One-Shot Robust Imitation Learning for Long-Horizon Visuomotor Tasks from Unsegmented Demonstrations

Shaokang Wu, Yijin Wang, and Yanlong Huang*

***Abstract*— In contrast to single-skill tasks, long-horizon tasks play a crucial role in our daily life, e.g., a pouring task requires a proper concatenation of reaching, grasping and pouring subtasks. As an efficient solution for transferring human skills to robots, imitation learning has achieved great progress over the last two decades. However, when learning long-horizon visuomotor skills, imitation learning often demands a large amount of semantically segmented demonstrations. Moreover, the performance of imitation learning could be susceptible to external perturbation and visual occlusion. In this paper, we exploit dynamical movement primitives and meta-learning to provide a new framework for imitation learning, called Meta-Imitation Learning with Adaptive Dynamical Primitives (MiLa). MiLa allows for learning unsegmented long-horizon demonstrations and adapting to unseen tasks with a single demonstration. MiLa can also resist external disturbances and visual occlusion during task execution. Real-world robotic experiments demonstrate the superiority of MiLa, irrespective of visual occlusion and random perturbations on robots.**

I. Introduction

Learning long-horizon visuomotor tasks is challenging due to dynamical visual observations and long-horizon decision-making processes. In the context of imitation learning, many methods have been developed towards solving long-horizon tasks [1]–[3]. However, these approaches may show limited generalization to new objects [1], require an impractical number of demonstrations [4], [5], or become inefficient when learning from unsegmented demonstrations [2]. In addition, imitation learning can be easily susceptible to external perturbations [3] (e.g., the robot's proprioceptive states dramatically change as a consequence of sudden interventions from human users) and visual occlusion, which further impedes its deployment in real robotics tasks.

One promising avenue to achieve perturbation resilience is to encode skills via an autonomous dynamical system (DS) [6], [7]. Unlike end-to-end mapping using deep neural networks, DS can handle out-of-distribution states and resist disturbances, where the convergence property of DS is theoretically guaranteed. However, DS usually requires predefined task parameters (e.g., the 3D location of an object in a reaching task), limiting its scalability to high-dimensional visuomotor tasks where only visual observations are available.

In this paper, we aim to endow robots with the capability of rapidly acquiring new long-horizon visuomotor tasks while ensuring robustness throughout task execution. To do so, we propose a novel approach called **M**eta-**I**mitation **L**earning with **A**daptive dynamical primitives (MiLa), which leverages meta-imitation learning for one-shot learning and dynamical movement primitives (DMP) for robustness and smoothness in task execution. Specifically, MiLa enables the learning of unsegmented long-horizon demonstrations, without additional semantic parsing or phase prediction for demonstration segmentation [2].

MiLa is predicated on the assumption that long-horizon tasks are composed of elementary subtasks. We build a skill repertoire consisting of movement primitives and encode them using a well-known dynamical system approach DMP, where only a single demonstration is needed for each primitive. On top of that, we learn a high-level policy via meta-imitation learning to predict the corresponding task parameters from visual observations for each subtask. Within each subtask, the predicted task parameters are fixed and the entire sequence of robot actions is generated using the appropriate DMP under the predicted task parameters. Since each motion primitive represents a type of motion pattern (e.g., reaching, placing, or pushing), robot actions throughout the same subtask are expected to be more consistent and legible, as opposed to the continuous prediction of robot actions as per the current visual observations [2], [3].

The contribution of this paper is a robust meta-imitation learning framework that is capable of

(*i*) learning from unsegmented long-horizon visuomotor demonstrations,
(*ii*) adapting to new tasks with only one-shot demonstration,
(*iii*) resisting external perturbations on robots and visual observations.

II. Related Work

A. Imitation Learning of Movement Trajectories

Many imitation learning algorithms with a focus on motion planning have been proposed and various successful applications have been reported [6], [8]–[11]. For instance, DMP learns the motion pattern of a single demonstration using a spring-damper system, wherein the equilibrium of the system corresponds to the desired target of the robot's motion. Probabilistic movement primitives (ProMP) [8] and kernelized movement primitives (KMP) [9] respectively employ basis and kernel functions to capture the probabilistic characteristics of multiple demonstrations. These approaches exhibit high sample efficiency, allowing for skill learning from just one or a few demonstrations. However, this type

All authors are with the School of Computer Science, University of Leeds, Leeds LS29JT, UK (scswu@leeds.ac.uk; ml20y3w@leeds.ac.uk; y.l.huang@leeds.ac.uk).
* Corresponding author

This work was supported by the Royal Society under grant number RG/R1/251228.

of approach targets the learning of demonstrations associated with time input or multiple-dimensional inputs and becomes inappropriate when dealing with high-dimensional visual inputs (i.e., images).

To connect imitation learning for motion planning with image inputs, there are some works on learning a mapping from images to DMP parameters (e.g., basis function weights, motion target and duration) [12], [13]. Once these parameters are obtained, smooth trajectories can be naturally generated via DMP. However, these approaches are restricted to the learning of single-skill tasks.

B. Imitation Learning of Long-Horizon Tasks

Many recent works addressed long-horizon visuomotor tasks via imitation learning. In [14]–[17], hierarchical imitation learning was studied, where a high-level policy was used to make 'plans' for compound tasks and a low-level controller was designed to execute these 'plans' (e.g., latent variables or sub-goals). In [18], [19], one-shot imitation learning was investigated, aiming to learn a new task from a single demonstration in the form of a complete or partial trajectory and video. Notably, meta-imitation learning has earned a relevant place due to its reliable performance [20]–[22]. The objective of meta-imitation learning is to exploit shared structures among the tasks sampled from the same distribution and search for an optimal policy capable of adapting quickly to new tasks. Most works in this line only consider learning a single skill, without tapping the learning of long-horizon skills.

A work on meta-learning that is closely related to ours is [2], which tackles long-horizon tasks by segmenting them into subtasks and subsequently performing meta-imitation learning on subtasks. However, this approach requires primitive-level demonstrations to train an additional motion phase predictor. The predictor is used for segmenting demonstrations into subtasks, acting as an indispensable step during meta-training. In fact, collecting unsegmented long-horizon demonstrations is more straightforward and raw demonstrations are more easily accessible. In this paper, we propose to meta-train a policy on unsegmented visuomotor demonstrations, without any specific treatment on demonstration parsing or segmentation. Besides, we leverage the dynamical feature of DMP to ensure that our framework is robust to external perturbations (e.g., from visual inputs and the robot's proprioceptive states), leading to another advantage against existing imitation learning methods [1], [3], [20]–[22].

III. Preliminaries

A. Dynamical Movement Primitives

Suppose we have access to a demonstration of time-length N, i.e., $\{t_n, \boldsymbol{\xi}_n, \dot{\boldsymbol{\xi}}_n, \ddot{\boldsymbol{\xi}}_n\}_{n=1}^{N}$. Here, $\boldsymbol{\xi}_n \in \mathbb{R}^{\mathcal{O}}$ represents $\mathcal{O}$-dimensional position (or joint angles) at the n–th time step, while $\dot{\boldsymbol{\xi}}_n$ and $\ddot{\boldsymbol{\xi}}_n$ respectively denote the corresponding velocity and acceleration. DMP encodes the demonstration using a second-order dynamical model:

$$\tau \dot{s} = -\alpha s, \tag{1}$$

$$\tau^2 \ddot{\boldsymbol{\xi}} = \mathbf{K}_p(\mathbf{g} - \boldsymbol{\xi}) - \tau \mathbf{K}_v \dot{\boldsymbol{\xi}} + s(\mathbf{g} - \boldsymbol{\xi}_0) \odot \boldsymbol{f}_{\mathbf{w}}(s), \tag{2}$$

$$\boldsymbol{f}_w(s) = \mathbf{W} \left[\frac{\varphi_1(s)}{\sum_{h=1}^{H} \varphi_h(s)} \ \frac{\varphi_2(s)}{\sum_{h=1}^{H} \varphi_h(s)} \ \cdots \ \frac{\varphi_H(s)}{\sum_{h=1}^{H} \varphi_h(s)} \right]^{\top}. \tag{3}$$

Equation (1) is utilized to convert time into the phase variable s, thereby eliminating explicit time dependence. Here, τ denotes the motion duration, and α signifies the decay factor. In (2), $\mathbf{K}_p$ and $\mathbf{K}_v$ denote the user-specified stiffness and damping matrices, respectively. $\mathbf{g}$ and $\boldsymbol{\xi}_0$ represent the goal (end-point) and start-point of a trajectory. The symbol $\odot$ stands for the element-wise product. $\boldsymbol{f}_w(s)$ represents the forcing term, typically expressed as a linear combination of pre-defined Gaussian basis functions (see (3)). $\mathbf{W} \in \mathbb{R}^{\mathcal{O} \times H}$ means learnable parameters corresponding to the motion pattern of the demonstration.

B. Model-Agnostic Meta-learning

Model-agnostic meta-learning (MAML) is a meta-learning algorithm proposed to rapidly learn new tasks using a small number of data [23]. It operates under the assumption that a shared structure exists among meta-training and meta-test tasks (i.e., all tasks are drawn from the same task distribution). Consider imitation learning using MAML with a policy π_θ parameterized by $\boldsymbol{\theta}$ and gauged by behaviour cloning (BC) loss function $\mathcal{L}_{\text{BC}}$. During meta-training, MAML randomly selects a meta-training task $\mathcal{T}$ from the task distribution $p(\mathcal{T})$ and partitions demonstrations from the task $\mathcal{T}$ into training dataset $\mathcal{D}_{\mathcal{T}}^{\text{tr}}$ and validation dataset $\mathcal{D}_{\mathcal{T}}^{\text{val}}$. MAML then optimizes the policy parameters $\boldsymbol{\theta}$ such that one (or a few) gradient update on $\mathcal{D}_{\mathcal{T}}^{\text{tr}}$ leads to favourable performance on $\mathcal{D}_{\mathcal{T}}^{\text{val}}$. The objective function of MAML is formulated as

$$\begin{aligned} &\min_{\boldsymbol{\theta}} \sum_{\mathcal{T} \sim p(\mathcal{T})} \mathcal{L}_{\text{BC}}(\boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{BC}}(\boldsymbol{\theta}, \mathcal{D}_{\mathcal{T}}^{\text{tr}}), \mathcal{D}_{\mathcal{T}}^{\text{val}}) \\ &= \min_{\boldsymbol{\theta}} \sum_{\mathcal{T} \sim p(\mathcal{T})} \mathcal{L}_{\text{BC}}(\boldsymbol{\phi}_{\mathcal{T}}, \mathcal{D}_{\mathcal{T}}^{\text{val}}), \end{aligned} \tag{4}$$

where $\alpha > 0$ refers to the step size of gradient descent, and $\boldsymbol{\phi}_{\mathcal{T}}$ corresponds to the updated parameters after learning on $\mathcal{D}_{\mathcal{T}}^{\text{tr}}$. During the meta-testing (i.e., model inference) phase, the policy adapts to a new task $\mathcal{D}_{\mathcal{T}_{\text{test}}}$ by utilizing the updated parameters $\boldsymbol{\phi}_{\mathcal{T}_{\text{test}}} = \boldsymbol{\theta} - \alpha \nabla_\theta \mathcal{L}_{\text{BC}}(\boldsymbol{\theta}, \mathcal{D}_{\mathcal{T}_{\text{test}}})$.

IV. Meta-Imitation Learning with Adaptive Dynamical Primitives

We assume that we have access to a dataset $\{\mathbf{d}_h\}_{h=1}^{H}$ comprising H demonstrations across K tasks $\{\mathcal{T}_k\}_{k=1}^{K}$, with $\mathbf{d}_h$ represents the h-th unsegmented, long-horizon visuomotor demonstration. Each demonstration $\mathbf{d}_h = \{\mathbf{v}_h, \boldsymbol{\kappa}_h\}$ consists of a sequence of visual images $\mathbf{v}_h = \{\mathbf{o}_t\}_{t=1}^{T(h)}$, as well as the corresponding robot trajectory $\boldsymbol{\kappa}_h = \{\boldsymbol{\xi}_t\}_{t=1}^{T(h)}$. Here, $T(h)$ denotes the time length, $\mathbf{o}_t$ and $\boldsymbol{\xi}_t$ represent image observation and robot action (e.g., end-effector position or joint angles), respectively. We assume all demonstrations can be decomposed into C different subtasks and the order of the subtasks is known. Formally, we write the robot trajectory

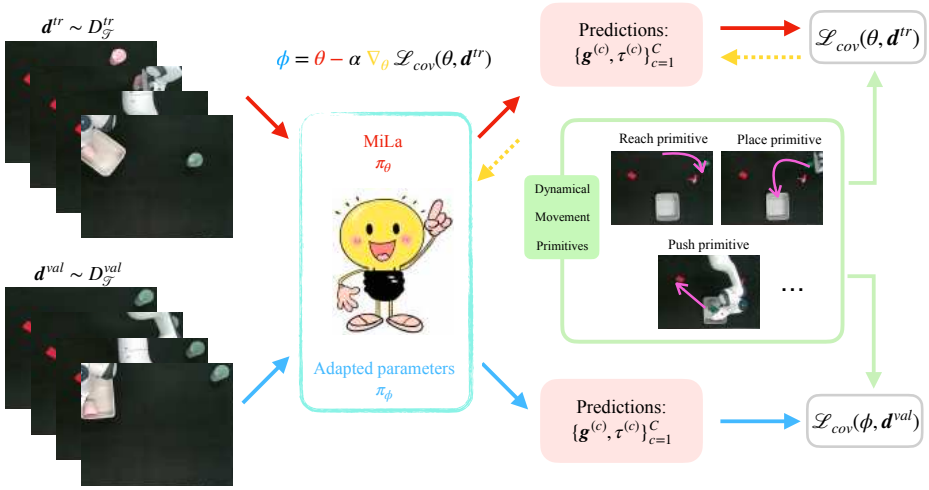

Fig. 1: An overview of the MiLa framework. After learning from $\mathbf{d}^{\text{tr}}$ (indicated by yellow arrows), MiLa acquires the capability to adapt to new tasks $\mathbf{d}^{\text{val}}$ (see blue arrows). Instead of predicting robot actions as per visual inputs at each time step, MiLa predicts task parameters for each subtask and a set of dynamical movement primitives are employed to generate robot trajectories across different subtasks.

as $\boldsymbol{\kappa} = \boldsymbol{\kappa}^1 \oplus \boldsymbol{\kappa}^2 \cdots \oplus \boldsymbol{\kappa}^C$, where $\boldsymbol{\kappa}^c$ represent the trajectory in subtask c and $\oplus$ denotes the concatenation of trajectories.

To learn from unsegmented demonstration, we first establish a repertoire of motion primitives and use DMP to encode each primitive. Given the elementary motion primitives, we propose to learn a high-level policy, referred to as MiLa, for predicting task parameters for motion primitives and composing primitives.

A. Skill Repertoire

The objective of constructing a skill repertoire is to provide a set of reliable primitives for a high-level policy. By reusing and composing motion primitives, complex and long-horizon tasks can be accomplished. For each type of skill (e.g., reaching, placing, and pushing), we collect one demonstration of robot trajectory via kinesthetic teaching. By feeding the demonstration into the DMP model in (3), the motion pattern that underlies the demonstration can be determined. Repeating the same procedure for all primitives, we can obtain a skill repertoire comprising different DMPs, denoted by $\{\rho^c\}_{c=1}^{C}$.

Given a DMP ρ^c, we can use it to generate an adapted trajectory for a new task, where we only need to specify starting point $\boldsymbol{\xi}_0$, desired target $\mathbf{g}$, and motion duration τ. The adapted trajectory is given by

$$\hat{\boldsymbol{\kappa}}^c(t) = \rho^c(\boldsymbol{\xi}_0, \mathbf{g}, \tau, t), \tag{5}$$

where $\hat{\boldsymbol{\kappa}}^c(t) \in \mathbb{R}^{\mathcal{O}}$ denotes the planned robot actions at time t via ρ^c. Therefore, DMP builds a connection between task parameters (i.e., $\{\mathbf{g}, \tau\}$) and robot trajectories. Specifically, the adapted trajectory via DMP maintains the motion style extracted from the demonstration, which is expected to be legible and predictable by human users.

B. Meta-Imitation Learning with Adaptive Dynamical Primitives

Instead of predicting robot actions according to visual inputs at each time step, we propose to predict task parameters for each subtask instead. Formally, given an initial image observation $\mathbf{o}_1$ and the goal image $\mathbf{o}_T$ (i.e., the last observation) of a demonstration, we propose to learn the policy $\pi_\theta(\mathbf{g}^{(1)}, \tau^{(1)} | \mathbf{o}_1, \mathbf{o}_T)$ that predicts the end-point and motion duration of the first subtask. These predicted task parameters, together with the robot's initial state $\boldsymbol{\xi}_0^{(1)}$, will be passed onto the motion primitive ρ^1, yielding a robotic trajectory $\hat{\boldsymbol{\kappa}}^1(t) = \rho^1(\boldsymbol{\xi}_0^{(1)}, \mathbf{g}^{(1)}, \tau^{(1)}, t)$. Similarly, new task parameters $(\mathbf{g}^{(2)}, \tau^{(2)})$ for the second subtask are predicted using observations at $\mathbf{o}_{1+\tau^{(1)}/\delta}$ together with the goal image $\mathbf{o}_T$, and a new robot trajectory is planned as $\hat{\boldsymbol{\kappa}}^2(t) = \rho^2(\boldsymbol{\xi}_0^{(2)}, \mathbf{g}^{(2)}, \tau^{(2)}, t)$. Here, δ denotes the time interval between two consecutive robot actions. $\boldsymbol{\xi}_0^{(2)}$ is the end-point of the trajectory $\hat{\boldsymbol{\kappa}}^1(t)$. By repeating the prediction procedure, we can predict C groups of task parameters and generate C trajectories for the robot.

Before we measure the difference between the predicted robot trajectory $\{\hat{\boldsymbol{\kappa}}^1, \hat{\boldsymbol{\kappa}}^2, \ldots, \hat{\boldsymbol{\kappa}}^C\}$ and the demonstrated robot trajectory $\boldsymbol{\kappa}_h = \{\boldsymbol{\xi}_t\}_{t=1}^{T(h)}$, we exploit the covariance weighted loss function. As pointed out in numerous works on probabilistic imitation learning [8], [24], [25], different demonstrations could be collected even for the same task and such variability is naturally reflected by the variance of demonstrations. In addition to the demonstrations collected for training DMPs in Section IV-A, we collect additional (approximately 5–6) demonstrations to model the intrinsic variability for each type of skill. We use Gaussian mixture model to learn the joint distribution $\mathcal{P}(t, \boldsymbol{\xi})$ and adopt Gaussian mixture regression [26] to compute the covariance function $\boldsymbol{\Sigma}^c(t) = \mathbb{D}(\boldsymbol{\xi}(t)|t)$. Now, we formulate the covariance weighted loss function between the predicted and demonstrated long-horizon robot trajectories as

$$\begin{aligned}\mathcal{L}_{cov}(\boldsymbol{\theta}, \mathbf{d}_h) = \frac{1}{T(h)} \sum_{c=1}^{C} \sum_{t=1}^{\tau^{(c)}} \gamma_c \big(\boldsymbol{\kappa}_h(t_c + t) - \hat{\boldsymbol{\kappa}}^c(t)\big)^\top \\ \big(\boldsymbol{\Sigma}^c(t)\big)^{-1} \big(\boldsymbol{\kappa}_h(t_c + t) - \hat{\boldsymbol{\kappa}}^c(t)\big),\end{aligned} \tag{6}$$

where $t_c = \sum_{i=1}^{c-1} \tau^{(i)}$ ($t_c = 0$ if $c = 1$), γ_c represents an

Algorithm 1 Meta-Imitation Learning with Adaptive Dynamical Primitives

Require: α: step size for gradient descent
Require: skill repertoire $\boldsymbol{\rho}_c$ and covariance function $\boldsymbol{\Sigma}^c(t)$, $c = 1, 2, \ldots, C$
1: randomly initialize $\boldsymbol{\theta}$
2: **while** meta-training **do**
3: Sample a task $\mathcal{T} \sim p(\mathcal{T})$
4: Random partition $\mathcal{D}_\mathcal{T}$ as $(\mathcal{D}_\mathcal{T}^{\text{tr}}, \mathcal{D}_\mathcal{T}^{\text{val}})$
5: Sample one demonstration $\mathbf{d}^{\text{tr}} \sim \mathcal{D}_\mathcal{T}^{\text{tr}}$
6: Predict C groups of task parameters along $\mathbf{d}^{\text{tr}}$
7: Generate robot trajectories for all subtasks with task parameters from line 6
8: Compute adapted parameters $\boldsymbol{\phi}_\mathcal{T} = \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} \mathcal{L}_{cov}(\boldsymbol{\theta}, \mathbf{d}^{\text{tr}})$ via (7)
9: Sample one demonstration $\mathbf{d}^{\text{val}} \sim \mathcal{D}_\mathcal{T}^{\text{val}}$
10: Predict C groups of task parameters along $\mathbf{d}^{\text{val}}$
11: Generate robot trajectories for all subtasks with task parameters from line 10
12: Update parameters $\boldsymbol{\theta}$ via minimizing (7)
13: **end while**
14: **return** θ

adjustable weight parameter for each subtask. The entire loss function for meta-training becomes

$$J(\boldsymbol{\theta}) = \sum_{\mathcal{T} \sim p(\mathcal{T})} \sum_{(\mathbf{d}^{\text{tr}}, \mathbf{d}^{\text{val}}) \in \mathcal{D}_\mathcal{T}} \mathcal{L}_{cov}(\boldsymbol{\phi}_\mathcal{T}, \mathbf{d}^{\text{val}}) \quad (7)$$
$$\text{with} \quad \boldsymbol{\phi}_\mathcal{T} = \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} \mathcal{L}_{cov}(\boldsymbol{\theta}, \mathbf{d}^{\text{tr}}).$$

With the loss function in (6), we can compute the training error without segmenting demonstrations beforehand, which largely facilitates meta-training on unsegmented demonstrations. Note that $\mathcal{L}_{cov}$ achieves its minimum value only when the predicted subtasks (i.e., task parameters) are accurate. Therefore, minimizing the loss function in (6) drives the policy towards learning all subtasks in each long-horizon task precisely. An overview of the proposed approach MiLa is depicted in Fig. 1. To comply with meta-testing where the goal image is not provided after the one-shot demonstration, we sample a random image as the goal image $\mathbf{o}_T^{val}$ and replace the embedding of the goal image with a vector $\mathbf{z}_0$ during the meta-training phase. The entire procedure of MiLa is summarized in Algorithm 1.

MiLa predicts task parameters by 'casting a glance' at the environment, it makes predictions solely from image observations at the beginning of each subtask, enabling it to resist visual occlusion over the course of the execution of the subtask. Furthermore, MiLa ensures that the robot convergences to the predicted target point owing to the dynamical feature offered by DMP [6], even when the robot encounters dramatic perturbations. Last, MiLa provides smooth trajectories for robots since the action trajectory for each subtask is planned as a whole by DMP.

V. Experiments

In this section, we aim to answer the following questions:

Fig. 2: The experimental setup as well as objects for training and testing in long-horizon tasks.

(1) Is MiLa competitive with state-of-the-art baselines in long-horizon tasks, including MAML and goal-conditioned BC?
(2) Can MiLa effectively resist external disturbances and visual occlusion during real-world task execution?

We consider the long-horizon visuomotor task that requires the robotic arm to reach and grasp a target object, place it into a basket, and finally push the basket to a desired location (i.e., the red squared marker). We collected 1,260 demonstrations as the training set using a 7-DoF Franka Emika Panda robotic arm and a top-view Intel RealSense D455 camera. The experimental setup is shown in Fig. 2 (left plot). The middle plot of Fig. 2 depicts the objects used for demonstration collection. All demonstrations, including videos and corresponding trajectories, were collected at the frequency of 30 Hz, where each demonstration lasts approximately 20 seconds. The demonstrations used to establish the skill repertoire and estimate the covariance function are depicted in Fig. 3.

We compare MiLa's performance against state-of-the-art approaches:

- **Model-Agnostic Meta-Learning (MAML)**: a meta-imitation learning policy follows the implementation of [2], [20], [22], [23], which requires the segmentation of long-horizon tasks into subtasks and then computes adapted parameters on the subtasks.
- **Goal-conditioned Behaviour Cloning (GCBC)**: a goal-image conditioned policy that takes as input a real-time image and the robot's current state, along with a final image of the subtask [27].

As an ablation study, we evaluate MiLa without using the covariance weighted loss function, i.e.,

- **MiLa-NoWeight**: setting $\boldsymbol{\Sigma}^c(t)$ as an identity matrix in (6).

To ensure a fair comparison, all methods use a similar network architecture and each method is evaluated with its optimal hyperparameters. As MAML requires the segmentation of long-horizon tasks into distinct subtasks, we train an individual model for each subtask. Similarly, GCBC also involves training an individual model for each subtask. In contrast, for MiLa and MiLa-NoWeight we train a single unified policy to directly learn the unsegmented, long-horizon demonstrations.

To assess the generalization capability in new settings (i.e., held-out objects, see the testing objects in Fig. 2), we

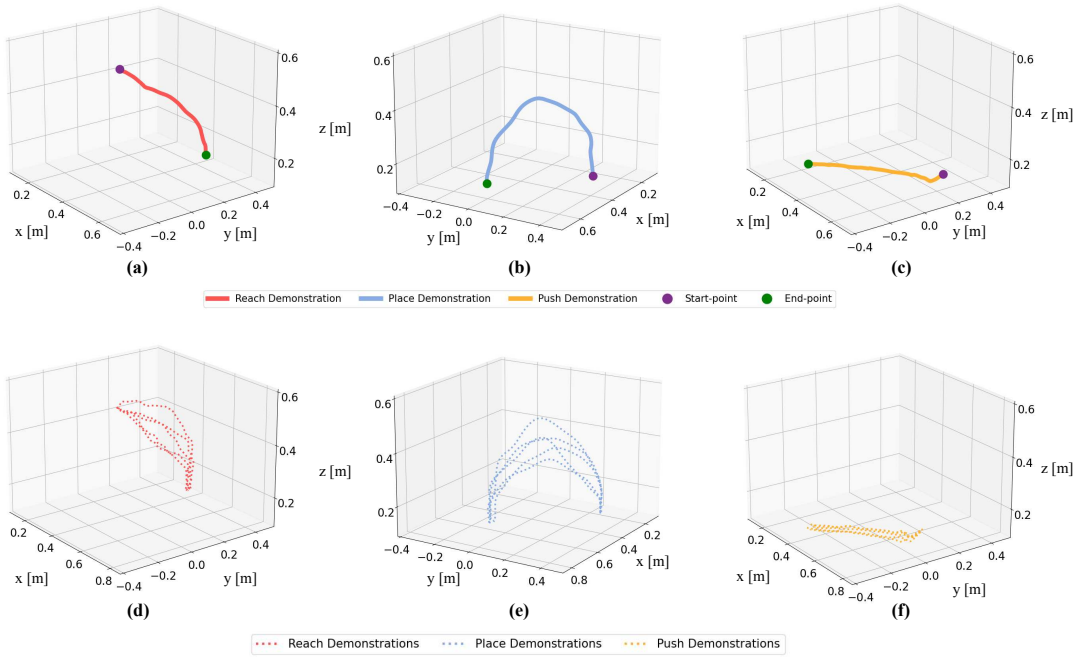

Fig. 3: Demonstrations utilized to establish the skill repertoire $\{\boldsymbol{\rho}^c\}_{c=1}^3$. Dynamical movement primitives for the reaching, placing, and pushing skills are learned from demonstrations depicted in (*a*)–(*c*), respectively. Additionally, we collect 5 demonstrations for each skill to model the intrinsic skill variability, as shown in (*d*)–(*f*).

TABLE I: Success rates of different methods.

Method	Success (inappropriate placing)	Success (proper placing)	Overall Success Rate
GCBC	25%	20%	45%
MAML	40%	5%	45%
MiLa-NoWeight	15%	55%	70%
MiLa	**0%**	**70%**	**70%**

provide one-shot demonstration for MiLa, MiLa-NoWeight, and MAML to adapt their policy parameters. For GCBC, we collect goal images for each subtask separately per evaluation.

A. Evaluations on Reaching-Placing-Pushing Tasks

An illustration of the long-horizon task is provided in the first row of Fig 4. We carry out 5 groups of evaluations on the held-out objects (see the right plot in Fig. 2) and each group includes 4 trails by altering the locations of the target object and the basket. In total, we have 20 trials to evaluate each method.

The results in the second column of Table I indicate that GCBC and MAML frequently place the grasped objects in the wrong places although the entire task is completed successfully due to the move and rotation of the basket, see examples in the fourth and fifth rows in Fig. 4, where the grasped object collides with the basket. A smaller number of inappropriate placements are also observed in MiLa-NoWeight, as displayed in the third row of Fig. 4. In comparison, MiLa consistently places objects into the basket appropriately, see an example in the second row of Fig. 4. This finding highlights the importance of using the covariance weighted loss function to mitigate the effect of large skill variability in the placing task.

The fourth column of Table I (i.e., the sum of the second and third columns) also shows that MiLa and MiLa-NoWeight achieve higher success rates than MAML and GCBC. We suggest that the low success rate in GCBC may be attributed to visual occlusion. The robot could obscure the object when approaching it since the camera is mounted over the object (see the left plot in Fig. 2). In comparison, MiLa requires only a single image to perform each subtask, effectively mitigating issues related to visual occlusion. Finally, we find that many failure cases across these four approaches are attributable to the reaching subtask. Although the robot's gripper either touches or nearly touches parts of the objects, it still fails to grasp them. This issue has also been observed in [2]. Considering that MiLa-NoWeight is an ablation study of MiLa and its performance is inferior to MiLa, we only use MiLa for comparison with MAML and GCBC in the following evaluations.

B. Evaluations in the Presence of External Pertubations

Now, we consider exerting external perturbations on the robot arm directly. Specifically, the perturbations are imposed by dragging the robot's joints or end-effector arbitrarily.

In Fig. 5, the yellow arrows depict the direction of perturbations. We can see that MiLa is the only method capable of recovering from the disturbances, whereas MAML and GCBC fail to do so, see the second and third rows of Fig. 5. Note that MAML and GCBC may exhibit abnormal behaviours under perturbations. For example, at the last column of the second and third rows, MAML becomes immobilized and GCBC moves to an area that is far away from the demonstrated robot workspace.

The ability to resist disturbances exhibited by MiLa is attributed to the use of DMP to ensure goal-oriented motion,

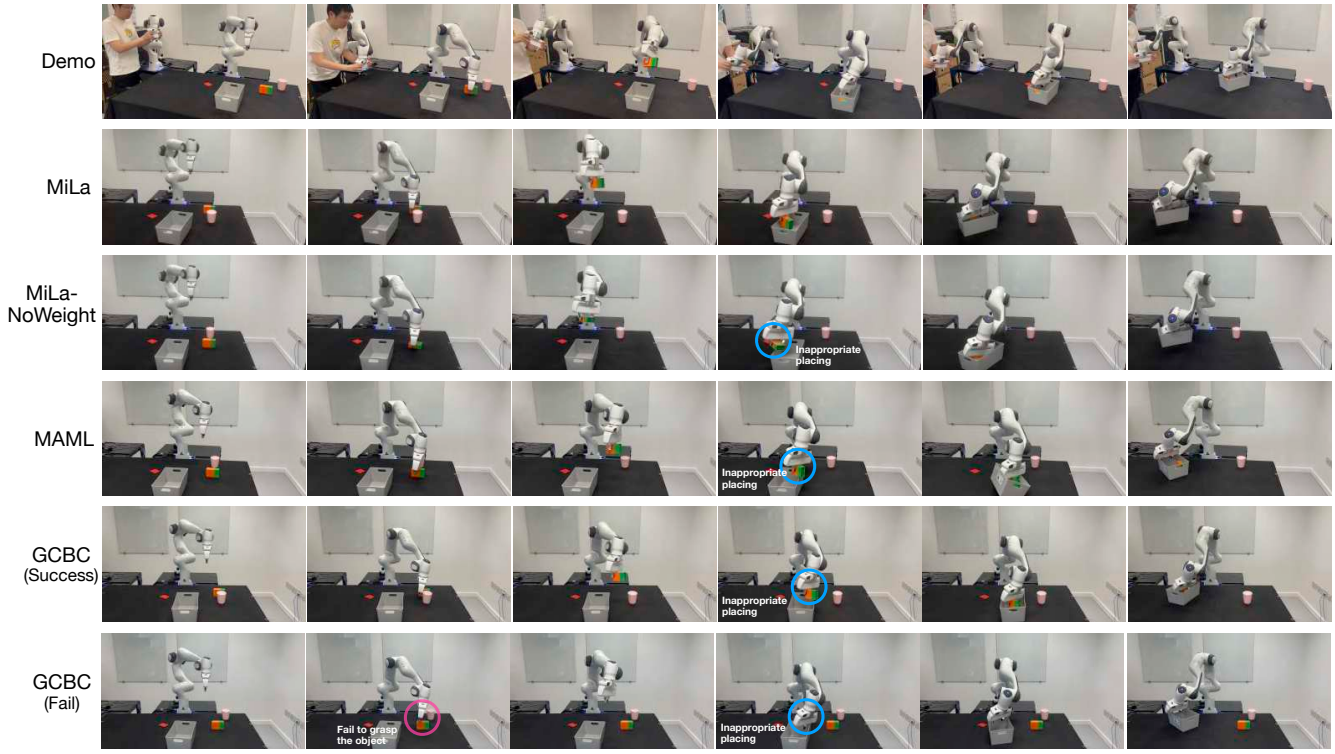

Fig. 4: Snapshot of the long-horizon task evaluations. *First row* shows the kinesthetic teaching of the reaching-placing-pushing task. *Second and third rows* correspond to the evaluations of MiLa and MiLa-NoWeight, respectively. *Fourth row* illustrates an evaluation of MAML. *Fifth and sixth rows* present the success and fail cases using GCBC, respectively.

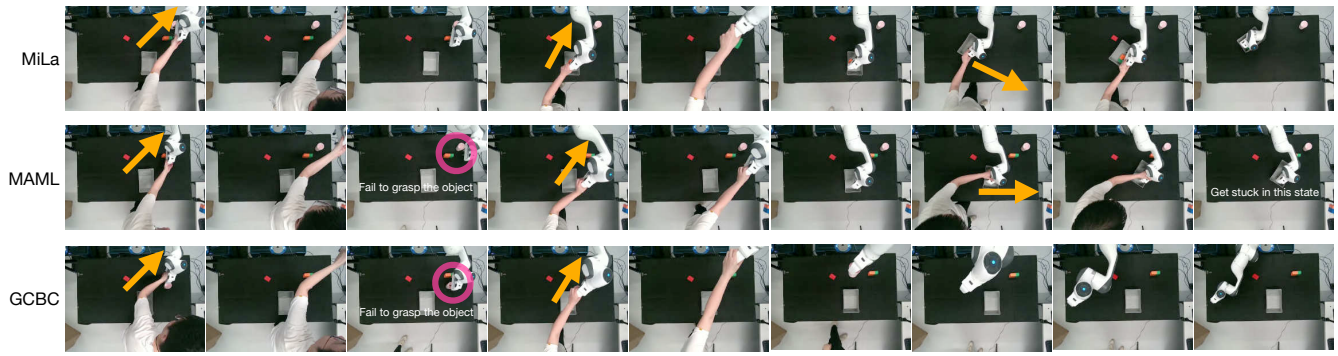

Fig. 5: Snapshot of the long-horizon task in the presence of external disturbances and visual occlusion (i.e., human arm). *First, second, and third row* correspond to evaluations using MiLa, MAML and GCBC, respectively.

where the goal, inferred from visual observation at the start of each subtask, remains fixed throughout the subtask execution. In contrast, MAML and GCBC predict actions based on image observations and proprioceptive states at each time step. Any perturbations during task execution may disrupt these predictions, potentially resulting in task failure. We emphasize that external perturbations applied to the robotic arm also introduce visual disturbances, as the user's arm is unseen in the training dataset, evidencing that MiLa can resist disturbances in terms of visual observations and robot states.

C. Evaluations in the Presence of Visual Occlusion

In addition to external pertubations, we evaluate the performance of different methods by considering visual occlusion during task execution. In our evaluations, visual occlusion is caused by the user's hand moving in front of the camera. The extreme case corresponds to the full occlusion of the camera's view.

The snapshots illustrating the evaluation of MiLa under visual occlusion are presented in the first and second rows of Fig. 6. The first row captures the robot's perspective, while the second row provides the human user's view. Despite significant occlusion by the user's hand, MiLa successfully executes the long-horizon task. This success is achieved because MiLa processes only the first frame at the beginning of each subtask, thereby enabling it to disregard any subsequent frames affected by occlusion. In contrast, MAML and GCBC predict actions in a per-timestep or per-frame fashion and visual occlusion can adversely affect their predictions due to the introduction of out-of-distribution images, as illustrated in the third to sixth rows in Fig. 6, where both

Fig. 6: Snapshot of the long-horizon task in the presence of visual occlusion. *First and second rows* correspond to evaluations using MiLa, with the depth camera's perspective and the user's view, respectively. Similarly, *third and fourth rows* correspond to evaluations using MAML. *Fifth and sixth rows* are evaluations with GCBC.

MAML and GCBC are unable to execute the placing task properly. Additionally, since MiLa operates at the trajectory level rather than on a per-timestep basis, this treatment provides smoother trajectories for the robot, compared to those generetated by MAML and GCBC.

VI. Conclusion

In this paper, we introduced a novel meta-imitation learning approach, MiLa, which is capable of rapidly learning new long-horizon visuomotor tasks and effectively resisting perturbations during task execution. MiLa establishes a skill repertoire capturing various elementary motion primitives and subsequently solving long-horizon tasks by reusing and composing motion primitives. Experimental results indicate that MiLa achieves superior performance compared to state-of-the-art baselines. Furthermore, MiLa enables learning from unsegmented demonstrations and demonstrates robust resistance to perturbations from both the robot's proprioceptive states and visual inputs.

While we assume that the order of subtasks is fixed, an important extension is to simultaneously learn both the order of primitives and their corresponding task parameters. As a long-term goal, it would be promising to extend MiLa to learn long-horizon tasks from cross-domain demonstrations, including those from humans and robots with different embodiments.

References

[1] M. Xu, Z. Xu, C. Chi, M. Veloso, and S. Song, "Xskill: Cross embodiment skill discovery," in *Conference on Robot Learning*. PMLR, 2023, pp. 3536–3555.

[2] T. Yu, P. Abbeel, S. Levine, and C. Finn, "One-shot hierarchical imitation learning of compound visuomotor tasks," *arXiv preprint arXiv:1810.11043*, 2018.

[3] C. Wang, L. Fan, J. Sun, R. Zhang, L. Fei-Fei, D. Xu, Y. Zhu, and A. Anandkumar, "Mimicplay: Long-horizon imitation learning by watching human play," *arXiv preprint arXiv:2302.12422*, 2023.

[4] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu *et al.*, "Rt-1: Robotics transformer for real-world control at scale," *arXiv preprint arXiv:2212.06817*, 2022.

[5] P. Sundaresan, Q. Vuong, J. Gu, P. Xu, T. Xiao, S. Kirmani, T. Yu, M. Stark, A. Jain, K. Hausman *et al.*, "Rt-sketch: Goal-conditioned imitation learning from hand-drawn sketches," *arXiv preprint arXiv:2403.02709*, 2024.

[6] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: learning attractor models for motor behaviors," *Neural computation*, vol. 25, no. 2, pp. 328–373, 2013.

[7] S. M. Khansari-Zadeh and A. Billard, "Learning stable nonlinear dynamical systems with gaussian mixture models," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 943–957, 2011.

[8] A. Paraschos, C. Daniel, J. R. Peters, and G. Neumann, "Probabilistic movement primitives," *Advances in neural information processing systems*, vol. 26, 2013.

[9] Y. Huang, L. Rozo, J. Silvério, and D. G. Caldwell, "Kernelized movement primitives," *IJRR*, vol. 38, no. 7, pp. 833–852, 2019.

[10] J. Silvério, Y. Huang, L. Rozo *et al.*, "An uncertainty-aware minimal intervention control strategy learned from demonstrations," in *2018 IROS*. IEEE, 2018, pp. 6065–6071.

[11] Y. Huang, J. Silvério, L. Rozo, and D. G. Caldwell, "Hybrid probabilistic trajectory optimization using null-space exploration," in *2018 ICRA*. IEEE, 2018, pp. 7226–7232.

[12] S. Bahl, M. Mukadam, A. Gupta, and D. Pathak, "Neural dynamic policies for end-to-end sensorimotor learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 5058–5069, 2020.

[13] A. Gams, A. Ude, J. Morimoto *et al.*, "Deep encoder-decoder networks for mapping raw images to dynamic movement primitives," in *2018 ICRA*. IEEE, 2018, pp. 5863–5868.

[14] A. Mandlekar, D. Xu, R. Martín-Martín, S. Savarese, and L. Fei-Fei, "Learning to generalize across long-horizon tasks from human demonstrations," *arXiv preprint arXiv:2003.06085*, 2020.

[15] T. Shankar, S. Tulsiani, L. Pinto, and A. Gupta, "Discovering motor programs by recomposing demonstrations," in *International Conference on Learning Representations*, 2019.

[16] S. Nasiriany, T. Gao, A. Mandlekar, and Y. Zhu, "Learning and retrieval from prior data for skill-based imitation learning," *arXiv preprint arXiv:2210.11435*, 2022.

[17] P. Sharma, D. Pathak, and A. Gupta, "Third-person visual imitation learning via decoupled hierarchical controller," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[18] S. Shin, D. Lee, M. Yoo, W. K. Kim, and H. Woo, "One-shot imitation in a non-stationary environment via multi-modal skill," in *ICML*. PMLR, 2023, pp. 31 562–31 578.

[19] Y. Duan, M. Andrychowicz, B. Stadie, O. Jonathan Ho, J. Schneider, I. Sutskever, P. Abbeel, and W. Zaremba, "One-shot imitation learning," *Advances in neural information processing systems*, vol. 30, 2017.

[20] C. Finn, T. Yu, T. Zhang, P. Abbeel, and S. Levine, "One-shot visual imitation learning via meta-learning," in *Conference on robot learning*. PMLR, 2017, pp. 357–368.

[21] J. Li, T. Lu, X. Cao, Y. Cai, and S. Wang, "Meta-imitation learning by watching video demonstrations," in *ICLR*, 2021.

[22] T. Yu, C. Finn, A. Xie, S. Dasari, T. Zhang, P. Abbeel, and S. Levine, "One-shot imitation from observing humans via domain-adaptive meta-learning," *arXiv preprint arXiv:1802.01557*, 2018.

[23] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International conference on machine learning*. PMLR, 2017, pp. 1126–1135.

[24] Y. Huang, J. Silvério, L. Rozo, and D. G. Caldwell, "Generalized task-parameterized skill learning," in *2018 ICRA*. IEEE, 2018, pp. 5667–5474.

[25] S. Calinon, "A tutorial on task-parameterized movement learning and retrieval," *Intelligent service robotics*, vol. 9, pp. 1–29, 2016.

[26] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *Journal of artificial intelligence research*, vol. 4, pp. 129–145, 1996.

[27] C. Lynch, M. Khansari, T. Xiao, V. Kumar, J. Tompson, S. Levine, and P. Sermanet, "Learning latent plans from play," in *Conference on robot learning*. PMLR, 2020, pp. 1113–1132.