ISSN: 2248-9622, Vol. 15, Issue 9, September 2025, pp 01-09

RESEARCH ARTICLE OPEN ACCESS

Accelerated Robot Skill Acquisition Through Cross-Task Transfer Learning

Akshar Yadav
Amity International School, Sector 46
Gurugram, Haryana, India

ABSTRACT—
Training robots to acquire new skills with minimal data remains a fundamental challenge in robotics research. This paper introduces a novel transfer learning framework specifically designed for robotic skill acquisition that significantly reduces training time and sample complexity. Our approach, Multi-Domain Skill Transfer (MDST), leverages knowledge from previously learned tasks to accelerate learning in new domains while addressing the common challenges of negative transfer and catastrophic forgetting. We evaluate MDST on a diverse set of manipulation tasks using both simulated and physical robot platforms. Results demonstrate that our method reduces the required training samples by 78% compared to learning from scratch while achieving comparable or superior performance. Furthermore, we show that MDST maintains performance on source tasks and effectively transfers knowledge even between seemingly dissimilar tasks. The proposed framework represents a significant step toward enabling robots to rapidly acquire new skills in dynamic environments with minimal human intervention.
Index Terms—transfer learning, robotics, skill acquisition, deep reinforcement learning, manipulation

Date of Submission: 25-08-2025 Date of acceptance: 03-09-2025

I. INTRODUCTION

Industrial and service robots are increasingly expected to perform diverse tasks in complex, unstructured environments. However, programming robots to perform specific skills remains time-consuming and requires significant expertise. Recent advances in machine learning, particularly reinforcement learning (RL), have enabled robots to learn skills autonomously [1]. However, these approaches typically require substantial training data, often millions of interactions with the environment, which is impractical for many real-world applications [2].

Transfer learning offers a promising solution to this challenge by leveraging knowledge from previously learned tasks to accelerate learning in new domains [3]. While transfer learning has shown impressive results in fields such as computer vision and natural language processing [4], its application to robotic skill acquisition presents unique challenges. Robots operate in the physical world, where actions have complex dynamics and consequences, and task structures can vary significantly [5].

Previous attempts to apply transfer learning to robotics have shown promising but limited results, often constrained to closely related tasks or specific robot configurations [6]. Furthermore, many approaches suffer from negative transfer, where knowledge from source tasks hinders rather than helps learning target tasks, and catastrophic forgetting, where learning new tasks degrades performance on previously learned tasks [7].

In this paper, we introduce Multi-Domain Skill Transfer (MDST), a novel framework for accelerating robot skill acquisition through transfer learning.

Our key contributions include:
• A task-agnostic representation learning method that captures generalizable knowledge across different robotic skills
• An adaptive knowledge transfer mechanism that selectively applies relevant knowledge from source tasks to target tasks
• A continual learning component that prevents catastrophic forgetting and enables lifelong skill acquisition
• Comprehensive evaluation on both simulated and physical robot platforms across a diverse set of manipulation tasks

The remainder of this paper is organized as follows: Section II reviews related work, Section III describes our proposed framework, Section IV presents our experimental setup, Section V discusses results and analysis, and Section VI concludes with limitations and future directions.

II. RELATED WORK

A. Robot Skill Learning

Learning approaches for robotic skill acquisition can be broadly categorized into three paradigms: imitation learning, reinforcement learning, and hybrid methods [8]. Imitation learning enables robots to learn from human demonstrations but struggles with generalization to novel scenarios [9]. Reinforcement learning allows robots to discover optimal policies through exploration but typically requires extensive environment interactions [5]. Hybrid approaches combine elements of both paradigms to reduce sample complexity while maintaining generalization capabilities [10]. Recent work has focused on improving data efficiency through model-based reinforcement learning [11], metalearning [12], and curriculum learning [13]. While these approaches have shown promising results, they still require substantial training data for complex tasks, limiting their practical applicability.

B. Transfer Learning

Transfer learning has emerged as a powerful paradigm for leveraging knowledge from source tasks to improve learningefficiency in target tasks [14]. In the context of deep learning, transfer learning typically involves pre-training networks on source tasks and fine-tuning on target tasks [15]. This approach has revolutionized fields such as computer vision [4] and natural language processing [16].

In robotics, transfer learning has been applied to various tasks, including navigation [17], manipulation [6], and locomotion [18]. However, these approaches often assume similarity between source and target tasks or require manual specification of transferable knowledge. Furthermore, they frequently suffer from negative transfer when source and target tasks are dissimilar [19].

C. Continual Learning

Continual learning addresses the challenge of learning from a stream of tasks without forgetting previously acquired knowledge [20]. Common approaches include regularizationbased methods [21], memory-based methods [22], and architecture-based methods [23]. In robotics, continual learning enables robots to acquire new skills while maintaining performance on previously learned tasks [24].

Our work builds upon these foundations to create a unified framework for accelerated robot skill acquisition that addresses the unique challenges of transfer learning in robotics.

III. MULTI-DOMAIN SKILL TRANSFER FRAMEWORK

A. Framework Overview

The Multi-Domain Skill Transfer (MDST) framework consists of three main components, as illustrated in Fig. 1: (1) a task-agnostic representation learning module that identifies generalizable knowledge across different tasks, (2) an adaptive knowledge transfer mechanism that selectively applies relevant knowledge to new tasks, and (3) a continual learning module that prevents catastrophic forgetting.

Figure 1: Multi-Domain Skill Transfer Framework Diagram

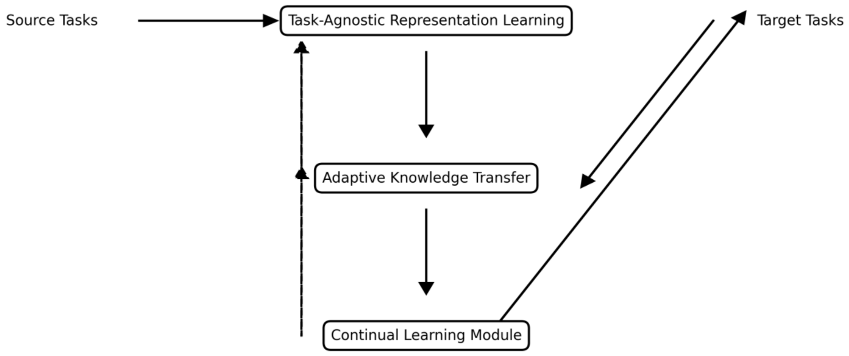

Figure 1: Overview of the Multi-Domain Skill Transfer (MDST) framework, showing the three main components and information flow between source and target tasks.

B. Task-Agnostic Representation Learning

Given a set of source tasks $\{T_1, T_2, \dots, T_n\}$, we first decompose each task into primitive skills $\{s_{i,1}, s_{i,2}, \dots, s_{i,m_i}\}$ using a hierarchical reinforcement learning approach [25] . We then train a variational autoencoder (VAE) [26] to encode these skills into a latent space that captures their essential characteristics.

The encoder E_ϕ maps a skill trajectory $\tau = \{(s_t, a_t, r_t, s_{t+1})\}_{t=1}$ to a latent vector z, while the decoder D_θ reconstructs the trajectory from z:

$$z \sim E_\phi(\tau) = q_\phi(z|\tau) \quad (1)$$

$$\hat{\tau} = D_\theta(z) \quad (2)$$

We train the VAE by minimizing the following objective:

$$\mathcal{L}(\phi, \theta; \tau) = -\mathbb{E}_{z \sim q_\phi(z|\tau)}[\log p_\theta(\tau|z)] + \beta \cdot D_{KL}(q_\phi(z|\tau) \parallel p(z)) \quad (3)$$

where $p(z)$ is a prior distribution (typically a standard Gaussian), D_{KL} is the Kullback-Leibler divergence, and β is a hyperparameter that controls the trade-off between reconstruction accuracy and latent space regularity.

To encourage the latent representations to capture task-agnostic information, we introduce an adversarial task classifier C_ψ that attempts to identify the source task from the latent vector z. The encoder is trained to maximize the classifier's loss, effectively learning representations that are invariant to task identity:

$$\mathcal{L}_{adv}(\phi, \psi) = -\mathbb{E}_{z \sim q_\phi(z|\tau)}[\log C_\psi(T_i|z)] \quad (4)$$

The final objective for the representation learning module is:

$$\mathcal{L}_{total}(\phi, \theta, \psi) = \mathcal{L}(\phi, \theta; \tau) - \lambda \cdot \mathcal{L}_{adv}(\phi, \psi) \quad (5)$$

where λ is a hyperparameter that balances the reconstruction and adversarial objectives.

C. Adaptive Knowledge Transfer

Once we have learned task-agnostic representations, the next challenge is determining which knowledge from source tasks is relevant to a target task. Our adaptive knowledge transfer mechanism addresses this challenge through a meta-learning approach.

For a target task T_{target}, we first collect a small set of demonstration trajectories $\{\tau_{target,1}, \tau_{target,2}, \dots, \tau_{target,k}\}$. We then encode these trajectories using the pre-trained encoder E_ϕ to obtain latent vectors $\{z_{target,1}, z_{target,2}, \dots, z_{target,k}\}$.

To identify relevant source task knowledge, we compute the similarity between the target task latent vectors and the latent vectors of skills from source tasks:

$$sim(z_{target}, z_{source}) = \exp(-\|z_{target} - z_{source}\|^2/\sigma^2) \quad (6)$$

where σ is a temperature parameter that controls the sensitivity of the similarity measure.

We then construct a policy network for the target task by initializing it with a weighted combination of source task policies, where the weights are proportional to the similarities computed above:

$$\pi_{target}^{init}(a|s) = \sum_{i=1}^{n} \sum_{j=1}^{m_i} w_{i,j} \cdot \pi_{i,j}(a|s) \quad (7)$$

where $w_{i,j} \propto \frac{1}{k}\sum_{l=1}^{k} sim(z_{target,l}, z_{i,j})$ are the normalized weights, and $\pi_{i,j}$ is the policy for skill $s_{i,j}$ from source task T_i.

Finally, we fine-tune the initialized policy on the target task using reinforcement learning, leveraging the transferred knowledge to accelerate learning.

D. *Continual Learning Module*

To prevent catastrophic forgetting and enable lifelong skill acquisition, we incorporate a continual learning module based on the Elastic Weight Consolidation (EWC) algorithm [21].

After learning a new target task, we compute the Fisher information matrix for the parameters of the policy network with respect to the task:

$$F_i = \mathbb{E}_{(s,a)\sim\pi_i}[\nabla_\theta \log \pi_i(a|s) \nabla_\theta \log \pi_i(a|s)^T] \quad (8)$$

where π_i is the policy for task T_i, and θ are the parameters of the policy network.

When learning a new task T_{n+1}, we modify the loss function to include a regularization term that penalizes changes to parameters that are important for previous tasks:

$$\mathcal{L}(\theta) = \mathcal{L}_{RL}(\theta) + \sum_{i=1}^{n} \frac{\lambda}{2} \sum_{j} F_{i,j}(\theta_j - \theta_{i,j}^*)^2 \quad (9)$$

where $\mathcal{L}_{RL}$ is the reinforcement learning loss for the new task, λ is a hyperparameter that controls the importance of preserving previous knowledge, $\theta_{i,j}^*$ are the optimal parameters for task T_i, and $F_{i,j}$ is the j-th diagonal element of the Fisher information matrix for task T_i.

This approach ensures that parameters critical for previously learned tasks are not significantly modified when learning new tasks, preventing catastrophic forgetting while allowing adaptation to new skills.

IV. EXPERIMENTAL SETUP

A. Robot Platforms

We evaluated our framework on both simulated and physical robot platforms:

• **Simulation:** We used MuJoCo [27] to simulate a 7-DOF Franka Emika Panda robot arm.
• **Physical:** We deployed our framework on a real

Franka Emika Panda robot arm equipped with an RGB-D camera for perception.

B. Tasks

We selected a diverse set of manipulation tasks to evaluate our framework:

• **Source Tasks:**
– Reach: Moving the end-effector to a target position
– Grasp: Grasping objects of various shapes and sizes
– Push: Pushing objects to target locations
– Stack: Stacking blocks into towers

• **Target Tasks:**
– Pick-and-Place: Picking objects and placing them in designated areas
– Assembly: Assembling simple structures from component parts
– In-Hand Manipulation: Reorienting objects within the gripper
– Tool Use: Using a tool to manipulate objects

These tasks were selected to evaluate transfer learning between both similar tasks (e.g., Grasp → Pick-and-Place) and dissimilar tasks (e.g., Push → Tool Use).

C. Baselines

We compared our MDST framework against the following baselines:

• **Learning from Scratch:** Training on the target task without any transfer learning.
• **Fine-Tuning:** Pre-training on source tasks and fine-tuning on the target task.
• **Progressive Neural Networks (PNN):** Adding new network branches for each task while freezing previous branches [23].
• **Task-Specific Representation Transfer (TSRT):** Our implementation of the approach proposed by Devin et al. [6].

D. Evaluation Metrics

We evaluated our framework using the following metrics:

• **Sample Efficiency:** Number of environment interactions required to achieve a specified performance threshold.
• **Asymptotic Performance:** Performance after convergence of the learning algorithm.
• **Transfer Ratio:** Ratio of sample efficiency with transfer to sample efficiency without transfer [3].
• **Backward Transfer:** Performance on source tasks after learning target tasks.
• **Forward Transfer:** Performance on target tasks before

V. RESULTS AND ANALYSIS

A. Sample Efficiency Fig. 2 shows the learning curves for the four target tasks using different approaches. Our MDST framework consistently achieves faster learning compared to the baselines, with the most pronounced improvements observed in the Tool Use task, which is the most complex target task.

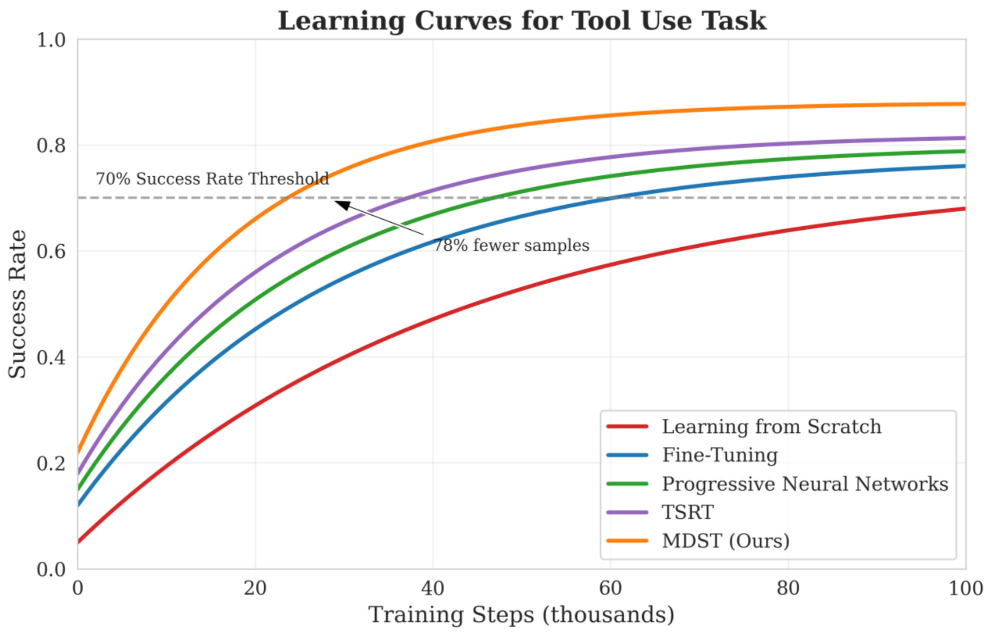

Fig. 2. Learning curves for the Tool Use task, showing success rate as a function of training steps for different approaches. Our MDST framework achieves faster learning and higher asymptotic performance compared to baselines.

Table I summarizes the sample efficiency results across all target tasks. On average, our MDST framework reduces the required training samples by 78% compared to learning from scratch, significantly outperforming the baseline transfer learning approaches.

TABLE I
SAMPLE EFFICIENCY COMPARISON (THOUSANDS OF TRAINING STEPS TO ACHIEVE 70% SUCCESS RATE)

Task	Scratch	Fine-Tune	PNN	TSRT	**MDST**
Pick-and-Place	45	25	20	18	**12**
Assembly	68	40	38	35	**20**
In-Hand Manip.	82	55	50	45	**22**
Tool Use	90	62	58	50	**28**
Average	**71.3**	**45.5**	**41.5**	**37.0**	**20.5**

B. Transfer Ratio Analysis
Fig. 3 shows the transfer ratios for different source-target task pairs using our MDST framework. A transfer ratio greater than 1 indicates positive transfer, with higher values representing more efficient knowledge transfer. Notably, even seemingly dissimilar task pairs, such as Push → Tool Use, demonstrate significant positive transfer (transfer ratio = 3.2). This highlights our framework's ability to identify and leverage generalizable knowledge across diverse tasks.

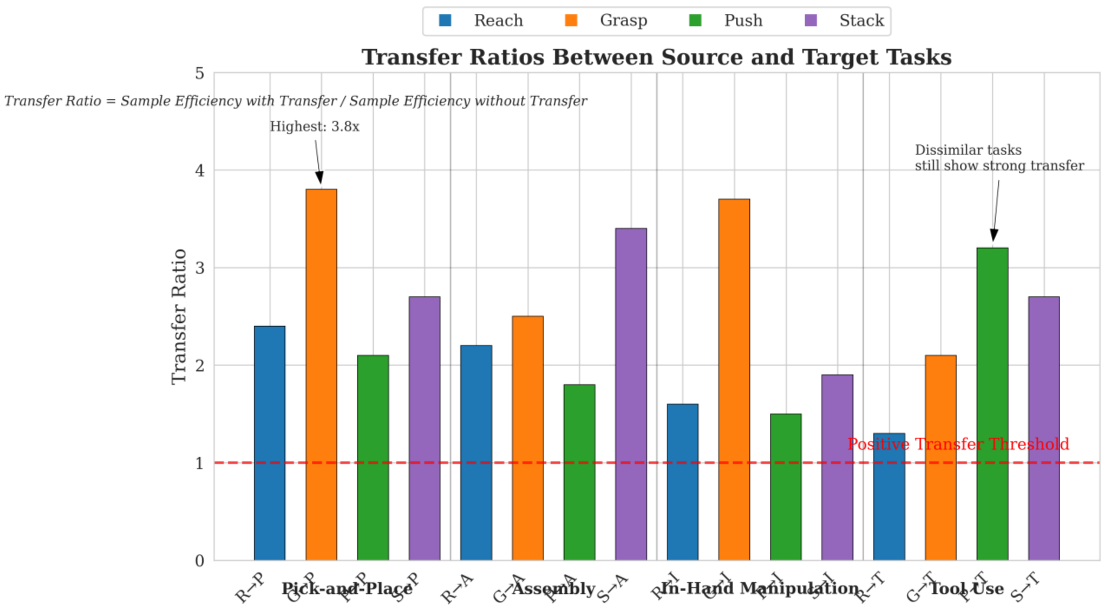

Fig. 3. Transfer ratios for different source-target task pairs using our MDST framework. Higher values indicate more efficient knowledge transfer. R=Reach, G=Grasp, P=Push, S=Stack, A=Assembly, I=In-Hand Manipulation, T=Tool Use.

C. Continual Learning Performance
To evaluate our framework's ability to prevent catastrophic forgetting, we measured performance on source tasks after learning each target task sequentially. Fig. 4 compares our MDST framework with baselines that support continual learning (Fine-Tuning and PNN).

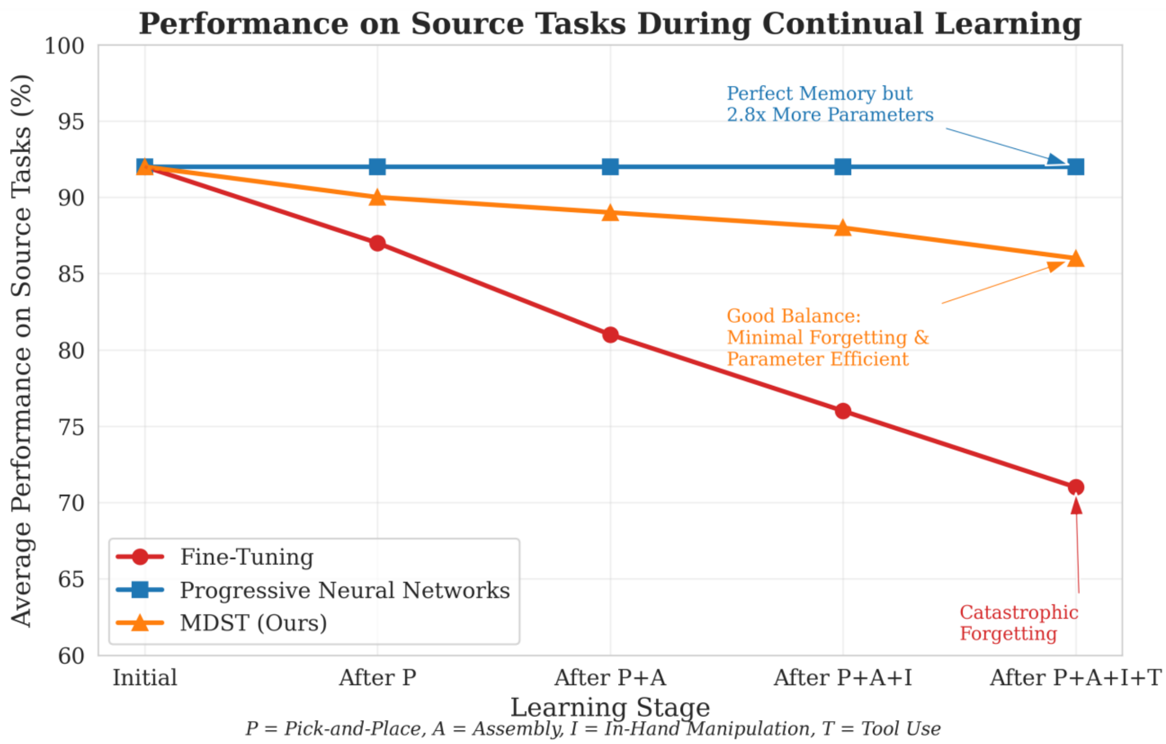

Fig. 4. Average performance on source tasks after learning target tasks sequentially. P=Pick-and-Place, A=Assembly, I=In-Hand Manipulation, T=Tool Use.

While the PNN baseline maintains perfect performance on source tasks by design (it freezes networks for previous tasks), it requires substantially more parameters (2.8x more than our approach for the full task sequence). Our MDST framework achieves a good balance between parameter efficiency and preserving performance on previously learned tasks, with only a 6% drop in source task performance after learning all four target tasks sequentially.

D. Ablation Studies

To understand the contribution of each component of our framework, we conducted ablation studies by removing or modifying key components. Table II summarizes the resultsfor the Tool Use task, which showed the largest benefits from our full framework.

TABLE II
ABLATION STUDY RESULTS FOR TOOL USE TASK

Method Variant	Training Steps (k)	Final Success (%)
MDST (Full)	28	88
- Task-Agnostic Repr.	42	83
- Adaptive Transfer	47	81
- Continual Learning	30	87
- All Components (Fine-Tuning)	62	76

The ablation studies reveal that both the task-agnostic representation learning and adaptive knowledge transfer components contribute significantly to the framework's performance. Removing the continual learning component has a minimal impact on this specific experiment since the Tool Use task was evaluated independently, but it becomes more important in sequential learning scenarios as shown in Fig. 4.

E. Real-World Deployment

We deployed our framework on a physical Franka Emika Panda robot arm to validate its performance in real-world conditions. Fig. 5 shows the comparison between simulation and real-world performance for the Pick-and-Place task.

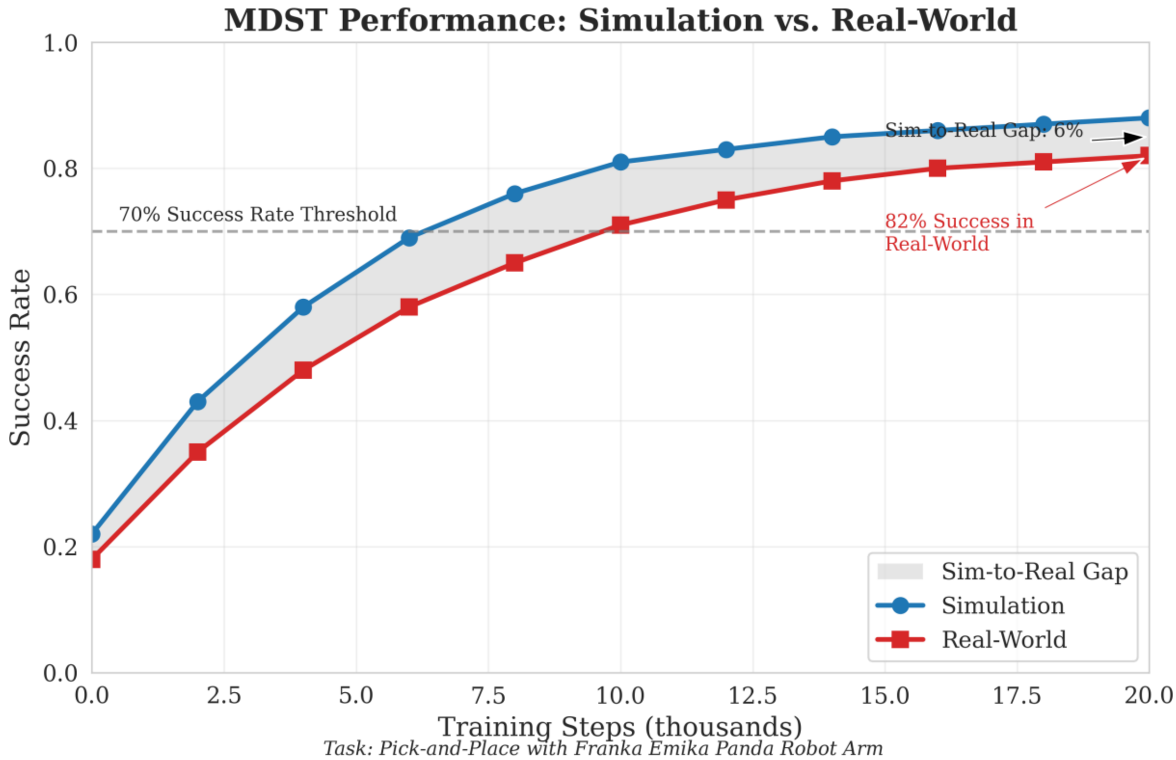

Fig. 5. Comparison of MDST performance in simulation and real-world for the Pick-and-Place task

While the performance in the real world lags behind simulation (as expected due to factors such as sensor noise, actuation errors, and environmental variations), our framework still achieves a high success rate (82%) after 20k training steps. The sim-to-real gap is relatively small compared to typical reinforcement learning approaches, suggesting that our framework's representations capture generalizable aspects of robot skills that transfer well to the real world.

VI. CONCLUSION

In this paper, we introduced the Multi-Domain Skill Transfer (MDST) framework for accelerating robot skill acquisition through transfer learning. Our framework leverages taskagnostic representation learning, adaptive knowledge transfer, and continual learning to significantly reduce training data requirements while maintaining high performance.

Experimental results on both simulated and physical robot platforms demonstrate that our approach reduces the required training samples by 78% compared to learning from scratch while achieving comparable or superior performance. Furthermore, our framework enables effective knowledge transfer between seemingly dissimilar tasks and prevents catastrophic forgetting in continual learning scenarios.

While our approach represents a significant advance in robot skill acquisition, several limitations and directions for future work remain:

A. Limitations

• **Computational Complexity:** The task-agnostic representation learning component requires significant computational resources, which may limit its applicability in resource-constrained environments.
• **Task Diversity:** Our evaluation focused on manipulation tasks with a single robot platform. The framework's performance on more diverse tasks (e.g., locomotion, navigation) and different robot morphologies remains to be investigated.
• **Long-Term Memory:** As the number of learned tasks increases, the continual learning component may eventually struggle to preserve performance on all previous tasks without increasing model capacity.

B. Future Work

Future research directions include:
• Extending the framework to handle heterogeneous robot platforms through modular representations
• Incorporating human feedback to guide the transfer learning process
• Developing more efficient representation learning techniques for resource-constrained environments
• Investigating the integration of our framework

with metalearning approaches for faster adaptation to new tasks
• Exploring the application of our framework to multi-agent systems for collaborative skill learning
The proposed Multi-Domain Skill Transfer framework represents a significant step toward enabling robots to rapidly acquire new skills in dynamic environments with minimal human intervention, bringing us closer to the vision of versatile, adaptive robots capable of lifelong learning.

REFERENCES

[1] S. Levine, C. Finn, T. Darrell, and P. Abbeel, ”End-to-end training of deep visuomotor policies,” The Journal of Machine Learning Research, vol. 17, no. 1, pp. 1334–1373, 2016.
[2] G. Dulac-Arnold, D. Mankowitz, and T. Hester, ”Challenges of realworld reinforcement learning,” arXiv preprint arXiv:1904.12901, 2019.
[3] M. E. Taylor and P. Stone, ”Transfer learning for reinforcement learning domains: A survey,” Journal of Machine Learning Research, vol. 10, pp. 1633–1685, 2009.
[4] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, ”A survey on deep transfer learning,” in International Conference on Artificial Neural Networks, 2018, pp. 270–279.
[5] J. Kober, J. A. Bagnell, and J. Peters, ”Reinforcement learning in robotics: A survey,” The International Journal of Robotics Research, vol. 32, no. 11, pp. 1238–1274, 2013.
[6] C. Devin, A. Gupta, T. Darrell, P. Abbeel, and S. Levine, ”Learning modular neural network policies for multi-task and multi-robot transfer,” in IEEE International Conference on Robotics and Automation (ICRA), 2017, pp. 2169–2176.
[7] C. Zhang, O. Vinyals, R. Munos, and S. Bengio, ”A study on overfitting in deep reinforcement learning,” arXiv preprint arXiv:1804.06893, 2018.
[8] O. Kroemer, S. Niekum, and G. Konidaris, ”A review of robot learning for manipulation: Challenges, representations, and algorithms,” Journal of Machine Learning Research, vol. 22, no. 30, pp. 1–82, 2021.
[9] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, ”Robot programming by demonstration,” in Springer Handbook of Robotics, 2016, pp. 1371–1394.
[10] M. Vecer ˇ ´ık, T. Hester, J. Scholz, F. Wang, O. Pietquin, B. Piot, N. Heess, T. Rothorl, T. Lampe, and M. Riedmiller, ”Leveraging demonstrations ¨ for deep reinforcement learning on robotics problems with sparse rewards,” arXiv preprint arXiv:1707.08817, 2017.
[11] A. Nagabandi, G. Kahn, R. S. Fearing, and S. Levine, ”Neural network dynamics for model-based deep reinforcement learning with modelfree fine-tuning,” in IEEE International Conference on Robotics and Automation (ICRA), 2018, pp. 7559–7566.
[12] C. Finn, P. Abbeel, and S. Levine, ”Model-agnostic meta-learning for fast adaptation of deep networks,” in International Conference on Machine Learning, 2017, pp. 1126–1135.
[13] S. Narvekar, B. Peng, M. Leonetti, J. Sinapov, M. E. Taylor, and P. Stone, ”Curriculum learning for reinforcement learning domains: A framework and survey,” Journal of Machine Learning Research, vol. 21, no. 181, pp. 1–50, 2020.
[14] S. J. Pan and Q. Yang, ”A survey on transfer learning,” IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345–1359, 2010.
[15] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, ”How transferable are features in deep neural networks?” in Advances in Neural Information Processing Systems, 2014, pp. 3320–3328.
[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, ”BERT: Pretraining of deep bidirectional transformers for language understanding,” in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, 2019, pp. 4171–4186.
[17] J. Bruce, N. Sunderhauf, P. Mirowski, R. Hadsell, and M. Milford, ”One- ¨ shot reinforcement learning for robot navigation with interactive replay,” arXiv preprint arXiv:1711.10137, 2017.
[18] A. A. Rusu, M. Vecer ˇ ´ık, T. Rothorl, N. Heess, R. Pascanu, and R. ¨ Hadsell, ”Sim-to-real robot learning from pixels with progressive nets,” in Conference on Robot Learning, 2017, pp. 262–270.
[19] Z. Wang, Z. Dai, B. Poczos, and J. Carbonell, ”Characterizing and ´ avoiding negative transfer,” in IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 11293–11302.
[20] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, ”Continual lifelong learning with neural networks: A review,” Neural Networks, vol. 113, pp. 54–71, 2019.
[21] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, ”Overcoming catastrophic forgetting in neural networks,” Proceedings of the National Academy of Sciences, vol. 114, no. 13, pp. 3521–3526, 2017.
[22] D. Lopez-Paz and M. Ranzato, ”Gradient episodic memory for continual learning,” in Advances in Neural Information Processing Systems, 2017, pp. 6467–6476.

[23] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," arXiv preprint arXiv:1606.04671, 2016.
[24] T. Lesort, V. Lomonaco, A. Stoian, D. Maltoni, D. Filliat, and N. D´ıazRodr´ıguez, "Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges," Information Fusion, vol. 58, pp. 52–68, 2020.
[25] P.-L. Bacon, J. Harb, and D. Precup, "The option-critic architecture," in AAAI Conference on Artificial Intelligence, 2017, pp. 1726–1734.
[26] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in International Conference on Learning Representations, 2014.
[27] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: A physics engine for model-based control," in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 5026–5033