2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Learning from Demonstration Framework for Multi-Robot Systems Using Interaction Keypoints and Soft Actor-Critic Methods

Vishnunandan L. N. Venkatesh and Byung-Cheol Min

***Abstract*— Learning from Demonstration (LfD) is a promising approach to enable Multi-Robot Systems (MRS) to acquire complex skills and behaviors. However, the intricate interactions and coordination challenges in MRS pose significant hurdles for effective LfD. In this paper, we present a novel LfD framework specifically designed for MRS, which leverages visual demonstrations to capture and learn from robot-robot and robot-object interactions. Our framework introduces the concept of Interaction Keypoints (IKs) to transform the visual demonstrations into a representation that facilitates the inference of various skills necessary for the task. The robots then execute the task using sensorimotor actions and reinforcement learning (RL) policies when required. A key feature of our approach is the ability to handle unseen contact-based skills that emerge during the demonstration. In such cases, RL is employed to learn the skill using a classifier-based reward function, eliminating the need for manual reward engineering and ensuring adaptability to environmental changes. We evaluate our framework across a range of mobile robot tasks, covering both behavior-based and contact-based domains. The results demonstrate the effectiveness of our approach in enabling robots to learn complex multi-robot tasks and behaviors from visual demonstrations.**

I. INTRODUCTION

Learning from Demonstration (LfD) represents a pivotal advancement in robotics, shifting paradigmatic approaches towards more intuitive, efficient skill acquisition in intelligent systems. By leveraging human demonstrations, LfD facilitates the teaching of complex behaviors to robots without the need for intricate programming, embodying a natural progression towards more accessible human-robot interactions. This methodology not only simplifies the programming landscape but also heralds a new era of potential applications that span the manufacturing, healthcare, and surveillance sectors [1], [2]. The growing importance of LfD is evident from its increasing presence in academic research [3].

Despite its growing promise, the exploration of LfD within multi-robot systems (MRS) remains nascent, presenting a unique array of challenges and opportunities [4], [3]. The complexity inherent to MRS, marked by intricate robot-robot and robot-environment interactions-significantly compounds the challenges faced in LfD. Multi-robot contexts escalate the variables and potential interactions exponentially, complicating both the design and control of these systems compared to their single-robot counterparts. These complexities necessitate innovative approaches in the development of LfD algorithms capable of navigating the multifaceted dynamics of MRS, including coordination and data processing between multiple agents [3].

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1846221. The authors are with SMART Lab, Department of Computer and Information Technology, Purdue University, West Lafayette, IN 47907, USA lvenkate@purdue.edu, minb@purdue.edu

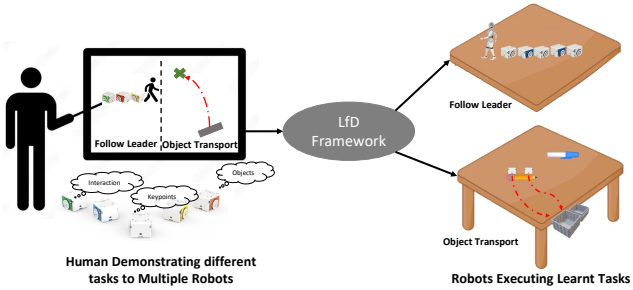


Fig. 1: Concept for learning from demonstration for multi-robot systems (MRS). The human expert demonstrator shows multiple tasks to the MRS, which are then learned and executed.

Recent research efforts in LfD for MRS have explored a spectrum of demonstration methodologies, ranging from simulator-based to vision-based demonstrations, each with its advantages and limitations. Despite significant advances, current approaches often struggle with task versatility and an over-reliance on extensive demonstration sets. Moreover, the predominant focus on nonvisual demonstration methods hampers intuitive natural human-robot communication, underscoring the critical need for more accessible, vision-based frameworks [5].

Our work introduces a vision-based learning-from-demonstration framework for multi-robot systems, leveraging **Visual Interaction Keypoints** and **Soft Actor-Critic (SAC)** methods. The choice of visual cues as the primary mode of demonstration is motivated by their inherent capacity for facilitating intuitive interactions between humans and robots [5], despite their inherent shortcoming in conveying tactile information. To overcome this challenge, we incorporate the SAC algorithm, which enables robots to master tasks that require physical contact. Central to our method is the utilization of a binary decision classifier alongside interaction keypoints, which collectively fine-tune the reward mechanism without resorting to modeled engineering. The interaction keypoints pinpoint crucial moments of interaction within the environment, such as instances of contact between a robot and an object or the proximity of robots to each other, thereby dividing complex tasks into manageable subtasks. This segmentation not only boosts the efficiency of the learning process but also improves the clarity with which MRS can be understood and interpreted. A conceptual overview of our framework is presented in Fig. 1.

MRS encompass various task categories, including navigation and exploration [6], coordination and communica-

tion, decision-making and planning, assembly and manufacturing, manipulation and grasping, and intensive contact-based tasks [7]. Our framework simplifies this complexity by classifying tasks into **Behavior-Based** and **Contact-Based** categories. Behavior-based tasks encompass activities like pattern formation and surveillance, while contact-based tasks involve direct physical interactions, such as pushing or lifting. What distinguishes our framework is its novel use of vision-based demonstrations to effectively learn and execute tasks within these categories. It efficiently processes behavior-based tasks using interaction keypoints from a single clear demonstration. For contact-based tasks, although multiple demonstrations may be required, the approach remains more streamlined and less demanding than traditional machine learning-based methods, enhancing both efficiency and applicability.

The main contributions of this paper are:

- We propose a novel LfD framework for MRS that utilizes *Visual Keypoint* inference and *SAC* methods, addressing the gap in current research. This framework is capable of performing behaviour and contact-based tasks.
- We evaluate our framework through real-time experiments on diverse tasks, demonstrating its efficacy in collaborative behavior and contact-based tasks.

The remainder of this paper is organized as follows. We begin with a comprehensive literature review focused on MRS LfD, followed by a detailed definition of the problem. The following sections outline our methodology, experimental setup, and results. We conclude with a discussion of the limitations of our framework and potential directions for future research.

II. RELATED WORK

While there is a plethora of research conducted in the domain of single robot learning from demonstration [3], [4], [8], this review of the literature explores MRS in the context of LfD. It covers a range of demonstration methods, from simulator demonstrations to vision-based human demonstrations, emphasizing the significant role of MRS in LfD research.

Notable studies include [9], which introduces a confidence execution algorithm for collaborative ball sorting, featuring an adaptive interruption mechanism for when robots require additional human demonstrations due to low confidence. Knepper et al. discuss task conveyance through geometric CAD designs for the assembly of furniture by robots with specialized roles [10]. Huang et al. combine human demonstrations with vision for bi-manual surgical tasks, employing Gaussian Mixture Models for learning, illustrating collaborative work even within a single robotic system [5]. The authors in [11] and [12] demonstrate teleoperation for navigation and door opening and a hierarchical reinforcement learning (RL) framework for abstract behavior tasks, respectively. In robot soccer, Freelan et al. in [13] employs reinforcement learning and state-space automata [14] to teach set plays, highlighting the domain's extensive research in LfD and reinforcement learning.

However, these advances often face challenges such as task specificity, extensive demonstration requirements, and lack of intuitive vision-based communication. Moreover, the limited research in the context of LfD for MRS shows the inherent complexity when dealing with an MRS that includes interactions that can affect the environment exponentially. Our framework addresses these issues by leveraging vision-based demonstrations and interaction keypoints [15], [16] for a wide range of tasks. Inspired by [17], we incorporate SAC methods for their real-time efficacy in complex task learning. Crucially, our framework processes behavior-based tasks with Interaction Keypoints from a single, clear demonstration, streamlining the learning process. For contact-based tasks, while multiple demonstrations may still be necessary, our method is significantly more streamlined and less demanding than traditional machine learning approaches [18], [3], [19], enhancing both efficiency and applicability.

III. PROBLEM DEFINITION

Our framework addresses the challenge of instructing MRS to perform tasks based on visual demonstrations, categorized into behavior-based and contact-based tasks. We formalize the inputs, processes, and outputs as follows:

Inputs:

- $D = \{f_1, f_2, \ldots, f_n\}$: A sequence of n frames from visual demonstrations captured by a 2D camera.
- $O = \{o_1, o_2, \ldots, o_m\}$: A set of m recognized objects within the frames.
- $R = \{r_1, r_2, \ldots, r_k\}$: A set of k identified robots within the frames.
- G: The goal positions for objects and robots that conform to the deduced goal state from the demonstrations.

Outputs:

- $IK = \{ik_1, ik_2, \ldots, ik_p\}$: Interaction Keypoints, indicating significant moments of interaction. p represents the total number of interaction keypoints identified.
- T_P: A Task Policy for performing the task.
- S_{RL}: Skills developed through Reinforcement Learning for contact-based tasks.

Problem 1: Behavior-Based Task Learning. The behaviour-based task learning problem involves deriving a task policy T_P from visual demonstrations, utilizing interaction keypoints and the spatial dynamics of robots and objects:

$$T_P = f(D, IK, O, R, G) \tag{1}$$

where the function f encapsulates the algorithms or set of processes that the LfD framework employs to interpret the visual demonstrations. The resulting task policy T_P details the actions or behaviors that robots are to perform to complete the demonstrated task.

Problem 2: Contact-Based Task Learning. The contact-based task learning problem extends behaviour-based learning by incorporating reinforcement learning for learning skills that involve physical contact interactions in the task,

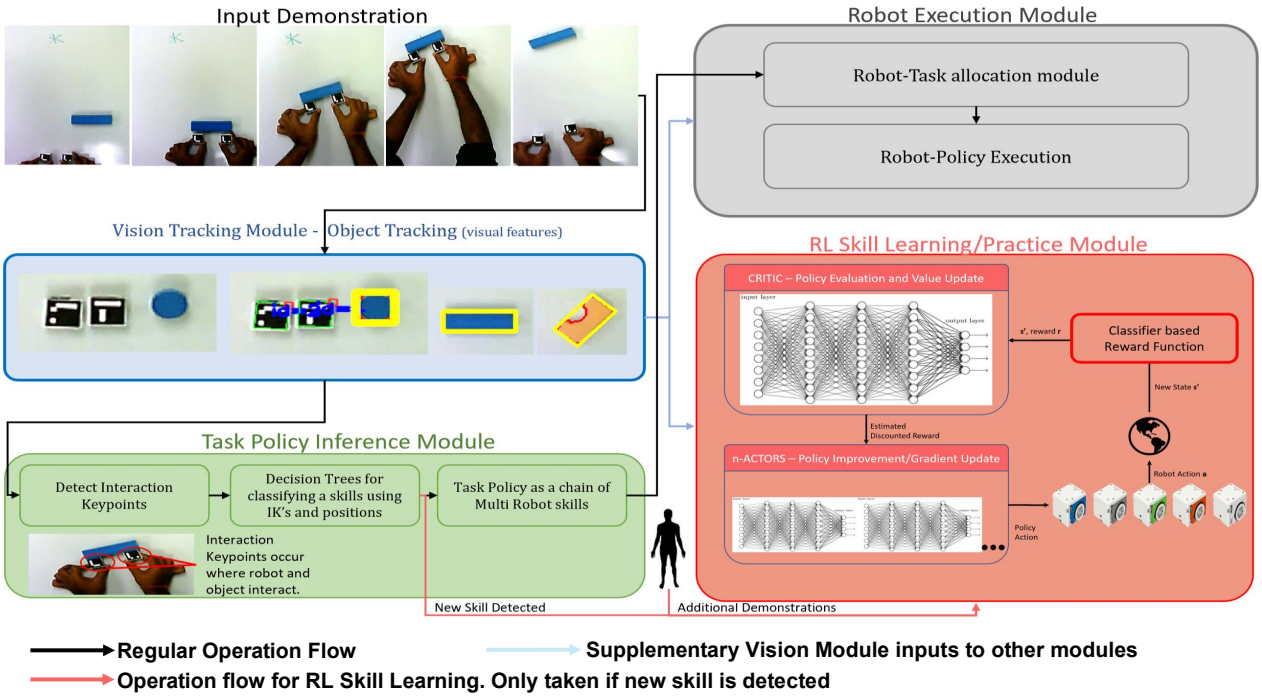


Fig. 2: The proposed learning from demonstration for multi-robot systems framework follows a streamlined process: Human experts visually demonstrate tasks, captured by a 2D camera. These demonstrations undergo feature extraction in the **Vision Tracking Module**. The **Task Policy Inference Module** segments the demonstrations and identifies *Interaction Keypoints*, forming a *Task Policy*. When new contact skills arise, the **RL skill Learning/Practice Module**, using SAC networks, learns them with guidance from a binary decision classifier's reward signals. Finally, the **Robot Execution Module** allocates and executes tasks across multiple robots, showcasing the adaptability of the framework in various environmental conditions.

requiring additional demonstrations to train a decision classifier C for goal state recognition:

$$S_{RL} = g(T_P, C(D_{add}), IK, O, R, G) \tag{2}$$

where the function g represents the learning process that integrates T_P, IK, O, R, and G, with the output of a decision classifier C trained on additional demonstrations D_{add} to synthesize reinforcement learning skills S_{RL} for contact-based tasks.

Our aim is to develop an integrated set of execution policies Π that enables the MRS to autonomously perform complex tasks, aligned with the learned behavior-based and contact-based policies, and ultimately matching the goal configuration G:

$$\Pi = h(T_P, S_{RL}, G) \tag{3}$$

where the function h denotes the integration process that amalgamates the task policy T_P, skills S_{RL} acquired from reinforcement learning for contact-based interactions, and the goal states G, to produce a set of executable policies Π for the multi-robot system. This integration represents the culmination of the learning from demonstration process, enabling the robots to perform tasks effectively.

This symbolic representation defines the transformation from input demonstrations to executable multi-robot behaviors, setting the foundation for detailed formulation in subsequent sections of the paper.

IV. Learning from Demonstration Framework for MRS

The core of our approach is encapsulated within a LfD framework, designed to enable MRS to learn and execute tasks through visual demonstration, as depicted in Fig. 2. This process begins with the acquisition of task demonstrations D via a 2D camera within the vision system, capturing RGB imagery that provides the robots with visual cues necessary for task performance.

The **Vision Tracking Module** represents the first processing stage within our framework. It analyzes the video demonstrations, which vary in length depending on the complexity of the task at hand. The module's primary function is to extract prominent features concerning the robots R and objects O within each frame. These data are crucial to building a comprehensive understanding of the subtasks to be learned.

Following feature extraction, the **Task Policy Inference Module** takes over to dissect the demonstrations into smaller, interpretable segments. This process identifies *Interaction Keypoints* IK, which are critical for delineating subtasks and individual robot apriori skills from the demonstration. These keypoints enable the formation of a *Task Policy* T_P, a sequence of actions representing the learned skills and decisions.

Should the inference process reveal the necessity for a robot to learn a new contact-based skill not encapsulated within the existing repertoire, the **RL Module** is activated. Here, Soft Actor-Critic algorithms are employed to teach

the robots these new skills, with the support of additional demonstrations as needed. The binary decision classifier C refines the reward structure, guiding the RL process to ensure effective skill acquisition.

The final step in our methodology involves the **Robot Execution Module**, which is tasked with the allocation of learned subtasks to the respective robots in the MRS. This module effectively translates the high-level *Task Policy* T_P into actionable steps, using both preexisting apriori skills and newly learned RL policies for task execution. Through this vision-based LfD framework, we equip MRS with the capacity not only to replicate demonstrated tasks, but also to apply these learned behaviors to novel scenarios, bridging the gap between demonstration and autonomous execution.

A. Vision Tracking Module

The Vision Tracking Module stands as a critical component of our framework, underpinning both real-time task execution and subsequent learning. It tracks a range of environmental features—specifically, the 2D positions (x, y) of objects O and robots R, as well as their goal positions G. This module also captures more nuanced attributes such as object shape and color, charting the relationships among objects and robots.

The module assumes an occlusion-free environment, ensuring clear visibility of all objects and robots for reliable data capture. While this simplifies tracking, we anticipate future iterations to tackle partial occlusions, broadening the framework's versatility. The approach relies on the detectability of objects to maintain a finite, yet scalable, database for object profiles, supporting system adaptability by accommodating new objects as needed.

Object detection leverages the power of Mask R-CNN [20], a state-of-the-art deep learning technique known for its robust object detection and instance segmentation capabilities. Complementing this, shape and color detection are performed using functionalities provided by OpenCV, enabling the precise identification of object features such as color, shape, corners and centers upon detection. For robot tracking, we use ArUco markers [21], designed for efficient pose estimation.

B. Task Policy Inference Module

This module leverages visual features captured by the Vision Tracking Module to identify *Interaction Keypoints*, which are critical for task segmentation and efficiency. Drawing inspiration from methods that utilize keypoints for task decomposition [15], [16], IKs enable the system to break complex tasks into smaller, manageable subtasks. This granularity allows each robot in the MRS to focus on specific segments of the task, streamlining the learning process, and improving system interpretability.

Interaction Types and Definitions: The module distinguishes four types of interactions within the MRS context:

1) *Object-Robot Interactions* (ϕ_i): Interactions between objects and robots.
2) *Object-Object Interactions* (ψ_i): Interactions among objects.
3) *Robot-Robot Interactions* (ω_i): Interactions among robots.
4) *Behavior Triggering Keypoints*: Environmental changes, such as the introduction of a new artifact, prompting a state transition from St to St^* and act as an action trigger.

Segmentation and Interaction Features: *Interaction Keypoints* aid in the temporal segmentation of tasks, highlighting each robot's role and interactions. Binary interaction features (ϕ_i, ψ_i, ω_i) denote the presence (1) or absence (0) of interactions. The system also tracks the relative and absolute positions of robots (Rr_i, Ar_i) and objects (Ro_i, Ao_i), along with motion flags $f(Ro_i)$ and $f(Rr_i)$ to indicate movement. We can denote each segment at time t for multiple robots as:

$$\Theta_{it} = (\theta_1, \theta_2, \ldots, \theta_r) \tag{4}$$

where task segment for a robot, θ_r, at an *Interaction Keypoint* is defined as:

$$\theta_r = (\phi_i, Ro_i, f(Ro_i), Ao_i, \psi_i, \omega_i, Rr_i, f(Rr_i), Ar_i) \tag{5}$$

These segments undergo classification via a decision tree [22], [23] to assign class labels C_i, corresponding to specific robot skills, whether pre-learned or acquired through RL.

Policy Formation: The resulting policy, Π, sequences robot interactions and actions as follows:

$$\Pi = \{(C_1, g_1, \Theta_{1t}), (C_2, g_2, \Theta_{2t}), \ldots, (C_m, g_m, \Theta_{mt})\} \tag{6}$$

where g_m denotes the goal state for each segment Φ_{mt}, reflecting the final environmental state within the segment. Class label C_m links to a specific skill set for execution during task realization.

Task Dimensionality: Task dimensionality, spanning from m to $r \times m$, is adjusted based on the synchronization of the robot operations, ensuring the scalability of performance with the quantity of the robot, mainly affecting the data storage needs.

C. RL Skill Learning

Our framework utilizes RL within the LfD paradigm, specifically focusing on demonstrations to generate reward signals for agent training. This is crucial for introducing new, often complex, collaborative contact-based manipulation skills. Our model approaches these multi-robot manipulation challenges as model-free RL problems, activating this module when new skills are identified.

Soft Actor-Critic (SAC) Method: The SAC method is chosen for its efficiency and suitability for real-time execution within dynamic multi-robot systems. Combining actor-critic architecture with soft Q-learning, SAC ensures stable, adaptive learning, balancing exploration and exploitation. It employs a shared critic and individual actor networks for personalized learning, supported by a replay buffer for

leveraging past experiences. This approach enhances responsiveness and learning diversity, following the SAC principles by [24] and implemented in stable baselines [25].

State Space and Action Space: The SAC framework's state space is enriched with a 224x224 pixel image input processed via a Resnet 50 architecture, effectively capturing visual features crucial for task execution. This visual input is combined with precise positional data of both the robots and objects within the environment, ensuring comprehensive situational awareness. Each robot's action space is defined by dual-speed parameters, controlling the two motors of our mobile robot platform. This action space is deliberately kept simple and generic, allowing for straightforward adaptation to other mobile robot systems.

Rewards: Manual reward crafting for multi-robot tasks presents significant challenges, especially in achieving generalization across diverse tasks. By integrating additional demonstrations, our approach generalizes the reward function, leveraging a binary decision classifier for the determination of the reward signal, thus facilitating more intuitive and effective reward configurations.

Given the demonstrations' capability to highlight the task's goal through specific frames as shown in [2], we utilize these frames to discern positive and negative goal examples. Formally, let $D = (I_n, y_n)$ represent the dataset, where I_n are the goal image frames, and y_n are the binary labels indicating positive (1) or negative (0) goal states.

The binary decision classifier C is trained on D to distinguish between goal states and non-goal states, optimizing its ability to function as a proxy for the reward function:

$$C(I_n) \rightarrow y_n \tag{7}$$

The total reward signal R for the SAC is derived from the classifier's output and is structured as follows, incorporating weights for balance:

$$R = w_1 \cdot C(I_n) + w_2 \cdot IK_{reward} - w_3 \cdot IKFail_{penalty} \tag{8}$$

where w_1, w_2, and w_3 are weights that adjust the influence of each component on the total reward. IK_{reward} is designed to be minimal and a significant penalty for failures, emphasizing the achievement of the task's primary goal over mere interaction with keypoints. This nuanced reward strategy ensures that agents are incentivized to pursue the overarching objectives while maintaining focus over individual robot objectives efficiently.

D. Robot Execution Module

The Robot Execution Module translates the *Task Policy*, derived from Section IV-B, into executable actions through Task Allocation. This allocation assigns robots either predefined apriori skills or RL skill policies from Section IV-C.

Apriori skills are sensory-motor actions each robot executes independently, aligned with specific goal states within the policy. Our framework includes four critical apriori skills as depicted in Fig. 3: 1) *Approach*, where a robot moves towards a target zone around an object or another robot; 2) *Move to Contact*, achieving physical connection with an object; 3) *Detach Contact*, where the robot disengages from the object; and 4) *Retreat*, withdrawing from the object's vicinity. Executed with a precise low-level PD controller, these skills enable basic task performance and fundamental *Interaction Keypoint* detection, such as object-object, object-robot and robot-robot interactions. This foundational capability is vital for understanding interactions within our system and also allows us to learn behavior-based tasks in a one-shot manner from a single demonstration.

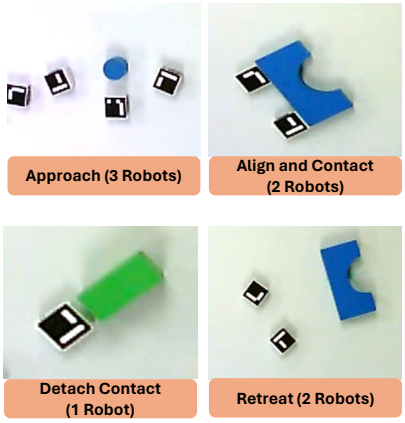


Fig. 3: Apriori skills are modeled as skills that each individual robot can perform. These are individual robot skills and do not constitute multi robot skills.

V. Experiments and Results

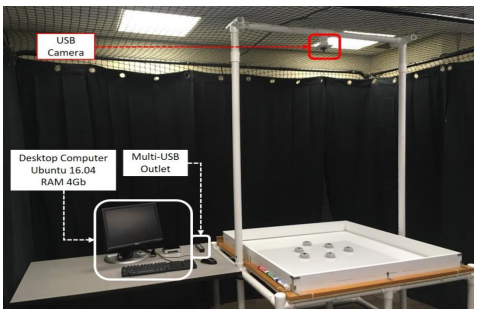


Fig. 4: Experimental Testbed shows the Hamster robots in the environment with a mounted overhead camera attached to the system.

Our experiments were carried out on a test bed featuring a table with an overhead camera aligned parallel to the surface of the table. The Hamster mobile robots [26] utilized for these experiments are equipped with dual-wheeled motors and IR proximity sensors, enabling precise navigation and interaction with their environment. This setup, depicted in Fig. 4, is designed to facilitate both real-time training and the execution of tasks. Owing to the framework's ability to execute behavior-based tasks from a single demonstration, it was imperative to evaluate performance under real-world conditions. Consequently, we ensured that RL training for contact-based tasks was also conducted in the real environment. This approach circumvents the simulation-to-reality gap, which can often hinder the transition of contact-based manipulation skills to practical application. By maintaining consistency in training and testing exclusively in a real-world

setting, we aimed to validate the framework's effectiveness in live scenarios.

We evaluate the performance of our learning from demonstration for multi-robot systems framework across various tasks, including a) Intruder Attack, b) Leader Follower, c) Object Transport, d) Object Rotate, and e) Object Color Sorting. The tasks (a) and (b) involve *behavior-based* objectives, while (c), (d) and (e) focus on *contact-based* objectives. The behavior-based tasks require robots to demonstrate specific behaviors, while the contact-based tasks involve rich manipulation, particularly in *Object Transport*, a well-studied area in multi-robot domains. Additionally, we implement a baseline method, a naive RL approach using the SAC algorithm. In this method, rewards are manually engineered according to each task's nature.

The principal metric for assessment was the success rate (SR), which represents the proportion of successful trials within a set of 30. Success was strictly defined by the robots' ability to fulfill the task's requirements, be it encircling an intruder with a defined boundary or executing accurate color-specific object sorting.

It is worth to note that we primarily focus on measuring the success or failure of task completion, employing metrics such as SR. This choice is influenced by the current state of Multi-Robot LfD approaches and the unique challenges they present. Present-day multi-robot LfD approaches often lack sophistication, and achieving reasonable skill goals is a significant accomplishment. Thus, assessing the success or failure of these tasks provides meaningful insights into the framework's capabilities. The absence of standardized evaluation parameters or approaches in LfD is due to the highly task-specific nature of these frameworks. Unlike more established fields, there is no one-size-fits-all approach or a baseline for comparison. Another challenge is the diversity of robot platforms used in LfD, each with varying physical structures, characteristics, and constraints. This diversity makes it challenging to conduct direct quantitative comparisons between different LfD approaches.

Given these factors, our emphasis on success or failure in task completion, along with the inclusion of a baseline RL method, provides a practical and relevant evaluation strategy for our framework.

A. Intruder Attack

The *Intruder Attack* task required a team of robots to encircle an intruder. Given only one expert demonstration, the robots had to adapt to variations in the intruder's features and positions. This behavior-oriented task leveraged the robots' apriori skills, specifically designed for approach and surround strategies. The reward function of the naive RL approach was predicated on the proximity of each robot to the intruder, aiming for a formation within 10 cm of the target. Success was evaluated based on the final positions of the robots in relation to the intruder and each other, the experiment achieving a success rate of 95% in a task with 3 robots and 92% with 5 robots when provided with just single demonstration, despite variations in environment and configurations.

B. Leader Follower

The *Leader Follower* task involved creating a follower formation behind a moving leader object, using just one expert demonstration as a reference. This behavior-oriented task utilized the robots' innate apriori skills to autonomously align in a sequential formation. The naive RL method calculated rewards based on the distance to maintain a coherent line behind the leader. The successful creation of this formation, as depicted in the accompanying figure, was achieved with a success rate 100% when provided with a single demonstration, which underscores the effectiveness of the behaviors demonstrated.

C. Object Transport

For the *Object Transport* task, pairs of robots were required to collaboratively move an object to a designated target location. The task complexity necessitated 50 demonstrations to effectively train the classifier for a nuanced reward signal. Employing a SAC architecture with a two-layer Multilayer Perceptron (MLP) network of 64 units each, the robots were trained over an 8-hour period. The SR, determined by the final proximity of the object to the goal, was recorded at 80%, demonstrating the robustness of the framework in facilitating cooperative transport.

D. Object Rotate

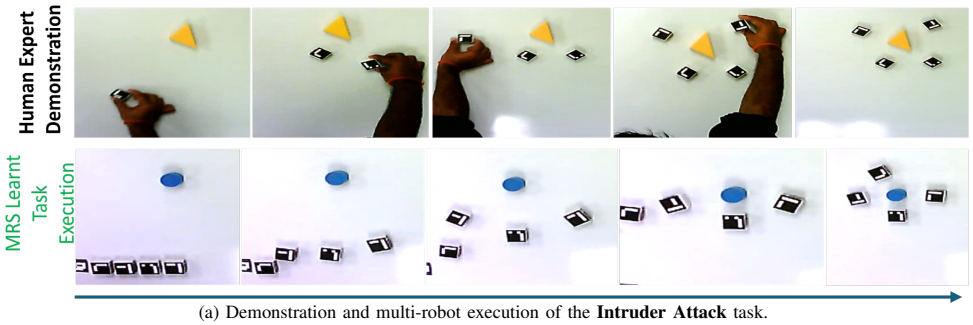
The *Object Rotate* task demanded precision as robots worked together to rotate an object by 180 degrees. To accommodate the required precision, the task also involved 50 demonstrations to refine the classifier's reward mechanism. After 6 hours of training with a SAC model similar to the *Object Transport* task, the robots accomplished a 67% SR, with a permissible margin of error of +/- 2 degrees, highlighting our framework's potential in tasks requiring exact maneuvers.

E. Object Color Sorting

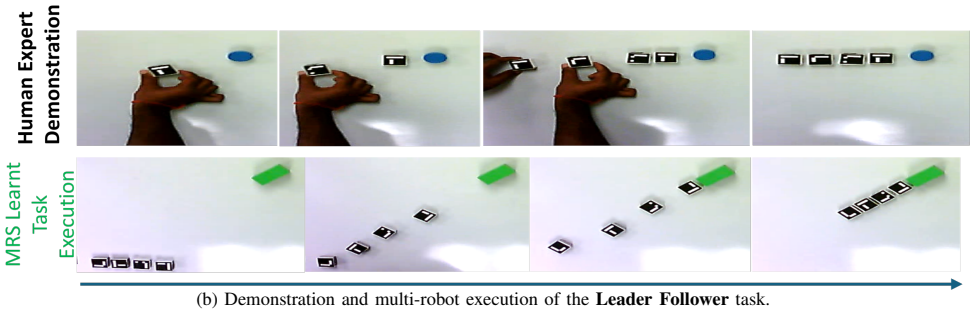
Building on the *Object Transport* mechanics, the *Object Color Sorting* task required the robots to sort objects by color into corresponding goals. No additional RL training was required, as the task extended previously learned transport skills. With a batch of 4 objects of varying colors introduced sequentially, the robots achieved a 77% SR, based on the accurate sorting of all objects. Errors were mainly attributed to the transport phase rather than the sorting, as robots consistently recognized the correct location of the goal.

The success rates, as shown in Table I, validate the effectiveness of our framework in both behavior and contact-based tasks. The high SR across different task types signifies our framework's competency in real-time execution and adaptability to task variations, especially when compared to traditional training intensive methods such as the baseline RL.

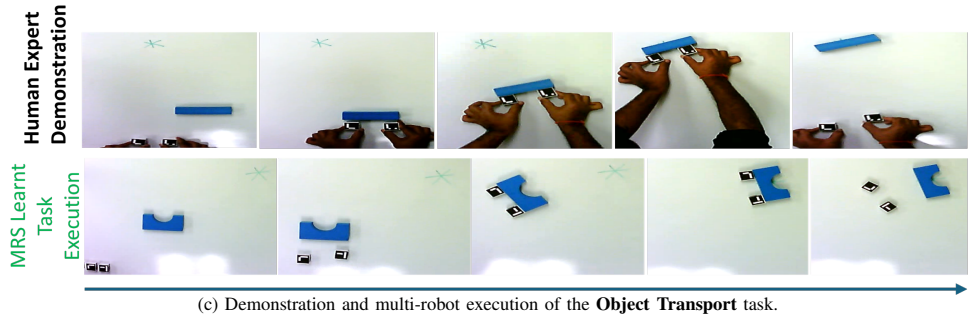
Furthermore, an ablation study on contact-based tasks, presented in Table II, examined the performance under three



(a) Demonstration and multi-robot execution of the **Intruder Attack** task.



(b) Demonstration and multi-robot execution of the **Leader Follower** task.



(c) Demonstration and multi-robot execution of the **Object Transport** task.

Fig. 5: Examples of demonstrations and task execution are presented. The objects used during demonstrations are different from the objects used during execution to showcase how our learning process is generalizable across different objects in the environment. For example, in (a), the yellow object was used for demonstration, while the blue object was used for execution. Experiment videos showing more examples can be found in the supplementary video.

TABLE I: Success rates for all tasks. Each task underwent 30 trials.

Task Type	Task	Proposed Methods (%)	Baseline (%)
Behaviour-based Tasks	Intruder Attack (3 Robots)	95	60
	Intruder Attack (5 Robots)	92	38
	Leader Follower (3 Robots)	100	65
	Leader Follower (4 Robots)	100	58
Contact-based Tasks	Object Transport	80	40
	Object Rotate	67	42
	Object Sorting	88	15

TABLE II: Success rates for all contact-based tasks with the skills learned under different reward conditions. Each task underwent 30 trials.

Task	Proposed Methods (%)	Only IK Rewards (%)	Only Classifier Rewards (%)
Object Transport	80	47	62
Object Rotate	67	24	48

conditions: using the full proposed method, only IK rewards, and only Classifier rewards. The complete framework consistently outperformed the other conditions, reinforcing the synergy between the Interaction Keypoints and the classifier-based rewards in our method. For object transport, the success rates dropped to 47% and 62% when relying solely on IK rewards and Classifier rewards, respectively. Object Rotate saw similar trends, with success rates of 24% for IK rewards and 48% for Classifier rewards, confirming the integral role of our hybrid reward strategy.

These results collectively emphasize our framework's efficiency and its ability to generalize across various MRS tasks with fewer demonstrations needed, paving the way for practical applications in dynamic environments.

VI. Discussion

Our investigation into the learning from demonstration framework for multi-robot systems has yielded promising results, revealing the framework's potential in a real-time, real-world setting. While the current version demonstrates a robust capability for managing tasks defined by discrete interaction keypoints, there is room to extend this proficiency to trajectory-based tasks, thereby broadening the scope of the framework's applicability.

One significant realization from our experiments is the value of real-time training and testing, which allowed us to bypass the sim2real gap often encountered in contact-based

tasks. However, scalability remains a key challenge, as the complexity of RL training grows with the number of robots, necessitating strategies for scalable, task-agnostic learning. Implementing our system in real-world scenarios like warehouse management is feasible with adjustments. Without an overhead camera, the system would rely on alternative localization and sensing technologies, such as LiDAR and onboard cameras, requiring enhanced onboard computing to process data in real time and reduce dependence on external resources. Future work could focus on developing more efficient algorithms and frameworks that support incremental learning as more robots are added, ensuring the system's adaptability across various environments.

Another exciting prospect is robot domain transfer, where a universal LfD framework can be applied across various robot types, simplifying MRS deployment. The adaptability of our method comes from using high-level visual demonstrations and IKs, which abstract task-specific details and enable generalization across mobile robotic platforms. These IKs focus on essential environmental interactions, reducing dependency on specific robot kinematics or dynamics. Additionally, the potential for heterogeneous multi-robot tasks, involving collaborations between different robot types like drones and ground robots, offers a promising area for further exploration of our framework.

Although this paper represents an initial step toward realizing a generalized LfD framework capable of handling a multitude of tasks in various environments, the journey ahead is expansive. Continuous research efforts are crucial to overcome existing limitations and to harness the full potential of MRS. The success we have demonstrated in a real-world context lays a strong foundation for future endeavors, aspiring towards a universally adaptable MRS framework.

VII. Conclusion

This paper presents an innovative Learning from Demonstration (LfD) framework for Multi-Robot Systems (MRS), leveraging visual demonstrations and a binary decision classifier to streamline skill acquisition and task execution. By reducing the need for extensive demonstrations, our approach addresses the challenge of data intensity in LfD research. Showing great promise for robust multi-robot learning, we plan to enhance our framework's scalability, support for heterogeneous robot teams, and trajectory-based skill inclusion. These advancements pave the way for a future of highly efficient and adaptable robot learning and collaboration, advancing autonomous systems.

References

[1] A. G. Billard, S. Calinon, and R. Dillmann, "Learning from humans," *Springer handbook of robotics*, pp. 1995–2014, 2016.

[2] M. V. Balakuntala, U. Kaur, X. Ma, J. Wachs, and R. M. Voyles, "Learning multimodal contact-rich skills from demonstrations without reward engineering," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 4679–4685.

[3] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Annual review of control, robotics, and autonomous systems*, vol. 3, pp. 297–330, 2020.

[4] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.

[5] B. Huang, M. Ye, S.-L. Lee, and G.-Z. Yang, "A vision-guided multi-robot cooperation framework for learning-by-demonstration and task reproduction," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 4797–4804.

[6] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on robotics*, vol. 21, no. 3, pp. 376–386, 2005.

[7] R. N. Darmanin and M. K. Bugeja, "A review on multi-robot systems categorised by application domain," in *2017 25th mediterranean conference on control and automation (MED)*. IEEE, 2017, pp. 701–706.

[8] L. Brunke, M. Greeff, A. W. Hall, Z. Yuan, S. Zhou, J. Panerati, and A. P. Schoellig, "Safe learning in robotics: From learning-based control to safe reinforcement learning," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 5, pp. 411–444, 2022.

[9] S. Chernova and M. Veloso, "Interactive policy learning through confidence-based autonomy," *Journal of Artificial Intelligence Research*, vol. 34, pp. 1–25, 2009.

[10] R. A. Knepper, T. Layton, J. Romanishin, and D. Rus, "Ikeabot: An autonomous multi-robot coordinated furniture assembly system," in *2013 IEEE International conference on robotics and automation*. IEEE, 2013, pp. 855–862.

[11] M. F. Martins and Y. Demiris, "Learning multirobot joint action plans from simultaneous task execution demonstrations." in *AAMAS*, 2010, pp. 931–938.

[12] K. Sullivan and S. Luke, "Hierarchical multi-robot learning from demonstration," in *Proceedings of the Robotics: Science and Systems Conference*, 2011.

[13] D. Freelan, D. Wicke, K. Sullivan, and S. Luke, "Towards rapid multi-robot learning from demonstration at the robocup competition," in *Robot Soccer World Cup*. Springer, 2014, pp. 369–382.

[14] M. A. Simões, R. M. da Silva, and T. Nogueira, "A dataset schema for cooperative learning from demonstration in multi-robot systems," *Journal of Intelligent & Robotic Systems*, vol. 99, no. 3, pp. 589–608, 2020.

[15] M. V. Balakuntala, V. L. Venkatesh, J. P. Bindu, R. M. Voyles, and J. Wachs, "Extending policy from one-shot learning through coaching," in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2019, pp. 1–7.

[16] S. S. Kannan, V. L. Venkatesh, R. K. Senthilkumaran, and B.-C. Min, "Upplied: Uav path planning for inspection through demonstration," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 1126–1133.

[17] A. Singh, L. Yang, K. Hartikainen, C. Finn, and S. Levine, "End-to-end robotic reinforcement learning without reward engineering," *arXiv preprint arXiv:1904.07854*, 2019.

[18] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the twenty-first international conference on Machine learning*, 2004, p. 1.

[19] A. Nair, B. McGrew, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Overcoming exploration in reinforcement learning with demonstrations," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 6292–6299.

[20] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[21] I. Lebedev, A. Erashov, and A. Shabanova, "Accurate autonomous uav landing using vision-based detection of aruco-marker," in *International Conference on Interactive Collaborative Robotics*. Springer, 2020, pp. 179–188.

[22] B. Charbuty and A. Abdulazeez, "Classification based on decision tree algorithm for machine learning," *Journal of Applied Science and Technology Trends*, vol. 2, no. 01, pp. 20–28, 2021.

[23] Y.-Y. Song and L. Ying, "Decision tree methods: applications for classification and prediction," *Shanghai archives of psychiatry*, vol. 27, no. 2, p. 130, 2015.

[24] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, *et al.*, "Soft actor-critic algorithms and applications," *arXiv preprint arXiv:1812.05905*, 2018.

[25] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, 2021.

[26] A. Lee, W. Jo, S. S. Kannan, and B.-C. Min, "Investigating the effect of deictic movements of a multi-robot," *International Journal of Human–Computer Interaction*, vol. 37, no. 3, pp. 197–210, 2021.