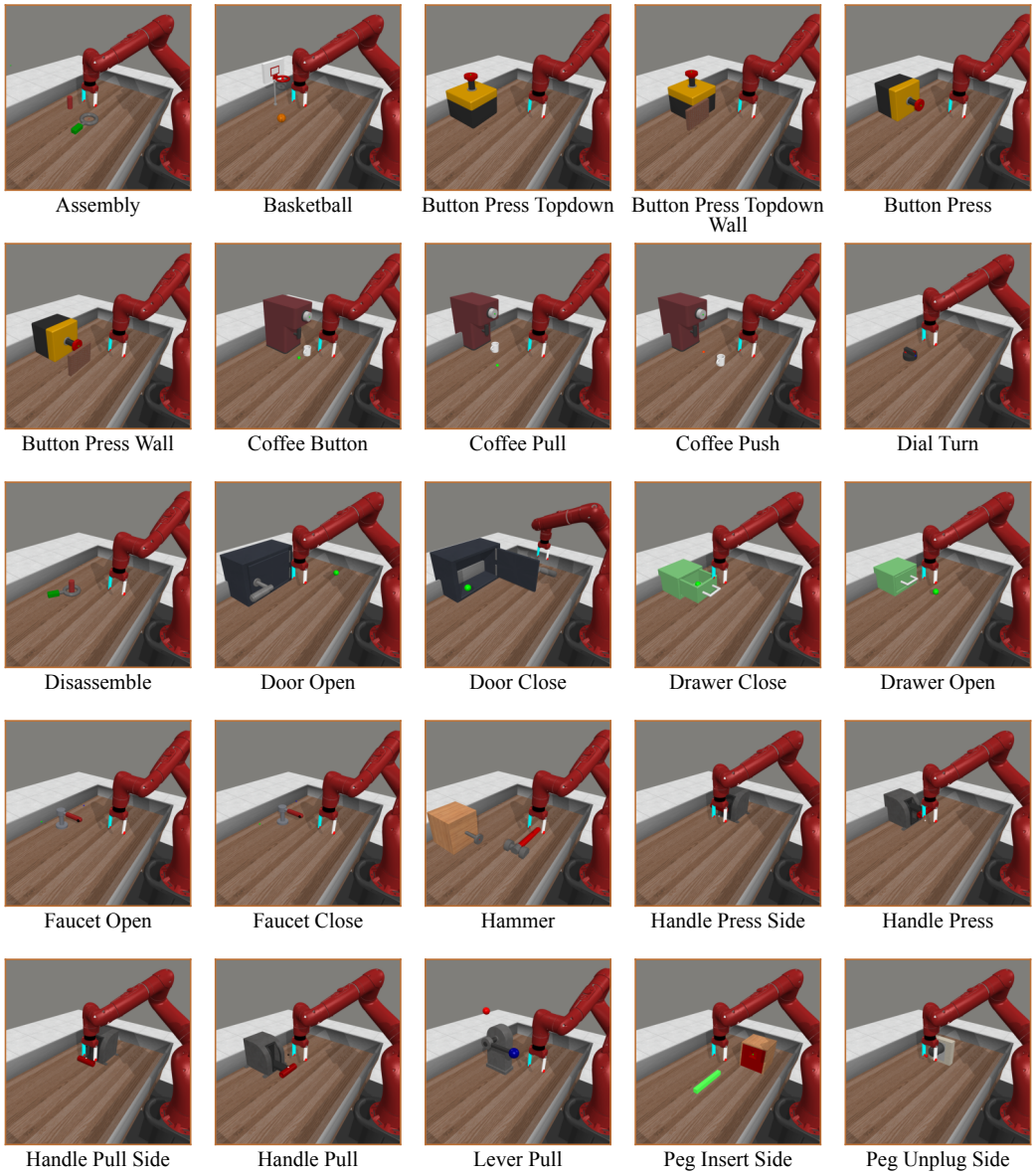


Figure 6: **Original Meta-World tasks, part 1.** The first 25 of the 45 manipulation tasks included in the original training set.

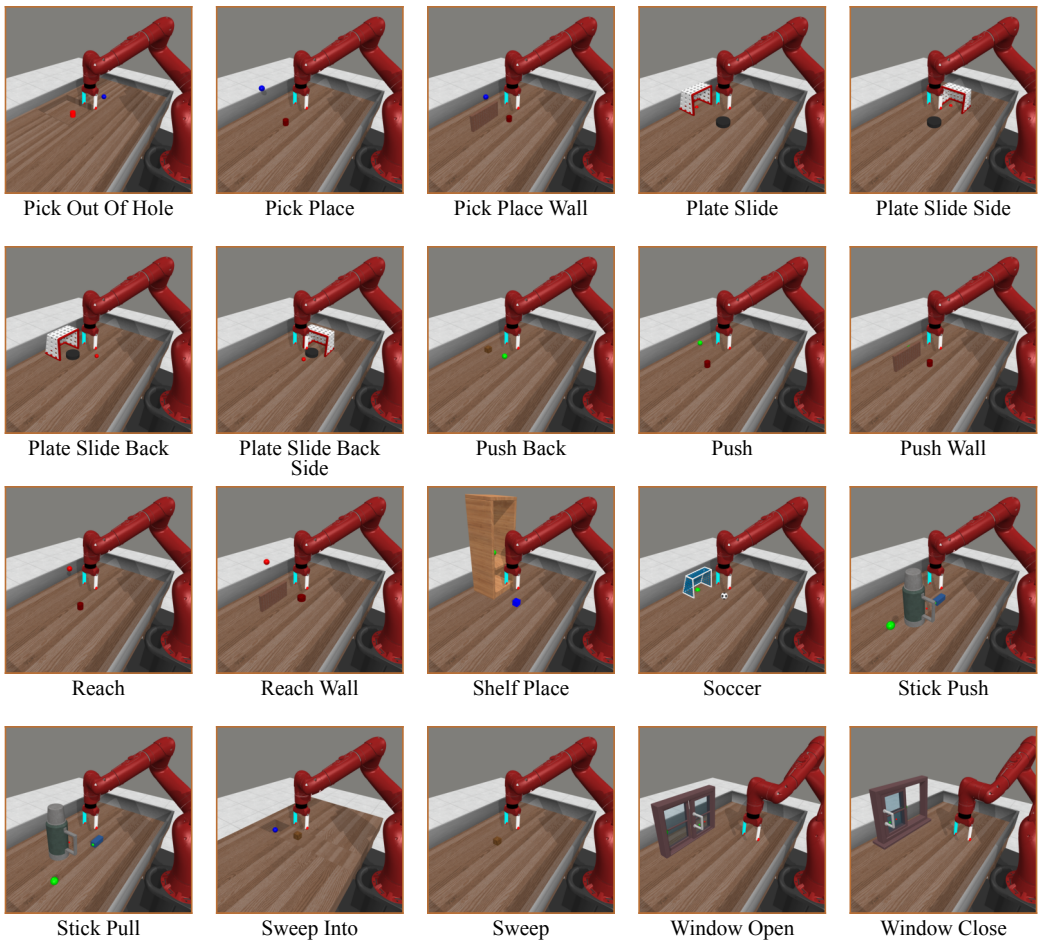


Figure 7: **Original Meta-World tasks, part 2.** The remaining 20 manipulation tasks included in the original training set.

## C Embedding Retrieval and Task Representations

**Sensitivity to random seeds.** Random probing contexts and stochastic candidate scoring can change the retrieval ranking. We repeat retrieval with seeds 1–3, keeping the trained 5M model and the budget of 500 transitions per new task fixed, to assess the sensitivity of the selected embeddings and their initial performance.

Tables 9 and 10 report zero-update performance and the selected source tasks. Dynamics-related choices agree across all three seeds for nine of ten tasks, compared with six for task-related choices; all disagreements occur on Meta-World tasks. Score variability includes both retrieval and evaluation randomness.

Table 9: **Initial performance with retrieved embeddings.** Mean normalized score across the ten new tasks, before gradient updates.

| Seed 1 | Seed 2 | Seed 3 | Mean $\pm$ SD |
|---|---|---|---|
| 11.03 | 7.93 | 11.07 | $10.01 \pm 1.80$ |

Table 10: **Retrieved original-task embeddings across three seeds.** Dynamics-related and Task-related selections for the ten new tasks using the 5M model (adaptation seeds 1–3).

**Seed 1**

| | Meta-World | | | | | DeepMind Control | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **New task** | Bin Picking | Box Close | Door Lock | Door Unlock | Hand Insert | Walker Run | Cheetah Run | Reacher Hard | Pendulum Swingup | Hopper Hop |
| Dynamics-related | Coffee Push | Pick Place Wall | Basketball | Pick Out of Hole | Coffee Push | Walker Walk Backwards | Cheetah Run Front | Reacher Easy | Pendulum Spin | Hopper Stand |
| Task-related | Stick Push | Reach Wall | Drawer Open | Window Close | Push Back | Walker Run Backwards | Cheetah Run Front | Reacher Easy | Pendulum Spin | Hopper Hop Backwards |

**Seed 2**

| | Meta-World | | | | | DeepMind Control | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **New task** | Bin Picking | Box Close | Door Lock | Door Unlock | Hand Insert | Walker Run | Cheetah Run | Reacher Hard | Pendulum Swingup | Hopper Hop |
| Dynamics-related | Coffee Pull | Pick Place Wall | Basketball | Pick Out of Hole | Coffee Push | Walker Walk Backwards | Cheetah Run Front | Reacher Easy | Pendulum Spin | Hopper Stand |
| Task-related | Button Press Wall | Button Press Topdown Wall | Window Close | Button Press Topdown Wall | Push Back | Walker Run Backwards | Cheetah Run Front | Reacher Easy | Pendulum Spin | Hopper Hop Backwards |

**Seed 3**

| | Meta-World | | | | | DeepMind Control | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **New task** | Bin Picking | Box Close | Door Lock | Door Unlock | Hand Insert | Walker Run | Cheetah Run | Reacher Hard | Pendulum Swingup | Hopper Hop |
| Dynamics-related | Coffee Pull | Pick Place Wall | Basketball | Pick Out of Hole | Coffee Push | Walker Walk Backwards | Cheetah Run Front | Reacher Easy | Pendulum Spin | Hopper Stand |
| Task-related | Handle Press | Faucet Open | Drawer Open | Window Close | Push Back | Walker Run Backwards | Cheetah Run Front | Reacher Easy | Pendulum Spin | Hopper Hop Backwards |

**Effect of the probing context.** To isolate variation due to the probing data, we also evaluate a fixed 19M model with three independently sampled 500-transition contexts per task. Candidate-scoring randomness and the ten evaluation episodes are held fixed across contexts. Table 11 compares retrieval within the same environment family and action dimension with retrieval over all 70 original tasks. The variation across contexts shows why a single probing seed is insufficient to characterize initialization.

Table 11: **Retrieval performance across probing contexts.** Mean normalized score on the ten new tasks before gradient updates with the 19M model. SD is across the three context seeds.

| Candidate pool | Context 1 | Context 2 | Context 3 | Mean $\pm$ SD |
|---|---|---|---|---|
| Same family and action dimension | 15.17 | 6.17 | 8.22 | $9.86 \pm 4.72$ |
| All 70 original tasks | 15.48 | 6.49 | 14.53 | $12.17 \pm 4.93$ |

**Task representations after adaptation.** Figure 8 visualizes the task embeddings of the 19M model after 0.5M total new-task interactions (seed 1). A single t-SNE projection contains the 70 original-task vectors and the two vectors for each of the ten new tasks. The panels share the original-task coordinates and show the corresponding new dynamics-related or task-related vectors. For example, Bin Picking lies near Stick Pull in the dynamics view and Push Back in the task-related view, illustrating the different task relationships represented by the two component groups.

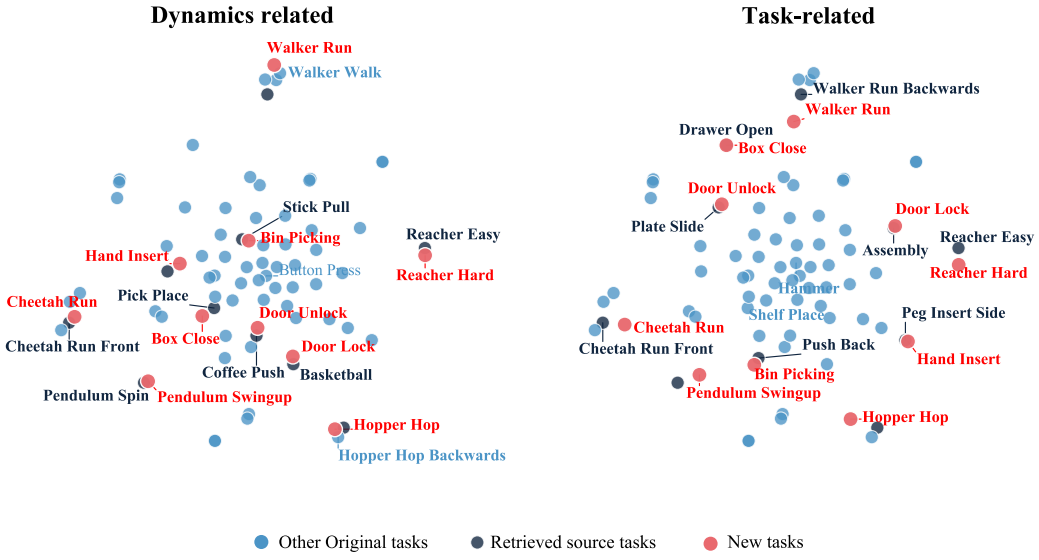


Figure 8: **Task embeddings after adaptation.** Dynamics-related and task-related views of one t-SNE projection of the 96-dimensional embeddings. Red denotes new tasks; dark blue marks original tasks selected for retrieval initialization; light blue marks other original tasks.

## D TASK-WISE PERFORMANCE BEFORE AND AFTER ADAPTATION

Figures 9 and 10 show all 70 original tasks for MARR (5M) before and after 0.6M total new-task interactions (60k per new task). We plot episodic returns; for Meta-World, these differ from the success rates underlying the normalized retention scores.

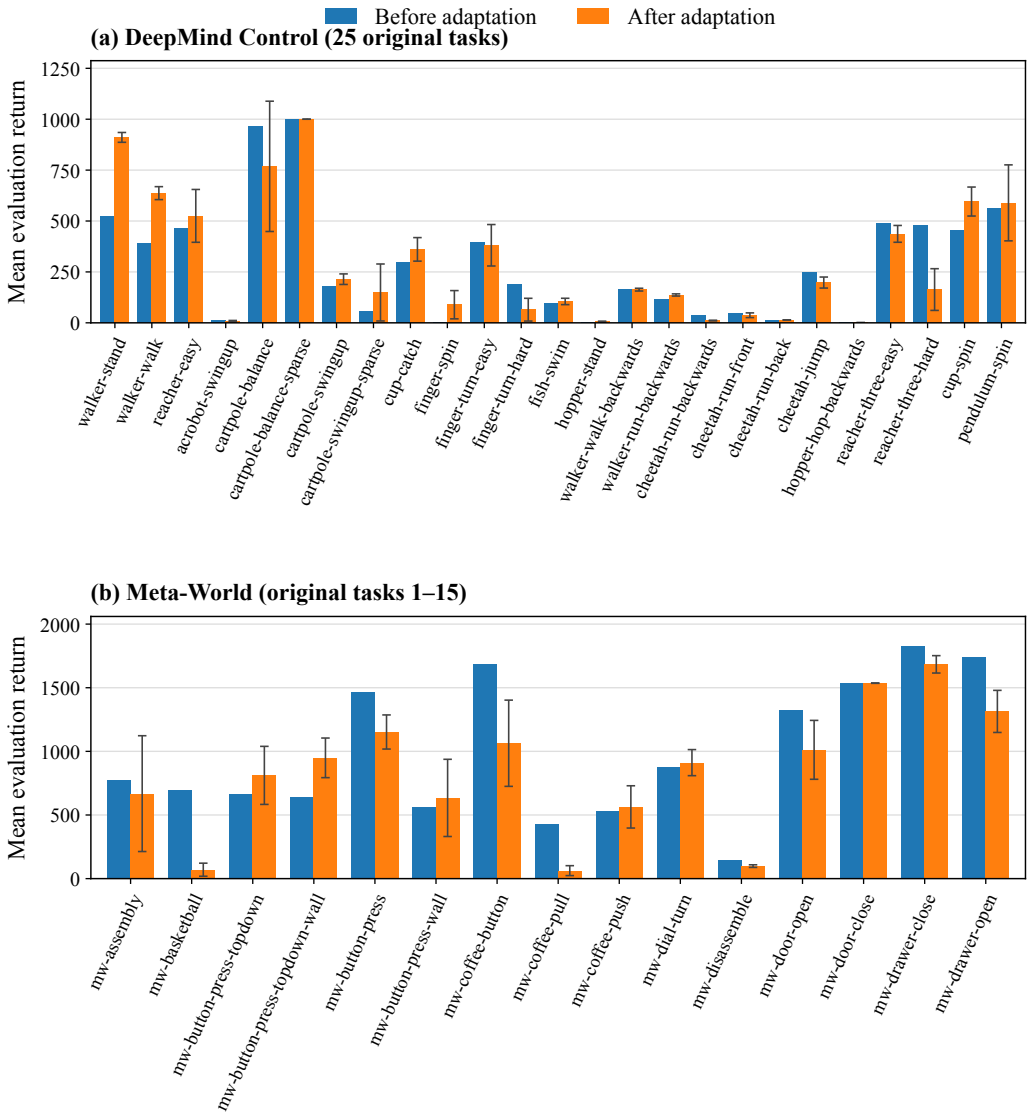


Figure 9: **Original-task performance before and after adaptation, part 1.** All 25 DeepMind Control tasks and the first 15 Meta-World tasks. Blue: the shared original-model reference, evaluated once without seed error bars. Orange: mean and sample standard deviation over three adaptation seeds. Each evaluation uses 10 episodes per task, separately from adaptation.

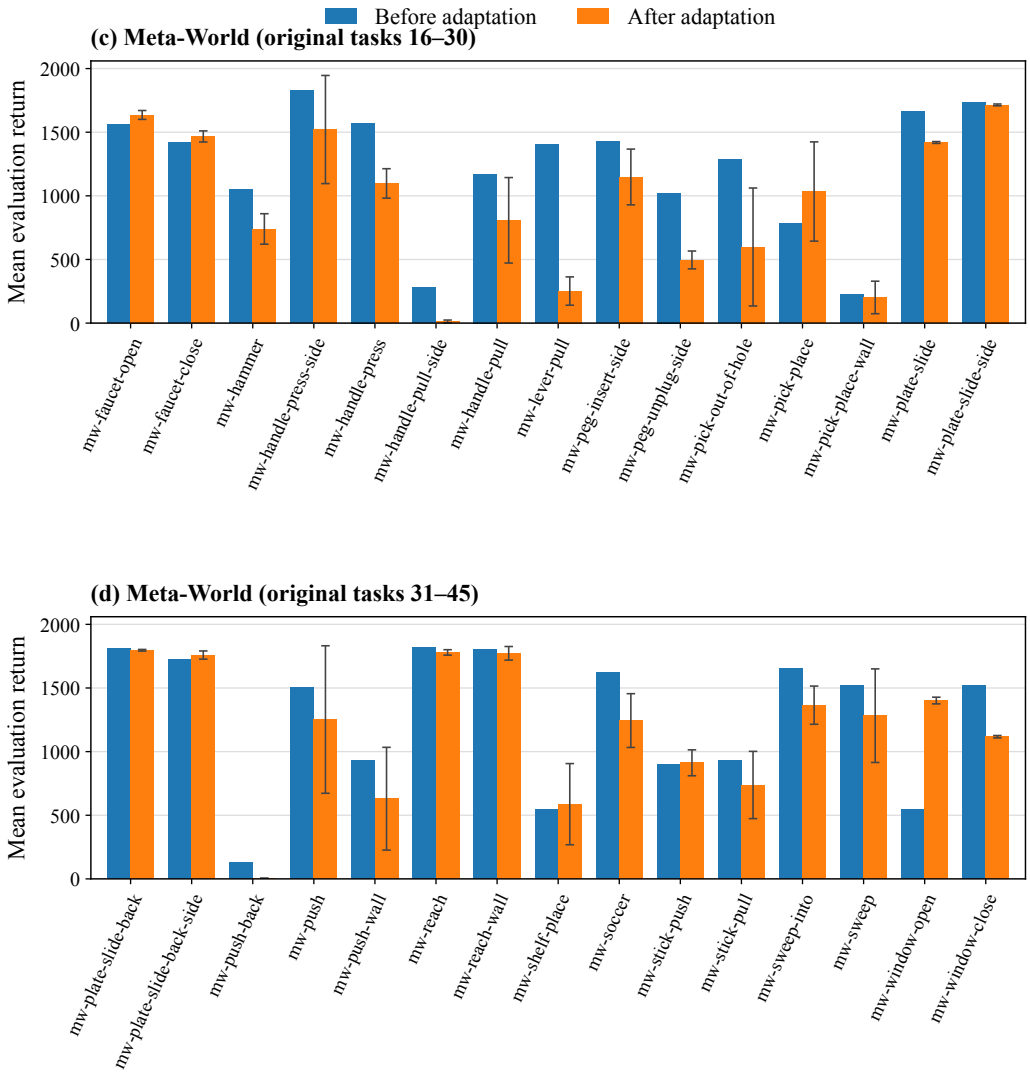


Figure 10: **Original-task performance before and after adaptation, part 2.** The remaining 30 Meta-World tasks, using the same protocol, colors, and uncertainty convention as Figure 9. All three Meta-World panels use the same vertical scale.