MULTIROBOT SYSTEMS

Cross-robot behavior adaptation through intention alignment

Xi Chen[1]*†, Yuan Gao[2,3]†, Hangxin Liu[1,4], Fangkai Yang[5], Ali Ghadirzadeh[6], Jun Yang[7], Bin Liang[7], Chongjie Zhang[8]*, Tin Lun Lam[2,3]*, Song-Chun Zhu[1,4,7]

Imitation learning (IL) has succeeded in enabling robots to perform new tasks by learning from demonstrations. However, its success is often constrained by the need for direct skill mappings between a learner and a demonstrator under identical conditions, limiting its adaptability to diverse environments and generalization across robots with different physical embodiments. To address these challenges, we introduce the Intention-Aligned Imitation Learning (IAIL) framework, a behavior adaptation approach that extends the conventional scope of IL by enabling robots to reproduce motions demonstrated by heterogeneous peers, even in previously unseen situations. Inspired by human cultural learning, IAIL aligns and adapts robot motions on the basis of high-level intentions annotated in natural language rather than by directly copying motor movements. This alignment is achieved by constructing a shared intention space that connects robot-generated motions with linguistic annotations, enabling inference-time behavior adaptation across diverse embodiments and environmental contexts. The framework further supports scalable task allocation in heterogeneous robot teams by leveraging differences in capabilities and constraints. We validated IAIL through real-world experiments involving seven distinct robots performing multistep collaboration tasks across 30 scenarios. Our results demonstrate that IAIL enables robust intention-aligned behavior adaptation across variations in embodiment, motion modality, and task configuration. These capabilities enable flexible behavior transfer across heterogeneous robots and support resilient, autonomous multirobot systems for reliable real-world collaboration.

INTRODUCTION

Deploying a team of robots to perform long-horizon tasks in dynamic real-world environments holds substantial potential to enhance productivity in manufacturing, increase system resilience in adversarial conditions, and enable collaborative behaviors beyond the capabilities of individual robots. However, programming a robot team with proficient individual skills and efficient task allocation could be difficult, particularly when the team varies substantially. Imitation learning (IL) has emerged as a promising method for enabling robots to efficiently acquire new skills by learning from expert demonstrations (*1*–*3*), thereby facilitating skill transfer for robotic systems. However, most IL approaches necessitate that the learner and demonstrator operate under identical task conditions and have precise mappings between their motor actions, limiting the adaptability of the tasks demonstrated in different environments and the generalizability in learning from robots with different physical embodiments.

The primary objective of IL is to establish meaningful motion correspondence between the actions of the demonstrator and the learner, thereby ensuring comparable outcomes after execution. However, achieving this correspondence becomes increasingly challenging in the presence of variations in environmental conditions and robotic embodiments (*4*–*6*). A comparison of IL settings with variations between the demonstrator and learner is illustrated in Fig. 1. For differences in physical embodiments, such as robots with different degrees of freedom, body structures, or actuator types, motion correspondence can be established on the basis of invariant body components (*7*, *8*) or transitions in environmental states (*9*, *10*). For environmental variations, such as different lighting conditions, varied camera views, or interactable objects with changing colors or textures, motion correspondence can be established by finding a common feature space with domain confusion approaches (*11*, *12*). Although these strategies have shown success under moderate variations, they become inadequate in cases involving substantial differences, such as between robots with fundamentally different motion modalities (ground vehicles versus drones) or environments with distinct sets of interactable objects (office versus home). In such settings, direct correspondence becomes unreliable because of mismatched robot capabilities and functionalities. To address this issue, recent works have suggested learning correspondence based on completed tasks or the final motion outcomes (*13*–*16*). However, these methods require datasets with manually labeled or paired motion trajectories for each learner-demonstrator combination. This pairing process demands extensive engineering effort, which limits the scalability of these methods to IL scenarios involving an increased number of robots. Alternatively, some methods leverage unsupervised learning techniques to learn the correspondence (*17*–*19*), which eliminates the engineering effort but requires robots with identical functionalities. An efficient approach that can be adapted to various environmental conditions and generalized to diverse robot forms is still lacking.

Moving beyond the agent-to-agent imitation setup, enabling a team of robots to replicate the multistep collaborative tasks performed by another team introduces a team-to-team imitation setup. When considering heterogeneity in team size, robot types, and individual robot capabilities, the critical challenge lies in enabling the learned model to generate feasible motion plans for the learner robot team, which may differ from the demonstrations, and in correctly assigning each

[1]State Key Laboratory of General Artificial Intelligence, Beijing Institute for General Artificial Intelligence (BIGAI), Beijing, China. [2]School of Science and Engineering, Chinese University of Hong Kong, Shenzhen, China. [3]Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen, China. [4]School of Intelligence Science and Technology, Peking University, Beijing, China. [5]Microsoft, Beijing, China. [6]Embark Studios, Stockholm, Sweden. [7]Department of Automation, Tsinghua University, Beijing, China. [8]McKelvey School of Engineering, Washington University in St. Louis, St. Louis, MO, USA.
*Corresponding author. Email: pcchenxi@gmail.com (X.C.); chongjie@wustl.edu (C.Z.); tllam@cuhk.edu.cn (T.L.L.)
†These authors contributed equally to this work.

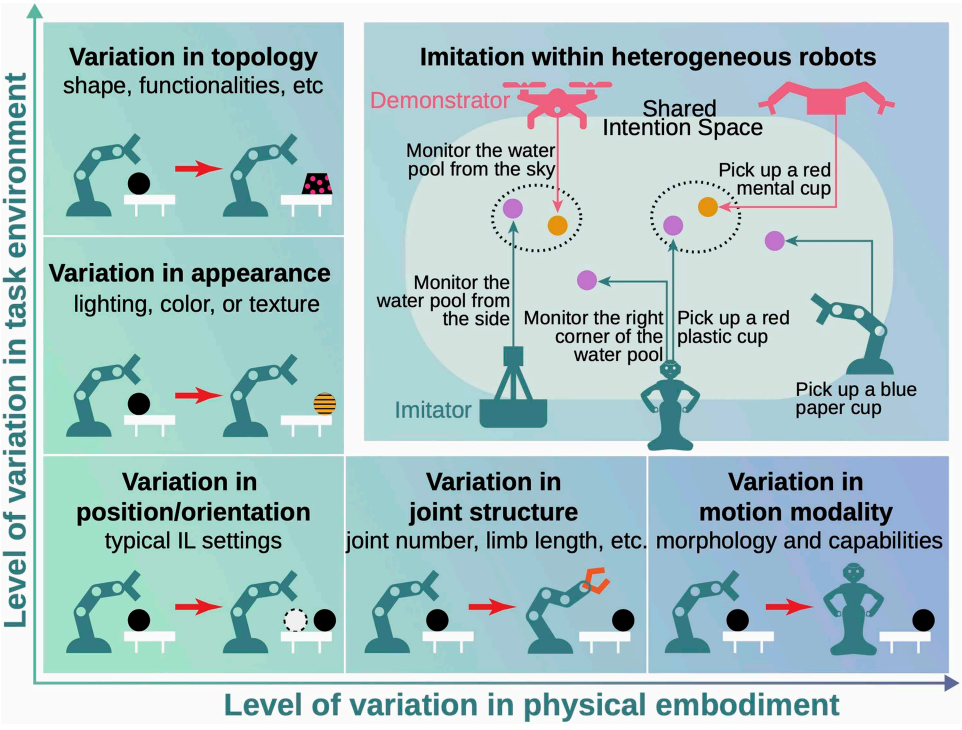

Fig. 1. A visual illustration of different IL scenarios. In prior works, IL typically occurred between two robots with varying degrees of differences in physical embodiments or task-specific environmental conditions. In contrast, our work explores a setting where heterogeneous robot teams, characterized by substantial differences in both embodiments and environmental conditions, can imitate each other through motion demonstrations. This imitation is achieved by associating motions on the basis of their underlying intentions, which are defined using human-annotated language descriptions. The association is performed in a shared intention space, constructed by aligning the embeddings of motion sequences with their corresponding language intention annotations. Each motion of the demonstrator robot is matched to the motion of the learner robot that has the smallest distance in the intention space. The dashed circles in the figure indicate successful motion associations in the intention space.

motion step to the appropriate learner robot on the basis of its capabilities. This necessitates an integration of IL for multirobot teams (*20*–*23*) and multirobot task allocation (*24*, *25*). However, owing to the varied functionalities of the learner and demonstrator robots and the unclear task definitions represented by implicit motion trajectories, imitation between heterogeneous robot teams remains an unexplored topic.

To transcend these limitations and enable imitation across robot embodiments and task conditions, we propose aligning robot behaviors on the basis of shared high-level intentions. Rather than replicating low-level motor actions or embodiment-specific features, our approach compares and associates behaviors according to their underlying task objectives, supporting both agent-to-agent and team-to-team transfer. We define an intention as the goal or outcome of a motion and represent it using a human-annotated language description that abstracts away control and embodiment details. We then construct a shared intention space by aligning motion embeddings with their corresponding annotations, enabling intention similarity to be measured across heterogeneous robots. At inference time, a learner retrieves the most demonstration-relevant behavior from its pretrained repertoire by matching intentions. This formulation extends naturally to team-level settings: Given multiple learners, the system assigns each demonstrated step to the robot that can most feasibly realize the intended outcome in its repertoire, enabling capability-aware task allocation across varying team compositions. We refer to this framework as Intention-Aligned Imitation Learning (IAIL).

This approach draws inspiration from human cultural learning mechanisms, where individuals understand others' actions in terms of intentions and reexpress them through contextually suitable behavior. This aligns with the cognitive science literature on rational imitation, which suggests that learners, even infants, prioritize reproducing a demonstrator's inferred goals over their exact movement patterns (*26*, *27*). Studies in neuroscience further support this view, indicating that humans interpret behavior at an intentional level rather than via motor mimicry (*28*–*35*).

Here, we study imitation across a diverse set of robots with distinct embodiments, operating environments, and capabilities, where any individual robot or team of robots may act as a demonstrator or a learner. We evaluate our framework on seven real-world robots, including boats, drones, mobile robots, and robotic arms, performing a compound task spanning multiple phases. The task is demonstrated by a team of three robots and executed by a separate team of three or four robots, with no overlap between demonstrators and learners. Across 30 scenarios with varying task and team configurations, our framework achieves robust task completion across diverse embodiments and motion modalities. These results demonstrate generalizability across robots, adaptability to varying tasks and environmental conditions, and scalability to heterogeneous teams with different compositions. We further compare IAIL against two baseline methods in simulation, quantifying consistent gains under multiple sources of variation. Together, these results highlight the potential of intention-aligned imitation to enable more flexible and efficient learning in multirobot systems and to extend IL to dynamic real-world deployments.

RESULTS

We evaluated our IAIL framework through experiments involving heterogeneous robots performing IL tasks in dynamic environments. The framework associates behaviors through underlying intentions rather than direct motor correspondence. This design enables adaptation across embodiments and environmental contexts and preserves the objectives of the demonstrations. This section presents an overview of our framework, followed by the experimental validation, detailed results, and a comparative analysis of our approach with baseline methods.

IAIL framework overview

Figure 2 provides an overview of the IAIL framework in an agent-to-agent scenario, illustrating the core modules, the three main stages of the imitation process, and how these modules interact across stages. The figure is intended as a high-level guide to the system's structure; detailed descriptions of each module and stage are provided in Materials and Methods.

The framework enables imitation among heterogeneous robots by associating actions on the basis of shared intentions. The process consists of three key stages: context-aware motion generation, motion intention extraction, and motion association based on intention similarity. Below, we describe each stage, identifying the networks involved, their training, and their role in the overall process. A video demonstration of the three stages is included in movie S2.

Context-aware motion generation

In this stage (Fig. 2A), we generate feasible actions for the learner robot i on the basis of its current state, defined as the information perceived by its onboard sensors, which captures its operational context in the environment. Each action is a single command or a sequence of commands that the learner can execute to achieve a specific goal in its environment. The set of generated actions reflects the current capabilities or skills of the learner robot in its present context and serves as candidates for intention-based association in later stages. These actions are sampled from a generative model p_{θ_i}, trained offline using a dataset collected from robot i.

Motion intention extraction

In this stage (Fig. 2B, i and ii), both the learner's generated actions and the demonstrator's executed actions are projected into a shared motion-intention embedding space. The projection is performed using a motion encoder f_{ψ_i} for the learner and f_{ψ_j} for the demonstrator, each trained jointly with a shared annotation encoder f_ξ. Training follows a contrastive learning objective in which human-annotated language descriptions of motions provide semantic supervision. This ensures that actions with similar annotated intentions, regardless of embodiment, are embedded close to each other in the space. Through this process, the system obtains an intention-level representation for both the demonstration and each candidate action, enabling embodiment-independent comparison.

Motion association based on intention similarity

In the final stage (Fig. 2C), we compare the embedded representation of the demonstrator's motion with the embeddings of the learner's candidate actions generated in the first stage. The system selects the candidate whose embedding is closest to the demonstrator's embedding in intention space, ensuring that the chosen action is executable for the learner and aligned with the demonstrated objective. When multiple learner robots are available, we perform this selection over all candidates from all learners and assign the best-matching action to the robot that generated it.

Team-to-team imitation setup

We evaluated the IAIL framework in a complex real-world environment involving seven heterogeneous robots performing multistep tasks. The robots had distinct capabilities and operated in a shared environment comprising several areas, each associated with different types of tasks. The tasks included the following: monitoring user activities at one of four locations (M_1 to M_4), fetching items at one of three areas (I_1 to I_3), and delivering items to one of two destinations (D_1 and D_2). A visual illustration of the robots, environment, and related tasks is shown in Fig. 3A.

Seven robots were involved in the study and were divided into two teams. The demonstrator team consisted of Tello (*36*), Dual-Arm (*37*), and Spark (*38*), whereas the learner team consisted of Cuboat

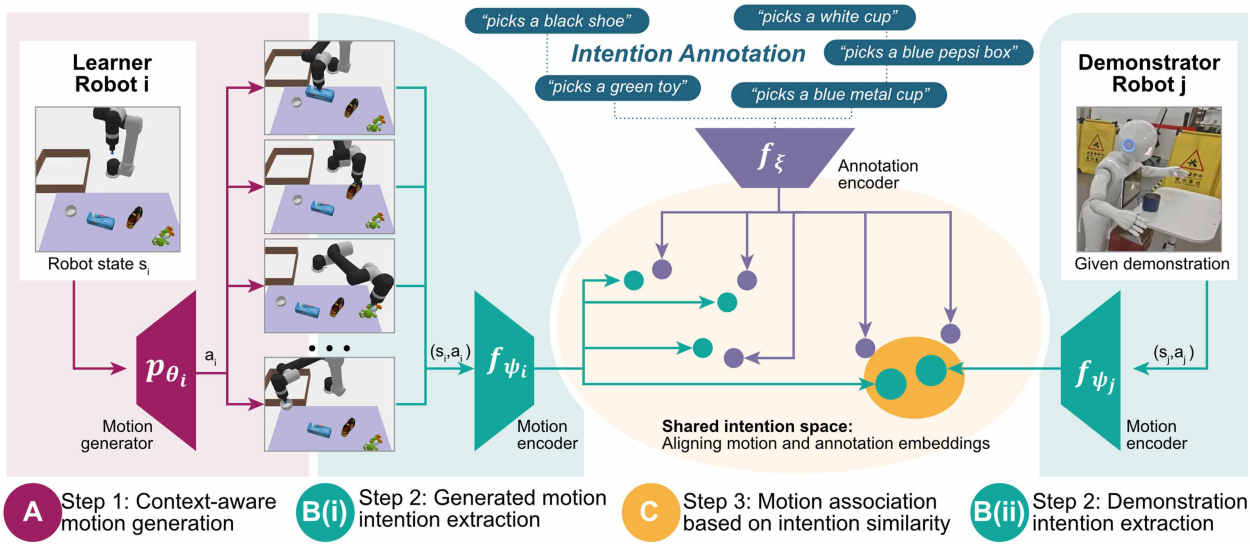

Fig. 2. Overview of the IAIL framework in an agent-to-agent scenario. The diagram illustrates the process of learner robot i reproducing an action demonstrated by robot j. The process involves three main stages: (**A**) A batch of executable actions, each capable of achieving various goals, is sampled from the pretrained motion generator p_{θ_i} of robot i. (**B**) (i) The intentions of the sampled motions are extracted by projecting them into a shared embedding space via the motion encoder f_{ψ_i} of robot i. (ii) In parallel, the intention of the demonstrator's motion is extracted using the motion encoder f_{ψ_j} of robot j. The shared embedding space is regularized by the annotation encoder f_ξ, trained jointly with the motion encoders to align motion embeddings with their high-level intentions. (**C**) The demonstration is then associated with the sampled learner motion whose embedding is closest in the shared intention space. Through this generation-encoding-association process, the learner interprets the demonstration and adapts it into an executable action aligned with the demonstrated objective.

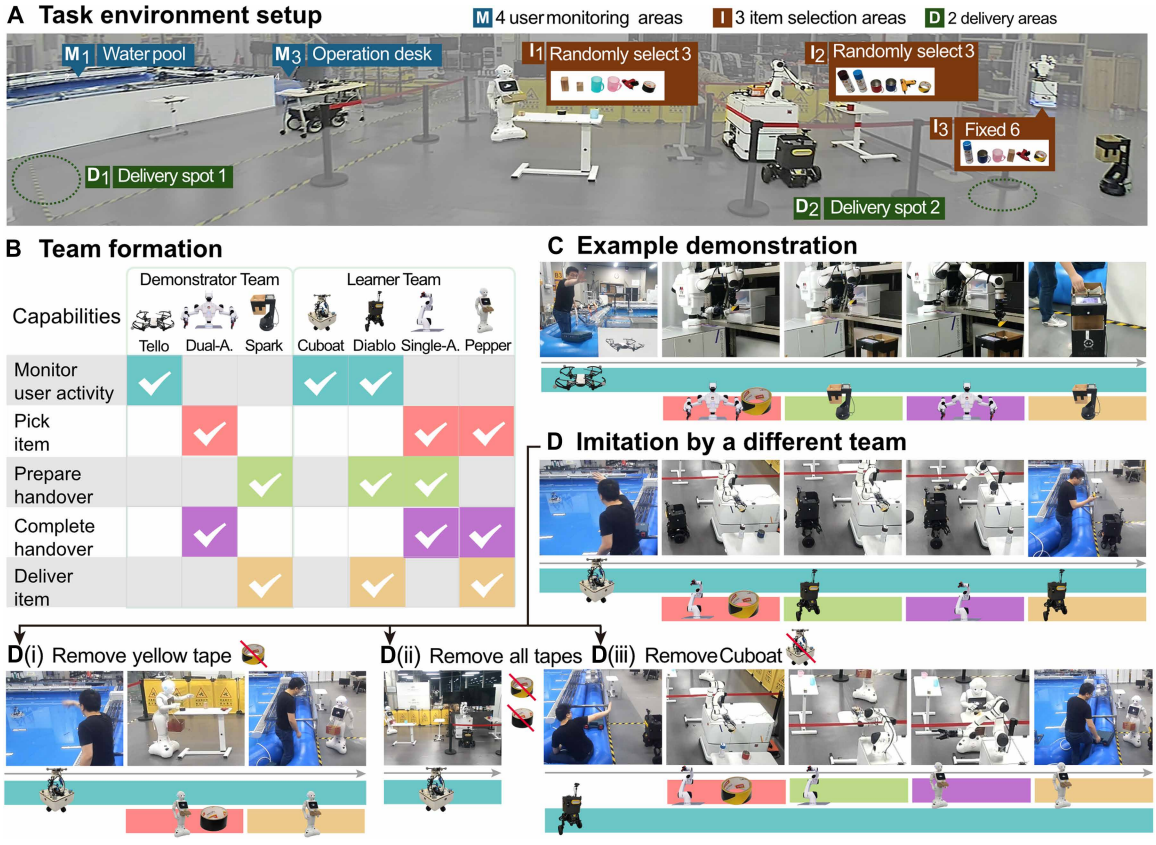

Fig. 3. The real-world experiment setup and imitation results obtained under various conditions. (**A**) The imitation task has three phases: monitoring user activity in M_1 to M_4, fetching an item from I_1 to I_3, and delivering it to D_1 or D_2. (**B**) Given that the robots on the learner team are different from those on the demonstrator team, direct imitation is not possible. (**C**) An example five-step demonstration performed by Tello, Dual-Arm, and Spark: Tello monitors the user signal, Dual-Arm picks a yellow tape, Spark moves closer to initiate an item handover, Single-Arm passes the item to complete the handover, and Spark delivers the item. (**D**) The imitation performed by the learner team. Our IAIL framework successfully allocated the imitative actions to the corresponding robots on the basis of their capabilities. (i) When the yellow tape was removed, the black tape was identified as an alternative, and the Pepper robot was assigned to pick up and deliver it. When neither the yellow tape nor the black tape was available, the team remained idle, because no imitative actions that could fulfill the intention were possible as shown in (ii). In (iii), Diablo was selected to monitor the user when Cuboat was removed from the team, which subsequently changed the imitative actions of the other robots. In the various cases, the executed performance differed substantially from the demonstration.

(*39*), Single-Arm (*37*), Pepper (*40*), and Diablo (*41*) (see Fig. 3B). Because of embodiment and workspace constraints, each robot had access to a subset of areas and supported distinct capabilities. Tello was a drone that could monitor all four locations M_1 to M_4. Dual-Arm had two manipulators and could retrieve items from the drawer at I_3, whereas Single-Arm could retrieve items placed on the table at I_2. Spark was a mobile transporter that could deliver items to D_1 and D_2. Cuboat operated in the water pool and could monitor locations M_1 and M_2. Diablo was a height-adjustable mobile robot that could monitor all four locations M_1 to M_4 and deliver items to D_1 and D_2. Pepper was a mobile manipulator that could access items on the table at I_1 and also deliver items to D_1 and D_2. The detailed robot and environmental setup is provided in the Supplementary Materials ("Real-world robot sensor and motor control" and "Real-world task and environment setup" sections).

During the IL process, the demonstrator team provided a demonstration consisting of five actions. First, Tello flew to one of the four monitoring locations (M_1 to M_4) to monitor user activities. The user then entered the area and signaled the robot by waving at its camera. Once the signal was detected, Dual-Arm picked an item from the drawer at I_3 and passed it to Spark for delivery. This handover process involved two actions: Spark initiated the handover by moving closer to Dual-Arm, and then Dual-Arm placed the picked item into Spark's basket to complete the handover. Last, Spark delivered the item to one of the delivery locations (D_1 or D_2) to complete the task. An example demonstration given in Fig. 3C shows that when the user detected pool leakage and sent signals, the team of robots monitored this activity at M_1, fetched a yellow tape, and delivered the tape to D_1.

After the five-step demonstration, the learner team was required to replicate the final outcome: delivering the correct item to the appropriate location. However, the user's location, the item type, and the delivery destination were not explicitly provided, requiring the team to infer this information from the demonstration.

Team-to-team imitation results

To assess the effectiveness of the proposed IAIL framework, we evaluated performance from two perspectives. First, we measured task success rate and adaptation accuracy, which reflected how well the learner robots imitated the demonstrated behaviors and achieved

the intended outcomes. Second, we analyzed how tasks were executed across varying environmental configurations and team compositions, highlighting the system's flexibility, robustness, and adaptability under dynamic conditions. We began by introducing the evaluation scenarios, followed by detailed results under these two perspectives.

Evaluation scenarios

To assess the system's ability to handle variations in environmental conditions and team formations, we conducted 30 scenarios with varying task configurations for both the demonstration and the learner robot team. In each scenario, the demonstration involved a unique user location, selected from M_1 to M_4; an item, chosen from the six items in I_3; and a delivery position, selected from D_1 or D_2. For the learner robots, the three items at I_1 and I_2 were randomized from their respective lists in each scenario. In addition, in one-third of the scenarios, one robot was randomly removed from the learner team, leaving only three robots to complete the tasks. The variations between the training and evaluation scenarios are provided in the Supplementary Materials ("Training and evaluation data variations" section).

Because of these random variations in robot availability and item distribution, 6 of the 30 scenarios were completion infeasible, either because no robot had the necessary capabilities or because the relevant item from the demonstration was entirely unavailable. Among the 24 feasible scenarios, 11 included the exact demonstrated item (same item available), whereas the remaining 13 included a functionally equivalent item but with a different shape, color, or specific attribute from the same semantic class (same-class item available). The item class is shown in fig. S3. To ensure the reliability and statistical significance of our results, we repeated the evaluation three times on the same 30 scenarios using models trained with different random seeds.

Task success rate

A task was considered successful when the exact demonstrated item, or an item in the same class, was delivered to the correct location. The task success rate was evaluated as the proportion of successful trials out of the total number of scenarios that could be completed successfully. This metric captured the overall effectiveness of IAIL in consistently imitating behaviors across teams of heterogeneous robots.

As shown in Table 1, the robots successfully completed an average of 22 of 24 scenarios, achieving an overall success rate of 0.92. This high success rate demonstrated the system's ability to effectively enable imitation between heterogeneous robot teams, allowing them to execute multistep tasks through action demonstrations.

There were seven failures among the 30 scenarios: One involved the robot picking an irrelevant item, whereas the other six remained inactive even when the exact item or an item of the same class was presented. These failures occurred primarily because of incorrect extraction of demonstration intentions, which could result from factors such as model errors or sensory noise. When this occurred, the intention distances between the demonstration and the sampled actions became large, and the sampled actions were identified as incapable of replicating the demonstration. In such scenarios, the robot opted to remain inactive rather than risk performing an action that could lead to an unexpected or undesirable state. This conservative behavior was critical in real-world applications where safety was paramount.

Adaptation accuracy

In addition to the task success rate, we evaluated the robots' ability to adjust their behavior under different environmental conditions using the adaptation accuracy metric. This metric measured the percentage of instances in which the robots achieved the best possible outcome under three conditions. If the demonstrated object was available, the learner robots were expected to deliver that object. If it was unavailable but a same-class object was available, the system was expected to deliver the available object as a substitute. If neither the demonstrated object nor a same-class object was available, the robots were expected to recognize that the task was infeasible and remain inactive. The items used in the experiment and their corresponding classes are shown in fig. S3.

As shown in Table 1, across all 30 evaluation scenarios, the learner team achieved an overall best-adaptation accuracy of 88% under three conditions. Specifically, in the 11 scenarios where the exact demonstrated item was available, the robots successfully delivered the exact item an average of 9.33 times. This result highlighted the system's precision in replicating demonstrated actions under ideal conditions. In the 13 scenarios where the exact item was unavailable but an alternative item in the same class was present, the system correctly identified and delivered the substitute item an average of 11.33 times. This demonstrated the framework's adaptability in selecting appropriate alternatives, effectively handling less-than-ideal conditions while still achieving task goals. Last, in the six scenarios where no relevant items were available, the robots correctly recognized the task's infeasibility and remained inactive an average of 5.67 times. This underscored the system's ability to skip actions when necessary, preventing errors and avoiding risky operations.

Figure 3D illustrates the team's imitative behaviors that best adapted to these three conditions. First, the learner team successfully selected and delivered the yellow tape as demonstrated, despite the presence of another item in the same class (black tape). In this scenario, the proposed IAIL framework allocated tasks to learner robots with functionalities most similar to those of the demonstrator team: Cuboat versus Tello, Single-Arm versus Dual-Arm, and Diablo versus Spark.

Table 1. Average performance over 30 diverse evaluation scenarios ± 1 SD. Each value reports the average number of successful or best-adapted scenarios out of the total number of applicable cases. The evaluations were repeated three times with different random seeds.

	Number of scenarios	Mean achieved trials	Success rate
Task success	24	22 ± 1.00	92%
Best adaptation	30	26.33 ± 1.15	88%
Same item available	11	9.33 ± 0.58	85%
Same-class item available	13	11.33 ± 1.15	87%
Completion infeasible	6	5.67 ± 0.58	94%

This indicated that the framework effectively enables team-to-team imitation. When the yellow tape from the demonstration was removed, the black tape was identified as a substitute. However, this item was out of reach for the Single-Arm robot. As a result, Pepper, which has both manipulation and navigation capabilities, was assigned to fetch and deliver the item (see Fig. 3D, i). In this case, the IAIL framework not only adapted to changes in environmental conditions but also ensured efficient task allocation. When the black tape was also removed (Fig. 3D, ii), no relevant items were available. The imitation was considered successful because the robots remained inactive; performing any action in this scenario would lead to incorrect or unintended outcomes. The IAIL framework also demonstrated robustness to variations in the composition of the learner team. For example, when Cuboat was absent (Fig. 3D, iii), Diablo substituted its role for monitoring, leaving Pepper to receive and deliver the tape. This emergent task allocation highlighted the strong generalizability of the proposed IAIL framework and its ability to bridge the skill sets of heterogeneous robots using shared intentions. The videos of the IL results shown in Fig. 3D (i, ii, and iii) are provided in movie S1.

Flexibility in robot role assignments

To further assess the flexibility and adaptability of our framework, we analyzed how tasks were distributed across different robots in the learner team. Table 2 presents the proportion of scenarios in which each robot was assigned to a given task phase (user monitoring, item picking, and item delivery), as well as the percentage of task assignments that fell within each robot's functional capabilities, confirming the validity of the assigned roles.

This analysis provided two key insights. First, across all scenarios, assigned tasks were strictly within each robot's physical and functional capabilities, resulting in a 100% feasibility rate. For example, only mobile robots were tasked with navigation and delivery, whereas fixed-base manipulators were assigned only object-picking roles. This confirmed that the system respected embodiment constraints and avoided unsafe or infeasible actions. Second, the distribution of roles varied with environmental configuration and team composition. For example, Cuboat and Diablo were both assigned to monitoring in different scenarios (38 and 62%, respectively), and item delivery was handled by Diablo (29%) and Pepper (71%). In 21% of scenarios, Pepper delivered items handed over by Single-Arm, typically when Diablo was unavailable, indicating that the system adapted task assignments in response to agent availability and context. Although the assignments remained feasible, the resulting role allocation was not hard-coded and instead reflected context-sensitive decision making.

Together, these results indicated that the IAIL framework enabled flexible and adaptive role assignments that respected robot capabilities and responded to team composition. This context-aware allocation strategy demonstrated the system's ability to generalize across embodiments and conditions, which was essential for scalable deployment in dynamic, heterogeneous robot systems.

Quantitative evaluation of the latent intention space

To further assess the internal structure and robustness of the shared intention space learned by IAIL, we conducted a quantitative analysis of the latent embedding distributions across tasks and robot embodiments. We measured semantic separation between task types using intraclass and interclass cosine distances in the latent space. We also assessed embodiment invariance by computing cross-embodiment alignment errors.

For this analysis, we randomly sampled three unseen trajectories per robot per task, yielding a total of 120 test samples. These samples were not used during training and spanned the same task set evaluated in the real-world experiments. To provide an intuitive understanding of the latent intention space, we applied *t*-distributed stochastic neighbor embedding to project the high-dimensional intention embeddings into a two-dimensional space. The resulting visualization is shown in fig. S5.

Semantic separation across tasks

We computed pairwise cosine distances among latent embeddings corresponding to five high-level task phases: monitoring, picking, prepare_handover, complete_handover, and delivery. Intraclass distance was defined as the average cosine distance between embeddings of the same task type, whereas interclass distance was computed across different task types. To summarize overall task separability, we report the semantic separation ratio: the mean interclass distance divided by the mean intraclass distance, a form commonly used in unsupervised clustering evaluation (*42*, *43*).

For the picking phase, we also computed intraclass distances at two levels of semantic granularity by treating each distinct object as a separate task ("same item") and by grouping objects into broader semantic categories ("same-class item"), such as grouping all cup variants into a single category. The item instances and classes are shown in fig. S3. The latter reflected the grouping used for success rate computation in Table 1 and was used for calculating the overall separation ratio.

Table 2. Distribution of task assignments across the learner robot team. Each cell indicates the proportion of scenarios in which the robot was assigned the specified task. The last column shows the percentage of assigned tasks that were within the robot's capability scope. Percentages are computed from evaluation scenarios in which the task was completed successfully. Dash entries indicate that the robot was not assigned to perform the task.

Robot	Monitor user activity	Pick item	Deliver item	Rate of assigning feasible tasks
Cuboat	38%	–	–	100%
Diablo	62%	–	29%	100%
Single-Arm	–	50%	–	100%
Pepper	–	50%	50% + 21%*	100%

*For Pepper, 50% of deliveries were of items it picked itself; 21% were handovers from Single-Arm, used when Diablo was unavailable for delivery.

As shown in Table 3, the global interclass distance was high (0.997 ± 0.003), indicating that task-specific latent representations were well separated and nearly orthogonal. In contrast, intraclass distances were substantially lower across all task phases (Tables 4 and 5). For instance, the monitoring tasks exhibited intraclass distances ranging from 0.276 to 0.375, whereas picking tasks ranged from 0.23 to 0.499, reflecting variability because of differences in object texture and geometry in each class. The handover phases showed distances from 0.226 to 0.352, and delivery tasks demonstrated the tightest clustering, with distances of 0.023. These values collectively yielded an average separation ratio of 3.764, confirming that IAIL's intention space formed compact, task-specific clusters. Even for picking cups, which involved four visually dissimilar cup types, the embeddings formed relatively tight latent clusters (up to 0.499 ± 0.034), indicating that high-level intent is preserved despite appearance variation. This well-structured latent space directly supported IAIL's ability to reliably associate intentions and retrieve appropriate policies, as reflected in its overall task success rate of 92% in Table 1.

Fine-grained object structure

When comparing intraclass distances between same-item and same-class item groupings in the picking phase, we observed that same-item distances were consistently much lower. For example, the intraclass distance for picking a specific cup was 0.119 ± 0.044, compared with 0.499 ± 0.034 when considering the class to consist of four types of cups. This suggested that the latent space preserved fine-grained semantic distinctions and formed tighter clusters around specific object instances.

This structure was functionally important, because it enabled IAIL to preferentially select the same item shown in the demonstrations (rather than same-class ones) whenever they were available during latent similarity–based motion association. As shown in Table 1, this preference translated into high adaptation accuracy, achieving 85% success adaptation when the same item was available and 87% when only same-class items were accessible. These results confirmed that the intention space encoded discriminative features at both coarse (task-type) and fine (object-instance) levels, which was essential for robust and flexible motion imitation.

Cross-embodiment alignment

To evaluate the embodiment invariance of the latent intention space, we defined the cross-embodiment alignment error, in analogy to the intraclass distance used in classic cluster validation metrics. Specifically, we computed the mean cosine distance among the embedding centroids of different robots performing the same task. This metric quantified how tightly intention representations from heterogeneous embodiments clustered in each task category, with lower values indicating stronger alignment across robots.

As shown in Tables 3 to 5, monitoring tasks exhibited alignment errors ranging from 0.460 to 0.499, approximately half the interclass distance, indicating consistent intention encoding despite substantial differences in robot morphology. The picking phase also demonstrated robust cross-embodiment consistency (errors from 0.162 to 0.423), even across objects with distinct shapes and grasp strategies. The delivery phase achieved the lowest alignment errors (0.030 to 0.031), indicating nearly identical latent encoding across robots. To summarize embodiment invariance across all tasks, we computed the embodiment alignment ratio, defined analogously to the separation ratio as the mean interclass distance divided by the mean cross-embodiment alignment error. The resulting ratio of 3.046 confirmed that robots with widely varying embodiments consistently encoded shared task intentions in a unified latent space.

Furthermore, consistent with the trend observed in semantic separation, we found that cross-embodiment alignment errors were smaller when comparing robots picking the same specific item than when aggregating items in the same semantic class, as demonstrated in Table 5. This further reinforced the latent space's capacity to preserve object-level specificity across embodiments.

This cross-embodiment alignment was functionally critical for enabling IAIL's flexible role allocation capabilities, as demonstrated in Table 2. In scenarios where a robot became unavailable or suboptimal, other robots could step in by evaluating the similarity of their candidate motions in the latent intention space. This shared representation enabled dynamic task redistribution based on individual capabilities and availability, with minimal coordination overhead.

Comparisons with baseline methods

Recent advances in cross-embodiment transfer have explored adaptation between individual robots (*7*, *18*, *44*, *45*). However, these approaches predominantly assume a fixed one-to-one correspondence between a demonstrator and a learner, and they do not directly address capability-aware role assignment and validity guarantees required for imitation in heterogeneous robot teams. To contextualize our contributions within existing frameworks, we evaluated our method in a simplified one-to-one agent setting, where the goal was to generate valid motions that achieved the same intended outcome as the demonstration. We benchmarked our approach against two representative paradigms in agent-to-agent imitation: density-based mapping and description-based translation.

Baseline methods

We compared IAIL against two representative baselines, a density-based approach adapted from (*18*) and a description-based approach that used natural language as an intermediate representation (*46*–*52*). For the density-based approach, the baseline learned unsupervised correspondences between robot behaviors by aligning the distributional divergence of skills extracted from motor trajectories, where skills are defined as temporally extended behaviors. The underlying assumption was that different robots share similar skill structures for accomplishing comparable tasks, even in the presence of substantial morphological differences. To align skill distributions

Table 3. The overall organization and generalizability of the latent space. The global interclass distance and the resulting semantic separation and embodiment alignment ratios are reported.

Global interclass distance	0.997 ± 0.003
Semantic separation ratio	3.764
Embodiment alignment ratio	3.046

Table 4. Quantitative analysis of latent intention space structure. Each row reports the mean ± SD of cosine distances for intraclass clustering and cross-embodiment alignment across tasks. The SD was computed using models trained with three different random seeds. Lower values indicate tighter clustering and stronger embodiment invariance.

Task type	Intraclass distance	Cross-embodiment error
	Monitoring	
M_1	0.375 ± 0.006	0.499 ± 0.008
M_2	0.356 ± 0.002	0.475 ± 0.002
M_3	0.290 ± 0.007	0.484 ± 0.012
M_4	0.276 ± 0.003	0.460 ± 0.005
Prepare_handover	0.352 ± 0.007	0.469 ± 0.009
Complete_handover	0.226 ± 0.006	0.301 ± 0.008
	Delivery	
D_1	0.023 ± 0.003	0.031 ± 0.004
D_2	0.023 ± 0.002	0.030 ± 0.002

Table 5. Quantitative analysis of latent intention space structure for picking up items. Each row reports the mean ± SD of cosine distances for intraclass clustering and cross-embodiment alignment. The SD was computed using models trained with three different random seeds. Lower values indicate tighter clustering and stronger embodiment invariance.

	Intraclass distance		Cross-embodiment error	
	Same item	Same-class item	Same item	Same-class item
Wood block	0.018 ± 0.003	0.230 ± 0.014	0.055 ± 0.011	0.162 ± 0.021
Cup	0.110 ± 0.044	0.499 ± 0.034	0.297 ± 0.030	0.423 ± 0.010
Glue gun	0.022 ± 0.01	0.280 ± 0.005	0.051 ± 0.005	0.373 ± 0.007
Tape	0.019 ± 0.003	0.235 ± 0.027	0.061 ± 0.009	0.358 ± 0.005
Paint	0.120 ± 0.128	0.271 ± 0.006	0.103 ± 0.076	0.180 ± 0.053

across robots, the method used a cycle-consistency loss that maximized the likelihood of both forward and reverse translations between the source and target domains. During inference, the learner robot encoded the demonstrator's trajectory into a latent skill using the demonstrator's encoder and then reconstructed an executable trajectory using its own decoder. In our implementation, we adopted the same encoder and decoder architectures as those used in the IAIL framework to ensure comparability. This baseline operated entirely in an unsupervised manner and did not rely on task labels or annotations. It aimed to establish correspondences purely on the basis of structural similarities in skill execution. In our experiments, we evaluated this method using robot pairs with differing task distributions and capabilities. Although the method could mitigate embodiment mismatches to some extent, its performance degraded substantially when the demonstrator and learner differed in task distributions, because of its lack of semantic grounding in task intentions.

For the description-based approach, the baseline represented a class of methods that leverage language-conditioned policy learning, in which natural language serves as an intermediate representation for imitation (*46*–*52*). Rather than establishing direct mappings between latent action spaces or trajectories, this paradigm enabled communication between the demonstrator and learner robot through language. This design facilitated zero-shot behavior transfer across embodiments by grounding actions in a shared semantic representation. In this baseline, demonstrated motions were first encoded into textual descriptions that summarized the observed behavior. These descriptions were then transmitted to the learner robot, which decoded them into executable motor actions intended to fulfill the described objective. To train the encoder, we followed a contrastive learning framework introduced in (*53*) to align the motion and the annotation. The decoder was implemented as a language-conditioned policy trained to reproduce the paired motor trajectory given the input annotation, independently for each learner robot. We used the same annotated dataset to train both the encoder and decoder as in the IAIL framework, ensuring that the level of supervision is matched for a fair comparison. This baseline could be viewed as a simplified variant of IAIL that omitted the intention association mechanism and therefore did not account for the learner robot's capabilities during action selection. In our experiments, we evaluated this method using robot pairs with differing task distributions and embodiment constraints. Unlike the density-based baseline, the description-based method was less sensitive to discrepancies in task distributions, because its semantic grounding provided a degree of robustness. However, it lacked an explicit mechanism to assess whether the learner robot can interpret the annotations correctly or execute the decoded actions feasibly. As a result, when the physical capabilities of the learner did not align with those

assumed in the demonstration, the method frequently produced invalid or ineffective behaviors. This issue was particularly pronounced in heterogeneous settings, where we observed substantial performance variability across robot pairs with different capabilities.

Simulation tasks

The evaluation was conducted on two simulated tasks analogous to the user monitoring and item picking phases. Simulation-based evaluation was less affected by real-world disturbances, such as limited training data, sensor noise, or motor inaccuracies, ensuring that the results accurately reflected the core capabilities of each method. In the monitoring task, we focused on the effects of robot variations, whereas in the item picking task, we focused on the effects of environmental variations. The detailed simulation experiment setup is provided in the Supplementary Materials.

The setup for the simulated target monitoring task is illustrated in Fig. 4A. We used four robots (Pepper, Drone, Carter, and Wheeled Biped) to monitor two targets: a blue box on a table and a red box under a table. Owing to their embodiment constraints, each robot had a different preference for monitoring these targets, influencing their action distribution during the training and evaluation phases (Fig. 4B). For example, 100% of the actions performed by Pepper involved monitoring the blue box, whereas 100% of the actions performed by Carter involved monitoring the red box. For the Drone, 90 and 10% of its actions involved the blue box and the red box, respectively, whereas for the Wheeled Biped, 90 and 10% of its actions involved the red box and the blue box, respectively.

The setup for the simulated item picking task is illustrated in Fig. 5B. In this task, we used three UR5 (*54*) robot arms from Universal Robots. The robots had identical kinematic structures but different camera viewpoints. Each robot was assigned a specific set of items and was tasked with picking items from the table in front of it. At each iteration, four items were randomly selected from the assigned set. They were placed on the table with random positions and orientations. The full collection comprised 18 items grouped into five classes, as shown in Fig. 5A.

Performance comparison

We evaluated the imitation performance between all possible demonstrator-learner robot pairs. On the basis of the alignment results after execution, we assigned a score of 1 if the learner robot monitored the correct target or picked the same item as in the demonstration, a score of 0.5 if the learner robot picked an item in the same class, a score of 0 for skipping the action, and a score of −1 for performing an irrelevant action. Ideally, the robot should have performed the demonstrated task when possible, chosen an alternative if an exact match was unfeasible, and remained inactive if the task could not be completed.

The average scores achieved by each demonstrator-learner pair in the monitoring task and item picking task are presented in Figs. 4C and 5C, respectively. Each pair was evaluated over 500 runs, and the evaluation was repeated three times using models trained with different random seeds.

The results of the monitoring task are shown in Fig. 4C. The density-based method was highly sensitive to differences in action distributions between robots. It performed well when the action distributions of the demonstrator and learner were similar, as observed in the Pepper-Drone and Carter–Wheeled Biped pairs. However, its

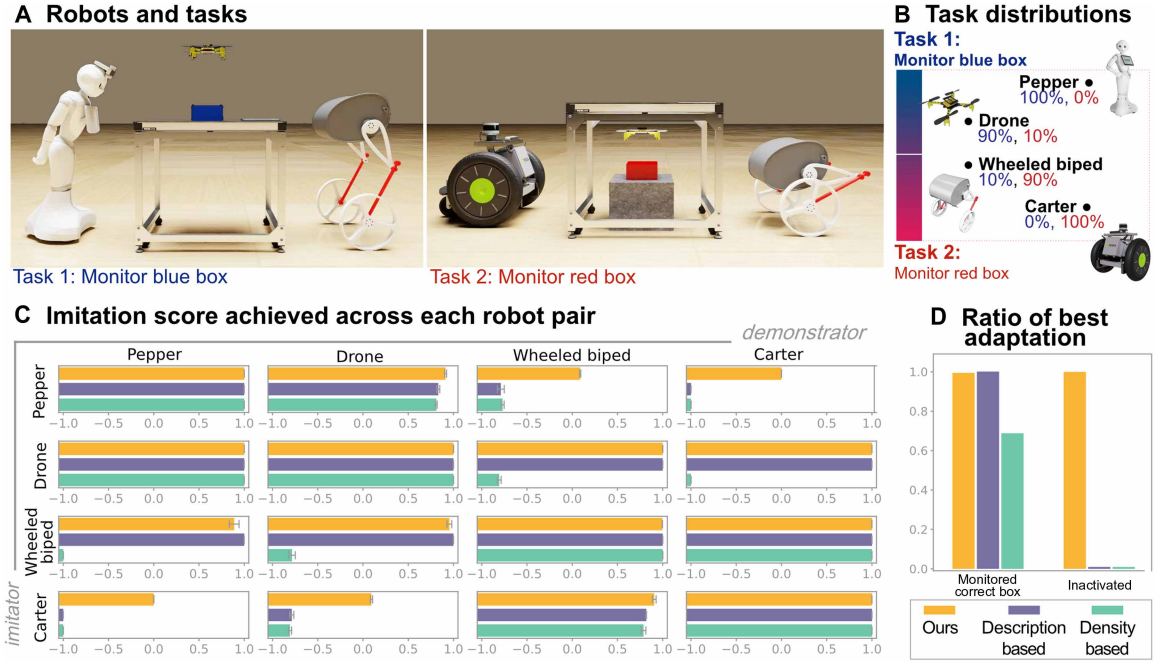

Fig. 4. The simulation study involves a monitoring task designed to evaluate imitation performance across robot pairs with different task distributions in their datasets. (**A**) Example poses of the robots monitoring two boxes: a blue box on the table and a red box under the table. (**B**) Trajectory distributions for monitoring different boxes in each robot's dataset: Because of embodiment limitations, Pepper can only see the blue box, whereas Carter can only see the red one. Drone and Wheeled Biped can see both but exhibit different preferences. (**C**) The average task scores achieved by each demonstrator-learner robot pair. Statistics: Data are presented as mean ± SD. Each bar represents the mean score across $N = 1500$ evaluation runs. Detailed per-pair statistics including effect sizes and CIs are provided in the Supplementary Materials. (**D**) The ratios of selecting the most appropriate action. The evaluation was performed three times using models trained with three different random seeds.

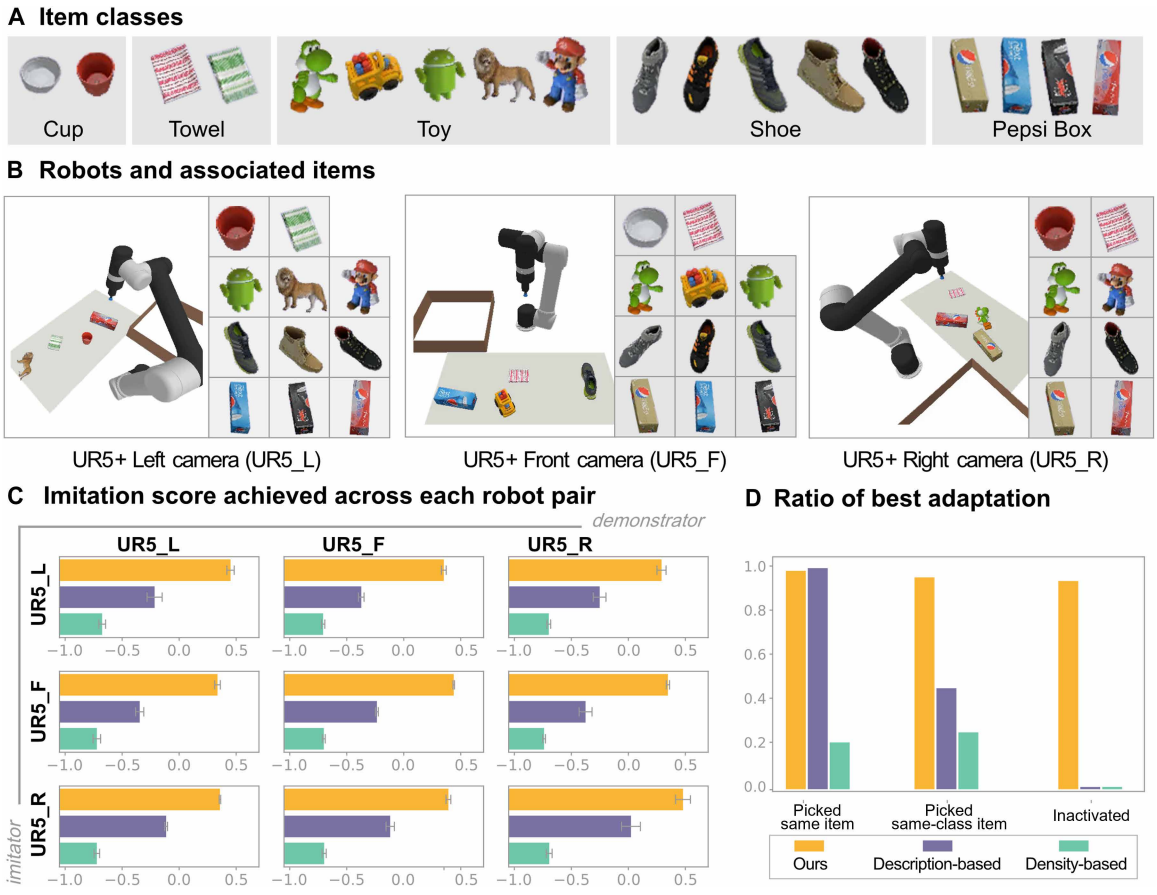

Fig. 5. The simulation study involving selecting a proper item on the basis of demonstrations under different environmental conditions. (**A**) The 18 items used in the task, categorized into five classes. (**B**) Variations in the task environment include different camera angles and selectable item types. An example image from each camera is shown on the left, and the selectable item list associated with each robot is displayed on the right. In each trial, four items were randomly chosen from the associated item list and placed on the table. (**C**) The average task scores achieved by each demonstrator-learner robot pair. Data are mean ± SD. $N = 1500$. Detailed per-pair statistics including effect sizes and CIs are provided in the Supplementary Materials and (**D**) the ratios of selecting the most appropriate item. The evaluation was performed three times using models trained with three different random seeds.

performance degraded substantially when the task distributions in the training data diverged (Fig. 4B). These challenging cases included the Pepper–Wheeled Biped, Pepper-Carter, Drone–Wheeled Biped, and Drone-Carter pairs, with each robot acting as both learner and demonstrator, for a total of eight imitation cases. To validate this observation, we conducted two-sided Welch's t tests for these eight robot pairs. For each pair, the test was performed on the 1500 scores (500 runs, repeated three times using models trained with different random seeds) achieved by IAIL and the density-based method. Across these eight cases with substantial distributional mismatch, IAIL significantly outperformed the density-based baseline (all $P < 0.001$). The unweighted average score difference across the eight cases was $\overline{\Delta} = 1.40$ with a 95% confidence interval (CI) (1.01, 1.79) (SD = 0.47, range from 0.86 to 2.00). For transparency, per-case statistics, including Δ with 95% CI, t, df, P, and Cohen's d (unequal variance), are reported in table S3.

In contrast, the description-based method demonstrated robustness to distribution shifts but struggled when the learner robot lacked the capabilities required to replicate the demonstrator's actions. For example, it succeeded when Carter served as the demonstrator and Drone as the learner but failed in the reverse scenario because of Carter's limited ability to monitor the blue box from an aerial viewpoint. Similar capability mismatches occurred in four imitation cases: Pepper–Wheeled Biped, Pepper-Carter, Carter-Pepper, and Carter-Drone, where the former (as learner) lacked the capability to complete some tasks demonstrated by the latter (as demonstrator). In these four cases, Welch's t tests again showed that IAIL significantly outperformed the description-based baseline (all $P \leq 0.001$). The unweighted average score difference of the four cases was $\overline{\Delta} = 0.94$ with 95% CI (0.84, 1.04) (SD = 0.63, range from 0.85 to 1.00). Full per-case statistics (Δ with 95% CI, t, df, P, and Cohen's d) are reported in table S4.

Both the density-based and description-based methods share a key limitation: They cannot detect when a demonstrated task is impossible for the learner robot to complete. This issue is particularly evident with the Pepper-Carter pair, for which the average score was −1, indicating consistent execution of incorrect actions by the robots, which could lead to failure or unexpected errors in real-world scenarios. These findings were supported by the ratio of the best adaptation in different scenarios, as shown in Fig. 4D. With the description-based method, the demonstrated task was correctly executed when the learner robot could perform the task; however, this

method failed to detect infeasible tasks. The performance of the density-based method was even worse, given that it was also affected by the action distributions of the robots.

The results of the item picking task are shown in Fig. 5C. There was a noticeable decline in the scores of each robot pair in the item picking task compared with the scores in the monitoring task. This decline was due to increased task complexity and higher-dimensional state and action spaces.

The density-based method yielded the lowest scores across all robot pairs. The description-based method performed better but still failed to achieve an average score of 0, indicating that irrelevant or incorrect actions were frequently executed. To validate these observations, we performed Welch's t tests across all nine imitation cases involving different robot pairs in the item-picking task. The performance differences between IAIL and both baseline methods were statistically significant (all $P < 0.001$). For the density-based method, the unweighted average score difference across the nine cases was $\overline{\Delta} = 1.11$ with 95% CI (1.08, 1.14) (SD = 0.04, range from 1.02 to 1.18). For the description-based method, the unweighted average score difference was $\overline{\Delta} = 0.63$ with 95% CI (0.55, 0.70) (SD = 0.10, range from 0.47 to 0.74). Full per-case statistics (Δ with 95% CI, t, df, P, and Cohen's d) are reported in tables S5 and S6.

A similar pattern was observed in the ratio of the best actions performed, as shown in Fig. 5D, which reflected the trend in Fig. 5C. The description-based method achieved good results when the exact demonstrated item was available. However, its performance decreased substantially when adapting to an alternative item in the same class. The density-based method struggled in both scenarios. In addition, both methods were unable to handle cases in which the demonstrated actions were impossible to perform.

For both tasks, our method consistently achieved the highest scores for almost all robot pairs and across all types of imitation scenarios. Unlike the density-based method, which relies solely on aligning the action distributions of the learner and demonstrator robots, our approach leveraged the semantic information embedded in the robot actions. This enabled more accurate and meaningful action association. Moreover, in contrast with the description-based method, which relies on descriptions without considering the learner's capabilities, our method integrated context-aware motion generation and intention-based association. This ensured that the selected actions were both executable for the learner robot and aligned with the demonstration, enabling greater adaptability to changing conditions.

In addition to the performance issues, the baseline methods had limited scalability for supporting IL between robot teams. Specifically, they lacked the ability to compare motions across different robots and, therefore, could not effectively address the role assignment problem in team-to-team imitation scenarios. This limitation notably restricted their applicability in complex, multirobot environments. In contrast, our IAIL framework overcame these challenges. By leveraging semantic motion understanding and association in the intention space, IAIL can seamlessly handle IL between heterogeneous teams of robots, regardless of their numbers, types, or configurations. It dynamically assigns roles within the learner team on the basis of individual capabilities and adapts to evolving task requirements without relying on predefined distributions or rigid descriptions. This flexibility enables IAIL to scale effectively across diverse robotic systems, ensuring robust and adaptable performance in real-world scenarios.

DISCUSSION

In this work, we introduced the IAIL framework, which enhances the adaptability, generalizability, and scalability of IL for heterogeneous robotic systems, thereby extending its applicability to dynamic and complex real-world scenarios. The framework leveraged the intentions underlying robot motions to enable heterogeneous robots to automatically adjust behaviors on the basis of demonstrator-learner relations, accommodating variations in tasks, robot types, and operating environments. Our results demonstrated that the IAIL framework substantially improves robots' abilities to learn and perform tasks by imitating the behaviors of other robots. It also reduces reliance on task-specific demonstrations requiring precise alignment with target configurations.

This adaptability is especially critical for heterogeneous robot teams, where robots with diverse characteristics must collaborate to complete a wide range of tasks under changing conditions. In addition to improving task performance, the framework's scalability allows it to support diverse robot team configurations, making it particularly well suited for real-world applications.

Recent advances have emphasized learning from large-scale, heterogeneous datasets collected from diverse robot platforms performing a wide range of tasks in varied environments, such as Open X-Embodiment (*50*), Octo (*51*), OpenVLA (*52*), and Heterogeneous Pretrained Transformers (HPT) (*55*). These efforts aim to develop general-purpose representations that can accelerate task learning for individual robots and enable implicit knowledge sharing across contexts. However, these methods primarily focus on learning universal policies or representations, without explicitly tackling the challenge of transferring task-solving behaviors between robots with different embodiments, an ability that is essential for effective cross-robot generalization. Inspired by human imitation mechanisms, the IAIL framework bridges this gap by using high-level motion intentions to enable direct behavior association and adaptation in a shared representation space (*56*). This formulation preserves task-agnostic representations and enables explicit and flexible skill transfer across heterogeneous robot embodiments. By allowing robots to interpret, imitate, and adapt the task-solving strategies of others, IAIL fosters more efficient collaboration and broader generalization in diverse, real-world robotic systems.

Although our framework leverages language to explicitly represent intention, we acknowledge that intentions can also be modeled implicitly through goal-inference processes, in which observers deduce goals from observational or behavioral cues (*57*–*59*). However, in the context of heterogeneous robot teams, relying solely on implicit behavioral cues is challenging, because the observable forms of motion and action modalities differ notably between agents. In this setting, our linguistic approach offers a complementary and robust perspective, providing a shared, high-level code for intention that bridges embodiment gaps where direct visual or motion correspondence is unreliable. Furthermore, by grounding imitation in linguistically specified intent, our framework is closely related to the objective of robot motion legibility in the robotics literature, which seeks to make a robot's goals efficiently inferable from its behavior (*60*, *61*). By organizing robot behaviors in a shared intention space aligned with human-understandable descriptors, IAIL supports both legibility and predictability, which is particularly advantageous in collaborative scenarios, especially when humans are involved (*62*, *63*).

As a promising extension, integrating large language models (LLMs) could further unlock the potential of the IAIL framework in robotic skill acquisition and task allocation. By using the annotation encoder trained alongside the motion encoder, the IAIL framework can process language instructions as seamlessly as it processes demonstration trajectories. This capability enables smooth integration with LLMs, expanding the framework's ability to interpret and act upon language-based inputs. When given a language instruction, the annotation encoder f_ξ extracts the underlying intention and identifies the motion most closely aligned in the intention space. This allows the framework to select the action most likely to achieve the desired outcome based on the inferred intention. If collecting trajectory demonstrations becomes impractical, LLMs can be leveraged to automatically generate instructions as demonstrations. The only required adjustment is switching the encoder from f_ψ to f_ξ in the computation of V_{task}. As a preliminary exploration, fig. S5 shows an example of the integrated pipeline for task planning and execution with a learner robot team, and a video of this example is provided in movie S3.

One limitation of our framework is identifying the decision boundary to determine when the generated actions do not satisfy the demonstrated task, requiring the robot to remain inactive. Now, we apply a fixed threshold for all of the robots in the group. Although this approach makes the system more generalizable and user-friendly, it may hinder performance because the fixed threshold might not be optimal for every robot. This process could be further enhanced by automatically assigning and tuning parameters for each robot.

In the future, we intend to explore more complex imitation scenarios in which demonstrations may need to be broken down into more manageable components or reassembled according to the capabilities of the learner robots. Furthermore, we aim to incorporate multimodal inputs, combining robot trajectory data, text descriptions, and sensor feedback to enable a more comprehensive understanding of the task, thereby enhancing the flexibility and applicability of the framework. Future efforts will focus on refining the framework to improve its user-friendliness and accessibility. For instance, we plan to devise methods for automatically annotating robot trajectories and tuning the hyperparameters of our model. These methods would simplify the creation of training data and reduce technical barriers, making the methodology available to a broader audience of researchers and practitioners.

MATERIALS AND METHODS

As described in Results, our IAIL framework operates through three key processes: context-aware motion generation, motion intention extraction, and motion association. These processes involve training three types of models: the motion generator (p_{θ_i}), the motion encoder (p_{ψ_i}), and the annotation encoder (f_ξ). The motion generator and encoder are robot-specific, whereas the annotation encoder is shared across robots. The following subsections provide detailed explanations of the training procedures for these models, the method for associating actions between robots, and the approach for enabling imitation in robot teams.

Learning process of the motion generator

The motion generator p_{θ_i} is responsible for assessing each robot's capabilities in various contexts. This capability is manifested through the action trajectories that p_{θ_i} can generate for each robot state. We trained the motion generator with a precollected expert dataset that contains action trajectories completing diverse tasks within the robot's domain. The objective of the motion generator is to reproduce the action trajectories from the dataset. As a result, we can generate safe and executable motion trajectories that enable diverse goals to be achieved considering the context of each robot.

Collecting the robot trajectory datasets

The dataset for each robot was collected independently by human experts. The experts first defined a set of tasks and then performed these tasks using the robot. We repeated each task multiple times, collecting trajectories under different conditions, such as varying object types, locations, or orientations, and different robot initial positions. Whenever possible, we used different trajectories to complete the same task, ensuring a comprehensive coverage of the variations in the task environment. We recorded the robot trajectories $\tau = (s, \mathbf{a})$, which consists of the initial state s and the sequence of actions $\mathbf{a} = (a_0, \ldots, a_H)$ executed by the robot. We denote the dataset for each robot as D_i and the combined dataset from all robots as $\mathrm{D} = \cup_{i=1}^{N} D_i$.

Training the generator

The motion generator for robot i is modeled as a latent-variable, state-conditioned generative model $p_{\theta_i}(\mathbf{a} \mid s, z)$, where s is the robot's state and z is a latent variable capturing the underlying variations in the action sequences. This generative model was trained to reproduce action sequences $\mathbf{a}$ in the precollected dataset given different states s. In this work, we used a variational autoencoder (VAE) (*64*) to train the generative model. Alternative methods such as generative adversarial networks (*65*) and diffusion models (*66*) could also be used.

The objective of the VAE for robot i is as follows

$$\underset{\vartheta_i,\theta_i}{\operatorname{argmax}} \mathop{\mathbb{E}}_{(s,\mathbf{a})\sim D_i}\left[\mathbb{E}_{q_{\vartheta_i}(z|s,\mathbf{a})}\left[\log\left(p_{\theta_i}(\mathbf{a}\mid s,z)\right) - \beta_{\mathrm{KL}}\left(q_{\vartheta_i}(z\mid s,\mathbf{a}) \| p(z)\right)\right]\right] \quad (1)$$

where $q_{\vartheta_i}(z|s,\mathbf{a})$ denotes the variational posterior distribution of the encoder. $p_{\theta_i}(\mathbf{a}|s,z)$ represents the decoder's posterior distribution, which serves as our generative model. $p(z)$ is the prior distribution over the latent variable, which is typically modeled as a standard normal distribution, and β_{KL} is a hyperparameter that balances the reconstruction loss and the Kullback-Leibler (KL) divergence between the posterior and prior distributions. The generative model for each robot is trained independently with its corresponding dataset D_i. Please refer to (*64*) for a more detailed description of the VAE.

In theory, the generative model should produce action sequences that lie within the distribution of the training dataset. This property allows for the sampling of safe and executable actions for performing diverse tasks within each robot's domain. However, in practice, the model may generate out-of-distribution (OOD) actions because of interpolation between data samples (*64*, *67*). These OOD actions can lead the robot to unexpected states, introducing uncertainties during task execution. To mitigate this issue, it is essential to identify and eliminate OOD actions from the generated actions. Such actions are identified and eliminated through the motion encoding and associating steps, which are detailed in the next section.

Joint learning of the motion and annotation encoders

The motion encoder p_{θ_i} for robot i is responsible for extracting the intentions underlying the robot's motions. These motion intentions were defined by human-annotated language descriptions that capture the core objectives of the motions. The motion intentions serve as unified motion representations across all robots, enabling each

robot to interpret and compare its own motions with those of other robots. The motion encoder achieves this goal by mapping robot motions to a latent space, also referred to as the intention space, where each dimension corresponds to a specific motion intention.

We use a shared annotation encoder f_ξ to process the language annotations. The robot-specific motion encoder p_{θ_i} and the shared annotation encoder f_ξ are trained jointly using a contrastive learning approach. The aim of this joint training is to align the motion encodings with their corresponding annotation encodings. By doing so, we ensure that motions with similar intentions are encoded closely together in the intention space, even if they originate from different robots.

Annotating the robot trajectory datasets

We annotated the trajectories in the precollected datasets to construct the motion-annotation pairs required to train the motion encoders. Each trajectory was assigned three to five language descriptions with varying levels of abstraction, ranging from detailed to concise. For instance, consider a trajectory where a robotic arm picks up a white paper cup. The annotations for this trajectory include "pick up a white paper cup," "pick up a paper cup," "pick up a white cup," and "pick up a cup." To further enrich the annotations, we used different synonyms and phrasings to augment the descriptions, ensuring that the key attributes of each trajectory are captured in diverse ways.

The OOD actions generated by the motion generator p_{θ_i} were not included in the initial dataset. Consequently, their encodings in the intention space did not accurately reflect the true outcomes of the motion executions. To correctly interpret the OOD actions, we extended the dataset by incorporating actions sampled directly from the motion generator $p_{\theta_i}(\mathbf{a}|s,z)$ using random z values drawn from the prior distribution $p(z)$. We annotated these sampled actions as "unknown" because their precise outcomes were uncertain.

In the annotated and extended dataset D^l, OOD actions were annotated solely as unknown, whereas in-distribution data included both specific annotations and unknown actions. By training the encoders with this extended dataset, OOD actions were encoded much closer to the unknown label in the intention space, enabling effective identification of these actions.

Joint training of the motion and annotation encoders

The motion encoder $f_{\psi_i}(\tau)$ processes and maps the trajectory of robot i to a multidimensional encoding. The annotation encoder $f_\xi(l)$ maps language annotations to encodings of the same dimensionality as those produced by the motion encoder. We maximized the mutual information between these encodings via a contrastive learning approach (*53*). In this approach, the encoding distance between correct (positive) motion-annotation pairs is minimized, whereas the encoding distance between incorrect (negative) motion-annotation pairs is maximized. The loss function is defined as follows

$$L_{\psi,\xi} = L^{\tau\to l}_{\psi,\xi} + L^{l\to\tau}_{\psi,\xi} \tag{2}$$

where

$$L^{\tau\to l}_{\psi,\xi} = \underset{\tau_i, l_i\sim D^l}{-\mathbb{E}}\left[\log\frac{\varepsilon(\tau_i,l_i)}{\varepsilon(\tau_i,l_i)+\sum_{l_j\sim D^l}^{N_b-1}\varepsilon(\tau_i,l_j)}\right] \tag{3}$$

and

$$L^{l\to\tau}_{\psi,\xi} = \underset{\tau_i, l_i\sim D^l}{-\mathbb{E}}\left[\log\frac{\varepsilon(\tau_i,l_i)}{\varepsilon(\tau_i,l_i)+\sum_{\tau_j\sim D^l}^{N_b-1}\varepsilon(\tau_j,l_i)}\right] \tag{4}$$

Here, D^l represents the union of the annotated datasets of all of the robots, $\tau_i = (s, \mathbf{a})$ is a sampled trajectory, and l_i is the annotation of τ_i. We construct negative motion-annotation pairs by randomly sampling $N_b - 1$ elements from D^l (sampling l_j for $L^{\tau\to l}_{\psi,\xi}$ and τ_j for $L^{l\to\tau}_{\psi,\xi}$), where N_b is a positive integer. The negative samples τ_j and l_j may belong to different domains than (τ_i, l_i). The score function $\varepsilon(\tau_i, l)$ is defined as the exponential of the cosine similarity between the encodings, which can be expressed as

$$\varepsilon(\tau_i,l) = \exp\left(\left\langle f_{\psi_i}(\tau_i), f_\xi(l)\right\rangle\right) \tag{5}$$

where f_{ψ_i} represents the trajectory encoder for the domain to which trajectory τ_i belongs. The operator $\langle,\rangle$ denotes the cosine similarity between the two inputs.

Associating motions on the basis of the intention similarity score

When robot i receives a demonstration τ_j from robot j for imitation, robot i replicates τ_j by associating τ_j with one of its executable motions. The association process involves the following steps. First, given the current state s_i, a batch of candidate trajectories $\left\{\tau_i^{(k)}\right\}$ from the motion generator p_{θ_i} of robot i is sampled. Each sampled trajectory $\tau_i^{(k)} = \left(s_i, \mathbf{a}_i^{(k)}\right)$ is then encoded by its motion encoder $f_{\psi_i}\left(\tau_i^{(k)}\right)$, and the demonstration τ_j is encoded by robot j's motion encoder $f_{\psi_j}(\tau_j)$.

Next, each sampled trajectory for robot i is evaluated on the basis of its validity as being in-distribution and the similarity of its intention to that of τ_j. The validity is calculated as

$$V_{\text{valid}}\left(\tau_i^{(k)}\right) = -\varepsilon\left(\tau_i^{(k)}, ''\text{unknown}''\right) = -\exp\left(\left\langle f_{\psi_i}\left(\tau_i^{(k)}\right), f_\xi(''\text{unknown}'')\right\rangle\right) \tag{6}$$

which represents the encoding distance between the given trajectory and the unknown label. The intention similarity is calculated as

$$V_{\text{aligned}}\left(\tau_i^{(k)}, \tau_j\right) = \varepsilon\left(\tau_i^{(k)}, \tau_j\right) = \exp\left(\left\langle f_{\psi_i}\left(\tau_i^{(k)}\right), f_{\psi_j}(\tau_j)\right\rangle\right) \tag{7}$$

which represents the encoding distance between the given trajectory and the demonstration τ_j. We use two preset thresholds, ϵ_{valid} and $\epsilon_{\text{aligned}}$, to define the minimum requirements for the in-distribution validity and intention alignment, respectively. The candidate batch B containing valid actions to replicate τ_j is then defined as follows

$$B = \left\{\tau_i^{(k)} \,|\, \tau_i^{(k)}\sim p_{\theta_i},\ V_{\text{aligned}}\left(\tau_i^{(k)}, \tau_j\right) > \epsilon_{\text{aligned}},\ V_{\text{valid}}\left(\tau_i^{(k)}\right) > \epsilon_{\text{valid}}\right\} \tag{8}$$

where p_{θ_i} denotes the motion generator of robot i and $\tau_i^{(k)} = \left(s_i, \mathbf{a}_i^{(k)}\right)$ represents the kth sampled trajectory for robot i.

If no candidate actions satisfy both thresholds, then robot i remains inactive, indicating its inability to replicate the demonstrated task. Conversely, if one or more candidate actions meet the criteria, then the trajectory with the highest overall score is selected for execution. The action selection policy $\pi(\tau|\tau_j)$ is therefore defined as

$$\pi(\tau \mid \tau_j) = \begin{cases} \underset{\tau_i^{(k)} \in B}{\operatorname{argmax}} V\left(\tau_i^{(k)}, \tau_j\right) & \text{if } |B| > 0, \\ \text{inactive} & \text{otherwise} \end{cases} \quad (9)$$

where V is the overall score, which is defined as

$$V\left(\tau_i^{(k)}, \tau_j\right) = V_{\text{aligned}}\left(\tau_i^{(k)}, \tau_j\right) + V_{\text{valid}}\left(\tau_i^{(k)}\right) \quad (10)$$

In summary, robot i selects an action that is both valid (in-distribution) and aligned with the intention of the demonstrated action, ensuring safe and effective imitation of motions across different robots.

Assigning motions to the best robot on the team

When a team of robots received a demonstration τ_j by robot j, we extended the action association process to leverage the capabilities of all available robots in the team. Instead of sampling actions from a single robot, we drew candidate trajectories from all learner robots on the team. This approach enhances the flexibility and adaptability of the system by using the diverse motion capabilities of multiple robots.

This flexibility was achieved by incorporating sample trajectories from all available robots in the selection process, as outlined in Eq. 11.

We defined a set of robots, denoted as R_{team}, representing the available target robots in the team. The extended candidate batch B_{team} is constructed as follows

$$B_{\text{team}} = \bigcup_{i \in R_{\text{team}}} \left\{ \tau_i^{(k)} \mid \tau_i^{(k)} \sim p_{\theta_i},\ V_{\text{valid}}\left(\tau_i^{(k)}\right) > \epsilon_{\text{valid}},\ V_{\text{aligned}}\left(\tau_i^{(k)}, \tau_j\right) > \epsilon_{\text{aligned}} \right\} \quad (11)$$

where p_{θ_i} denotes the motion generator of robot i on the team and $\tau_i^{(k)} = \left(s_i, \mathbf{a}_i^{(k)}\right)$ represents the kth sampled trajectory for robot i. Here, s_i is the current state of robot i, and $\mathbf{a}_i^{(k)}$ is the action sequence generated by $p_{\theta_i}(s_i, z)$ with $z \sim p(z)$.

The action selection criteria are the same as those in Eq. 9. If no candidate trajectories satisfy both thresholds, then the robot team remains inactive, indicating that no robot on the team is able to replicate the demonstrated task. Conversely, if one or more candidates meet the criteria, the trajectory with the highest overall score $V = V_{\text{aligned}} + V_{\text{valid}}$ is selected for execution by the robot that generated this action. This procedure is illustrated in Fig. 6.

Statistical analysis

Monitoring task

We used the Welch's unequal-variance t test for the monitoring task. Test statistics for the 16 demonstrator-learner pairs for comparing IAIL against the density-based baseline and the description-based baseline are reported in tables S3 and S4, respectively. The statistics were computed using 1500 test scores for each method. For each pair, we report the mean difference Δ (IAIL – baseline), its 95% CI, Welch's t statistic, df, two-sided P, and the effect size Cohen's d (with

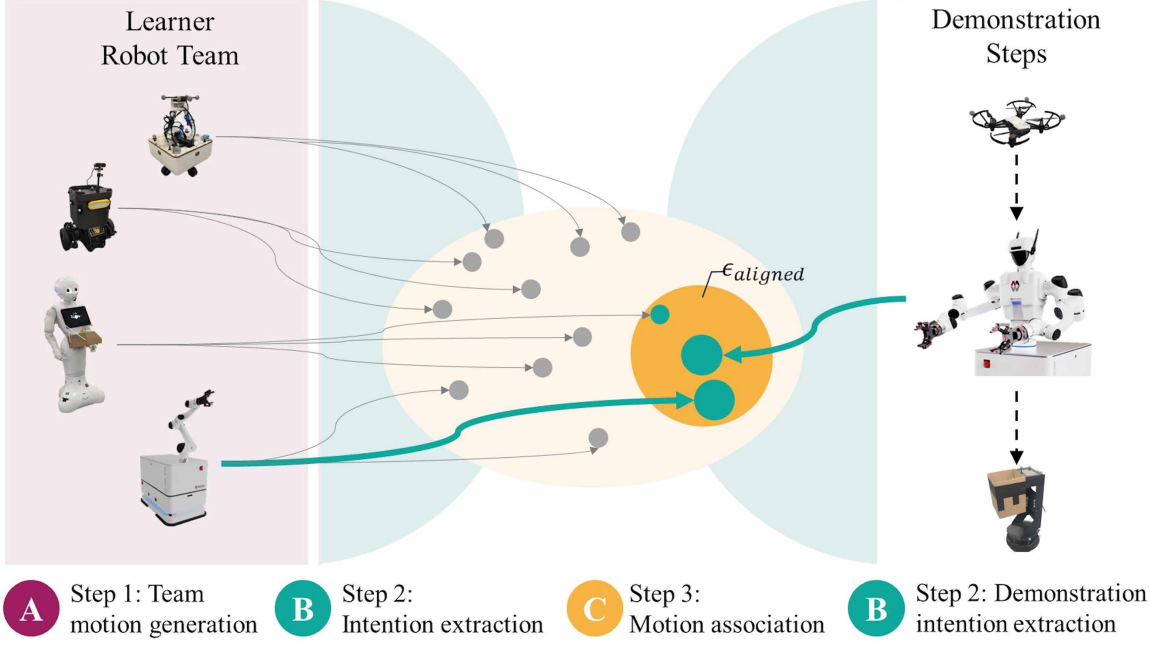

Fig. 6. The process of action association between robot teams. We processed the demonstrations provided by the demonstrator team individually. For each demonstration, we used the following steps: (**A**) First, a candidate batch of actions from all of the robots on the learner team was sampled using their motion generators given their current contexts. (**B**) Next, we extracted the intentions of these actions by projecting them to a shared embedding space using their motion encoders. (**C**) Last, we associated the demonstration with one of the sampled actions that was within the boundary of $\epsilon_{\text{aligned}}$ and was closest to the demonstration in the embedding space. The selected action was then sent to the robot that generated this action for execution.

95% CI). All values are shown with three decimals; *P* values smaller than 0.001 are shown as "<0.001."

When scores were nearly deterministic (for example, all 1500 tests had identical scores) for both methods, the within-condition SDs became extremely small. The corresponding *t*, df, and Cohen's *d* were ill defined or numerically unstable. For such rows, we indicate *t*, df, and Cohen's *d* as "–." The eight pairs with substantial distributional divergence and the four pairs with capability mismatch are highlighted in bold in tables S3 and S4, respectively.

Item picking task

We used the Welch's unequal-variance *t* test for the item picking task. Test statistics for the nine demonstrator-learner pairs for comparing IAIL against the density-based baseline and the description-based baseline are reported in tables S5 and S6, respectively. For each pair, we report the mean difference Δ (IAIL – baseline), its 95% CI, Welch's *t* statistic, df, two-sided *P*, and the effect size Cohen's *d* (with 95% CI). Following the standard practice, *P* values smaller than 0.001 are shown as <0.001.

Supplementary Materials

The PDF file includes:

Supplementary Methods
Supplementary Experimental Setups
Supplementary Results
Figs. S1 to S5
Tables S1 to S6
Legends for movies S1 to S3
References (*68*–*73*)

Other Supplementary Material for this manuscript includes the following:

Movies S1 to S3
Data files S1 and S2

REFERENCES AND NOTES

1. B. D. Argall, S. Chernova, M. Veloso, B. Browning, A survey of robot learning from demonstration. *Robot. Auton. Syst.* **57**, 469–483 (2009).
2. H. Ravichandar, A. S. Polydoros, S. Chernova, A. Billard, Recent advances in robot learning from demonstration. *Annu. Rev. Control Robot. Auton. Syst.* **3**, 297–330 (2020).
3. Y. Ma, A. Cramariuc, F. Farshidian, M. Hutter, Learning coordinated badminton skills for legged manipulators. *Sci. Robot.* **10**, eadu3922 (2025).
4. X. Chen, A. Ghadirzadeh, T. Yu, J. Wang, A. Y. Gao, W. Li, L. Bin, C. Finn, C. Zhang, "LAPO: Latent-variable advantage-weighted policy optimization for offline reinforcement learning" in *Advances in Neural Information Processing Systems 35 NeurIPS 2022*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh, Eds. (Curran Associates, 2022), pp. 36902–36913.
5. M. Zare, P. M. Kebria, A. Khosravi, S. Nahavandi, A survey of imitation learning: Algorithms, recent developments, and challenges. *IEEE Trans. Cybern.* **54**, 7173–7186 (2024).
6. Y. Hu, F. J. Abu-Dakka, F. Chen, X. Luo, Z. Li, A. Knoll, W. Ding, Fusion dynamical systems with machine learning in imitation learning: A comprehensive overview. *Inf. Fusion* **108**, 102379 (2024).
7. A. Ghadirzadeh, X. Chen, P. Poklukar, C. Finn, M. Bjo¨rkman, D. Kragic, "Bayesian meta-learning for few-shot policy adaptation across robotic platforms" in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2021), pp. 1274–1280.
8. D. Hejna, L. Pinto, P. Abbeel, "Hierarchically decoupled imitation for morphological transfer" in *Proceedings of the 37th International Conference on Machine Learning*, H. Daumé III, A. Singh, Eds., vol. 119 of *Proceedings of Machine Learning Research* (PMLR, 2020), pp. 4159–4171.
9. Y. Liu, A. Gupta, P. Abbeel, S. Levine, "Imitation from observation: Learning to imitate behaviors from raw video via context translation" in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2018), pp. 1118–1125.
10. H. Liu, C. Zhang, Y. Zhu, C. Jiang, S.-C. Zhu, Mirroring without overimitation: Learning functionally equivalent manipulation actions. *Proc. AAAI Conf. Artif. Intell.* **33**, 8025–8033 (2019).
11. X. Chen, A. Ghadirzadeh, M. Björkman, P. Jensfelt, "Adversarial feature training for generalizable robotic visuomotor control" in *2020 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2020), pp. 1142–1148.
12. J. Lee, M. Bjelonic, A. Reske, L. Wellhausen, T. Miki, M. Hutter, Learning robust autonomous navigation and locomotion for wheeled-legged robots. *Sci. Robot.* **9**, eadi9641 (2024).
13. C. Finn, T. Yu, T. Zhang, P. Abbeel, S. Levine, "One-shot visual imitation learning via meta-learning" in *Proceedings of the First Conference on Robot Learning*, S. Levine, V. Vanhoucke, K. Goldberg, Eds., vol. 78 of *Proceedings of Machine Learning Research* (PMLR, 2017), pp. 357–368.
14. T. Yu, C. Finn, S. Dasari, A. Xie, T. Zhang, P. Abbeel, S. Levine, "One-shot imitation from observing humans via domain-adaptive meta-learning" in *Proceedings of Robotics: Science and Systems XIV*, H. Kress-Gazit, S. Srinivasa, T. Howard, N. Atanasov, Eds. (RSS Foundation, 2018), 10.15607/RSS.2018.XIV.002.
15. E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, C. Finn, "BC-Z: Zero-shot task generalization with robotic imitation learning" in *Proceedings of the 5th Conference on Robot Learning*, A. Faust, D. Hsu, G. Neumann, Eds., vol. 164 of *Proceedings of Machine Learning Research* (PMLR, 2022), pp. 991–1002.
16. J. Beck, R. Vuorio, E. Z. Liu, Z. Xiong, L. Zintgraf, C. Finn, S. Whiteson, A tutorial on meta-reinforcement learning. *Found. Trends Mach. Learn.* **18**, 224–384 (2025).
17. J. Li, T. Lu, X. Cao, Y. Cai, S. Wang, "Meta-imitation learning by watching video demonstrations" poster presented at the Tenth International Conference on Learning Representations (ICLR 2022), virtual, 25 April 2022. https://openreview.net/forum?id=KTPuIsx4pmo.
18. T. Shankar, Y. Lin, A. Rajeswaran, V. Kumar, S. Anderson, J. Oh, "Translating robot skills: Learning unsupervised skill correspondences across robots" in *Proceedings of the 39th International Conference on Machine Learning*, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, S. Sabato, Eds., vol. 162 of *Proceedings of Machine Learning Research* (PMLR, 2022), pp. 19626–19644.
19. M. Bauza, A. Bronars, Y. Hou, I. Taylor, N. Chavan-Dafle, A. Rodriguez, SimPLE, a visuotactile method learned in simulation to precisely pick, localize, regrasp, and place objects. *Sci. Robot.* **9**, eadi8808 (2024).
20. J. Allen, J. Anderson, J. Baltes, Vision-based imitation learning in heterogeneous multi-robot systems: Varying physiology and skill. *Int. J. Autom. Smart Technol.* **2**, 147–161 (2012).
21. J. Song, H. Ren, D. Sadigh, S. Ermon, "Multi-agent generative adversarial imitation learning" in *Advances in Neural Information Processing Systems 31 NeurIPS 2018*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett, Eds. (Curran Associates, 2018), pp. 7472–7483.
22. P. Feng, T. Yang, M. Liang, L. Wang, Y. Gao, OC-HMAS: Dynamic self-organization and self-correction in heterogeneous multiagent systems using multimodal large models. *IEEE Internet Things J.* **12**, 13538–13555 (2025).
23. Y. Gao, J. Chen, X. Chen, C. Wang, J. Hu, F. Deng, T. L. Lam, Asymmetric self-play-enabled intelligent heterogeneous multirobot catching system using deep multiagent reinforcement learning. *IEEE Trans. Robot.* **39**, 2603–2622 (2023).
24. H. Chakraa, F. Gue'rin, E. Leclercq, D. Lefebvre, Optimization techniques for Multi-Robot Task Allocation problems: Review on the state-of-the-art. *Robot. Auton. Syst.* **168**, 104492 (2023).
25. K. Athira, S. Umashankar, A systematic literature review on multi-robot task allocation. *ACM Comput. Surv.* **57**, 1–28 (2024).
26. G. Gergely, H. Bekkering, I. Király, Rational imitation in preverbal infants. *Nature* **415**, 755–755 (2002).
27. A. N. Meltzoff, Understanding the intentions of others: Re-enactment of intended acts by 18-month-old children. *Dev. Psychol.* **31**, 838–850 (1995).
28. M. Tomasello, Cultural learning redux. *Child Dev.* **87**, 643–653 (2016).
29. B. S. Hewlett, A. H. Boyette, S. Lew-Levy, S. Gallois, S. J. Dira, Cultural transmission among hunter-gatherers. *Proc. Natl. Acad. Sci. U.S.A.* **121**, e2322883121 (2024).
30. Z. H. Garfield, S. Lew-Levy, Teaching is associated with the transmission of opaque culture and leadership across 23 egalitarian hunter-gatherer societies. *Nat. Commun.* **16**, 3387 (2025).
31. L. Bonini, C. Rotunno, E. Arcuri, V. Gallese, Mirror neurons 30 years later: Implications and applications. *Trends Cogn. Sci.* **26**, 767–781 (2022).
32. E. Oztop, M. Kawato, M. A. Arbib, Mirror neurons: Functions, mechanisms and models. *Neurosci. Lett.* **540**, 43–55 (2013).
33. E. Oztop, M. Kawato, M. Arbib, Mirror neurons and imitation: A computationally guided review. *Neural Netw.* **19**, 254–271 (2006).
34. N. Nishitani, R. Hari, Temporal dynamics of cortical representation for action. *Proc. Natl. Acad. Sci. U.S.A.* **97**, 913–918 (2000).
35. J. Fischer, Physical reasoning is the missing link between action goals and kinematics. A comment on "An active inference model of hierarchical action understanding, learning, and imitation" by Proietti *et al*. *Phys. Life Rev.* **48**, 198–200 (2024).
36. Ryze Tech, TELLO User Manual v1.4 (2018); https://dl-cdn.ryzerobotics.com/downloads/Tello/Tello User Manual v1.4.pdf.
37. Shenzhen Moying Technology Co. Ltd., Robot product information; https://moyingrobotics.com/en/Robot/index.html.

38. NXROBO, spark noetic: ROS wrapper and applications for the Spark robot (GitHub repository); https://github.com/NXROBO/spark_noetic.
39. L. Zhang, Y. Huang, Z. Cao, Y. Jiao, H. Qian, Parallel self-assembly for a multi-USV system on water surface with obstacles. *IEEE Trans. Autom. Sci. Eng.* **22**, 2213–2224 (2025).
40. SoftBank Robotics America, Meet Pepper: The robot built for people; https://us.softbankrobotics.com/pepper.
41. Direct Drive Technology Limited, DIABLO: Product page; https://en.directdrive.com/product_diablo.
42. D. L. Davies, D. W. Bouldin, A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)* **1**, 224–227 (1979).
43. K. Hu, Z. Rui, Y. He, Y. Liu, P. Hua, H. Xu, "Stem-OB: Generalizable visual imitation learning with stem-like convergent observation through diffusion inversion" in *The Tenth International Conference on Learning Representations (ICLR 2025)* (Curran Associates, 2025); https://openreview.net/forum?id=KTPuIsx4pmo.
44. K. Pertsch, R. Desai, V. Kumar, F. Meier, J. J. Lim, D. Batra, A. Rai, "Cross-domain transfer via semantic skill imitation" in *Proceedings of the 6th Conference on Robot Learning* K. Liu, D. Kulic, J. Ichnowski, Eds., vol. 205 of *Proceedings of Machine Learning Research* (PMLR, 2023), pp. 690–700.
45. L. Y. Chen, C. Xu, K. Dharmarajan, R. Cheng, K. Keutzer, M. Tomizuka, Q. Vuong, K. Goldberg, "RoVi-Aug: Robot and viewpoint augmentation for cross-embodiment robot learning" in *Proceedings of the 8th Conference on Robot Learning*, P. Agrawal, O. Kroemer, W. Burgard, Eds., vol. 270 of *Proceedings of Machine Learning Research* (PMLR, 2025), pp. 209–233.
46. I. Nematollahi, B. DeMoss, A. L. Chandra, N. Hawes, W. Burgard, I. Posner, "LUMOS: Language-conditioned imitation learning with world models" in *2025 IEEE International Conference on Robotics and Automation (ICRA)* (2025), pp. 8219–8225.
47. X. Yao, T. Blei, Y. Meng, Y. Zhang, H. Zhou, Z. Bing, K. Huang, F. Sun, A. Knoll, Long-horizon language-conditioned imitation learning for robotic manipulation. *IEEE/ASME Trans. Mechatron.* **30**, 5628–5639 (2025).
48. B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid, Q. Vuong, V. Vanhoucke, H. Tran, R. Soricut, A. Singh, J. Singh, P. Sermanet, P. R. Sanketi, G. Salazar, M. S. Ryoo, K. Reymann, K. Rao, K. Pertsch, I. Mordatch, H. Michalewski, Y. Lu, S. Levine, L. Lee, T.-W. E. Lee, I. Leal, Y. Kuang, D. Kalashnikov, R. Julian, N. J. Joshi, A. Irpan, B. Ichter, J. Hsu, A. Herzog, K. Hausman, K. Gopalakrishnan, C. Fu, P. Florence, C. Finn, K. A. Dubey, D. Driess, T. Ding, K. M. Choromanski, X. Chen, Y. Chebotar, J. Carbajal, N. Brown, A. Brohan, M. G. Arenas, K. Han, "RT-2: Vision-language-action models transfer web knowledge to robotic control" in *Proceedings of the 7th Conference on Robot Learning*, J. Tan, M. Toussaint, K. Darvish, Eds., vol. 229 of *Proceedings of Machine Learning Research* (PMLR, 2023), pp. 2165–2183.
49. H. Zhou, Z. Bing, X. Yao, X. Su, C. Yang, K. Huang, A. Knoll, Language-conditioned imitation learning with base skill priors under unstructured data. *IEEE Robot. Autom. Lett.* **9**, 9805–9812 (2024).
50. A. O'Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, Mandlekar, A. Jain, A. Tung, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, Gupta, A. Wang, A. Singh, A. Garg, A. Kembhavi, A. Xie, A. Brohan, A. Raffin, A. Sharma, Yavary, A. Jain, A. Balakrishna, A. Wahid, B. Burgess-Limerick, B. Kim, B. Scho¨lkopf, Wulfe, B. Ichter, C. Lu, C. Xu, C. Le, C. Finn, C. Wang, C. Xu, C. Chi, C. Huang, Chan, C. Agia, C. Pan, C. Fu, C. Devin, D. Xu, D. Morton, D. Driess, D. Chen, D. Pathak, Shah, D. Bu¨chler, D. Jayaraman, D. Kalashnikov, D. Sadigh, E. Johns, E. Foster, F. Liu, F. Ceola, F. Xia, F. Zhao, F. Stulp, G. Zhou, G. S. Sukhatme, G. Salhotra, G. Yan, G. Feng, G. Schiavi, G. Berseth, G. Kahn, G. Wang, H. Su, H.-S. Fang, H. Shi, H. Bao, H. Ben Amor, H. I. Christensen, H. Furuta, H. Walke, H. Fang, H. Ha, I. Mordatch, I. Radosavovic, I. Leal, J. Liang, J. Abou-Chakra, J. Kim, J. Drake, J. Peters, J. Schneider, J. Hsu, J. Bohg, J. Bingham, J. Wu, J. Gao, J. Hu, J. Wu, J. Wu, J. Sun, J. Luo, J. Gu, J. Tan, J. Oh, J. Wu, J. Lu, J. Yang, J. Malik, J. Silve'rio, J. Hejna, J. Booher, J. Tompson, J. Yang, J. Salvador, J. J. Lim, J. Han, K. Wang, K. Rao, K. Pertsch, K. Hausman, K. Go, K. Gopalakrishnan, K. Goldberg, K. Byrne, K. Oslund, K. Kawaharazuka, K. Black, K. Lin, K. Zhang, K. Ehsani, K. Lekkala, K. Ellis, K. Rana, K. Srinivasan, K. Fang, K. P. Singh, K.-H. Zeng, K. Hatch, K. Hsu, L. Itti, L. Y. Chen, L. Pinto, L. Fei-Fei, L. Tan, L. J. Fan, L. Ott, L. Lee, L. Weihs, M. Chen, M. Lepert, M. Memmel, M. Tomizuka, M. Itkina, M. G. Castro, M. Spero, M. Du, M. Ahn, M. C. Yip, M. Zhang, M. Ding, M. Heo, M. K. Srirama, M. Sharma, M. J. Kim, N. Kanazawa, N. Hansen, N. Heess, N. J. Joshi, N. Suenderhauf, N. Liu, N. Di Palo, N. M. M. Shafiullah, O. Mees, O. Kroemer, O. Bastani, P. R. Sanketi, P. T. Miller, P. Yin, P. Wohlhart, P. Xu, P. D. Fagan, P. Mitrano, P. Sermanet, P. Abbeel, P. Sundaresan, Q. Chen, Q. Vuong, R. Rafailov, R. Tian, R. Doshi, R. Martín-Martín, R. Baijal, R. Scalise, R. Hendrix, R. Lin, R. Qian, R. Zhang, R. Mendonca, R. Shah, R. Hoque, R. Julian, S. Bustamante, S. Kirmani, S. Levine, S. Lin, S. Moore, S. Bahl, S. Dass, S. Sonawani, S. Song, S. Xu, S. Haldar, S. Karamcheti, S. Adebola, S. Guist, S. Nasiriany, S. Schaal, S. Welker, S. Tian, S. Ramamoorthy, S. Dasari, S. Belkhale, S. Park, S. Nair, S. Mirchandani, T. Osa, T. Gupta, T. Harada, T. Matsushima, T. Xiao, T. Kollar, T. Yu, T. Ding, T. Davchev, T. Z. Zhao, T. Armstrong, T. Darrell, T. Chung, V. Jain, V. Vanhoucke, W. Zhan, W. Zhou, W. Burgard, X. Chen, X. Wang, X. Zhu, X. Geng, X. Liu, X. Liangwei, X. Li, Y. Lu, Y. J. Ma, Y. Kim, Y. Chebotar, Y. Zhou, Y. Zhu, Y. Wu, Y. Xu, Y. Wang, Y. Bisk, Y. Cho, Y. Lee, Y. Cui, Y. Cao, Y.-H. Wu, Y. Tang, Y. Zhu, Y. Zhang, Y. Jiang, Y. Li, Y. Li, Y. Iwasawa, Y. Matsuo, Z. Ma, Z. Xu, Z. J. Cui, Z. Zhang, Z. Lin, "Open X-Embodiment: Robotic learning datasets and RT-X models: Open X-Embodiment collaboration" in *2024 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2024), pp. 6892–6903.
51. D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, C. Xu, J. Luo, T. Kreiman, Y. Tan, P. Sanketi, Q. Vuong, T. Xiao, D. Sadigh, C. Finn, S. Levine, "Octo: An open-source generalist robot policy" in *Proceedings of Robotics: Science and Systems XX*, D. Kulic, G. Venture, K. Bekris, E. Coronado, Eds. (RSS Foundation, 2024); 10.15607/RSS.2024.XX.090.
52. M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, Q. Vuong, T. Kollar, B. Burchfiel, R. Tedrake, D. Sadigh, S. Levine, P. Liang, C. Finn, "OpenVLA: An open-source vision-language-action model" in *Proceedings of the 8th Conference on Robot Learning Proceedings of the 8th Conference on Robot Learning*, P. Agrawal, O. Kroemer, W. Burgard, Eds., vol. 270 of *Proceedings of Machine Learning Research* (PMLR, 2025), pp. 2679–2713.
53. A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, "Learning transferable visual models from natural language supervision" in *Proceedings of the 38th International Conference on Machine Learning*, M. Meila, T. Zhang, Eds., vol. 139 of *Proceedings of Machine Learning Research* (PMLR, 2021), pp. 8748–8763.
54. University Robots, UR5 technical specifications; https://universal-robots.com/media/50588/ur5_en.pdf.
55. L. Wang, X. Chen, J. Zhao, K. He, "Scaling proprioceptive-visual learning with heterogeneous pre-trained transformers" in *Advances in Neural Information Processing Systems 37 NeurIPS 2024*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, C. Zhang, Eds. (Curran Associates, 2024), pp. 124420–124450.
56. D. He, C. Fang, Y. Wang, Y. Peng, Y. Wang, S.-C. Zhu, A mathematical formulation of AGI in the (*C*, *U*, *V*) framework. *Engineering* 10.1016/j.eng.2025.08.034 (2025).
57. P. Wei, Y. Liu, T. Shu, N. Zheng, S.-C. Zhu, "Where and why are they looking? Jointly inferring human attention and intentions in complex tasks" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (IEEE Computer Society, 2018), pp. 6801–6809.
58. C. L. Baker, R. Saxe, J. B. Tenenbaum, Action understanding as inverse planning. *Cognition* **113**, 329–349 (2009).
59. J. A. Sommerville, A. L. Woodward, A. Needham, Action experience alters 3-month-old infants' perception of others' actions. *Cognition* **96**, B1–B11 (2005).
60. A. D. Dragan, K. C. Lee, S. S. Srinivasa, "Legibility and predictability of robot motion" in *ACM/IEEE Int. Conf. Hum.-Robot Interact. (HRI)* (IEEE, 2013), pp. 301–308.
61. M. Zurek, A. Bobu, D. S. Brown, A. D. Dragan, "Situational confidence assistance for lifelong shared autonomy" in *2021 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2021), pp. 2783–2789.
62. C. Liu, J. B. Hamrick, J. F. Fisac, A. D. Dragan, J. K. Hedrick, S. S. Sastry, T. L. Griffiths, "Goal inference improves objective and perceived performance in human-robot collaboration" in *Proc. Int. Conf. Auton. Agents Multiagent Syst. (AAMAS)* (IFAAMAS, 2016), pp. 940–948.
63. K. M. Collins, I. Sucholutsky, U. Bhatt, K. Chandra, L. Wong, M. Lee, C. E. Zhang, T. Zhi-Xuan, M. Ho, V. Mansinghka, A. Weller, J. B. Tenenbaum, T. L. Griffiths, Building machines that learn and think with people. *Nat. Hum. Behav.* **8**, 1851–1863 (2024).
64. I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, Lerchner, "beta-VAE: Learning basic visual concepts with a constrained variational framework" poster presented at the *5th International Conference on Learning Representations* (ICLR 2017), Toulon, France, 24 to 26 April 2017; https://openreview.net/forum?id=Sy2fzU9gl.
65. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks. *Commun. ACM* **63**, 139–144 (2020).
66. C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, S. Song, Diffusion policy: Visuomotor policy learning via action diffusion. *Int. J. Robot. Res.* **44**, 1684–1704 (2025).
67. A. Salmona, V. De Bortoli, J. Delon, A. Desolneux, "Can push-forward generative models fit multimodal distributions?" in *Advances in Neural Information Processing Systems 35 NeurIPS 2022*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh, Eds. (Curran Associates, 2022), pp. 10766–10779.
68. R. Bellman, A Markovian decision process. *J. Math. Mech.* **6**, 679–684 (1957).
69. M. Shridhar, L. Manuelli, D. Fox, "CLIPort: What and where pathways for robotic manipulation" in *Proceedings of the 5th Conference on Robot Learning*, A. Faust, D. Hsu, G. Neumann, Eds., vol. 164 of *Proceedings of Machine Learning Research* (PMLR, 2022), pp. 894–906.
70. L. Downs, A. Francis, N. Koenig, B. Kinman, R. Hickman, K. Reymann, T. B. McHugh, V. Vanhoucke, "Google scanned objects: A high-quality dataset of 3D scanned household items" in *2022 IEEE International Conference on Robotics and Automation (ICRA)* (2022), pp. 2553–2560.
71. E. Perez, F. Strub, H. de Vries, V. Dumoulin, A. Courville, FiLM: Visual reasoning with a general conditioning layer. *Proc. AAAI Conf. Artif. Intell.* **32**, 3942–3951 (2018).
72. S. H. Vemprala, R. Bonatti, A. Bucker, A. Kapoor, ChatGPT for robotics: Design principles and model abilities. *IEEE Access* **12**, 55682–55696 (2024).

73. P. Vijayaraghavan, J. F. Queißer, S. V. Flores, J. Tani, Development of compositionality through interactive learning of language and action of robots. *Sci. Robot.* **10**, eadp0751 (2025).

Acknowledgments: The first authors thank their beloved son H. Gao for his support during the preparation of this manuscript. **Funding:** This work was supported by the National Key R&D Program of China (grant nos. 2024YFB4505500 and 2024YFB4505503), in part by the National Natural Science Foundation of China (grant no. 62376031), the Shenzhen Science and Technology Program (grant nos. JSGGKQTD20221101115656029 and ZDCY20250901104706008), and Guangdong Basic and Applied Basic Research Foundation (grant no. 2023B1515020089). **Author contributions:** X.C. conceived the methodology, designed and performed the simulation and real-world experiments, wrote the manuscript, and partially supervised the project. Y.G. defined the problem conceptualization and contributed to methodology idea discussions, designed and performed the real-world experiments, wrote the manuscript, and provided partial funding. H.L. and F.Y. contributed to the discussions and writing of the manuscript. A.G. contributed to the technical discussions. J.Y., B.L., and S.-C.Z. acquired funding and resources. C.Z. and T.L.L. contributed to the technical discussions and partially supervised the project. **Competing interests:** The authors declare that they have no competing interests. **Data, code, and materials availability:** All data needed to support the conclusions of this manuscript are included in the main text or Supplementary Materials. The evaluation scripts used in this study are available at https://zenodo.org/records/18618978. All materials used in the study are commercially available.

Submitted 10 December 2024
Accepted 23 February 2026
Published 18 March 2026
10.1126/scirobotics.adv2250

Science Robotics

Cross-robot behavior adaptation through intention alignment

Xi Chen, Yuan Gao, Hangxin Liu, Fangkai Yang, Ali Ghadirzadeh, Jun Yang, Bin Liang, Chongjie Zhang, Tin Lun Lam, and Song-Chun Zhu

Sci. Robot. **11** (112), eadv2250. DOI: 10.1126/scirobotics.adv2250

Science Robotics (ISSN 2470-9476) is published by the American Association for the Advancement of Science. 1200 New York Avenue NW, Washington, DC 20005. The title *Science Robotics* is a registered trademark of AAAS.