RHINO: Learning Real-Time Humanoid-Human-Object Interaction from Human Demonstrations

Jingxiao Chen*, Xinyao Li*, Jiahang Cao*, Zhengbang Zhu, Wentao Dong,
Minghuan Liu†, Ying Wen, Yong Yu, Liqing Zhang, Weinan Zhang
*Equal Contribution †Project Lead
Shanghai Jiao Tong University
humanoid-interaction.github.io

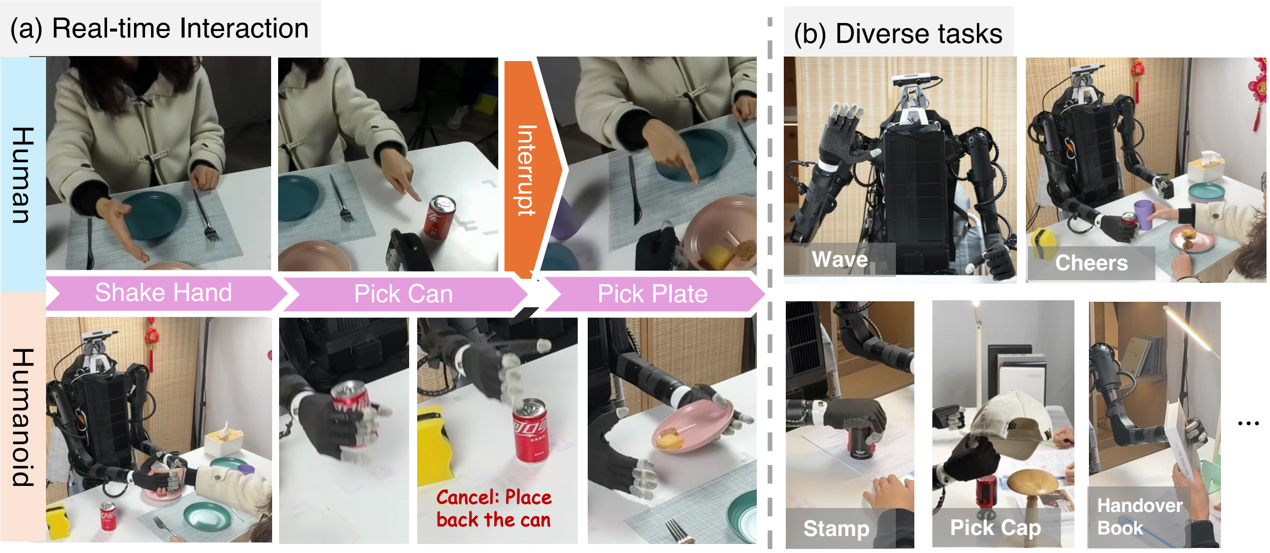


Fig. 1: **RHINO has the capabilities of real-time interaction on diverse tasks.** (a) RHINO enables real-time humanoid-human-object interaction, allowing seamless task interruption and dynamic switching during operation. (b) The system demonstrates diverse capabilities, including waving, cheering, stamping, object pickup, handovers, and more.

***Abstract*—Humanoid robots have shown success in locomotion and manipulation. Despite these basic abilities, humanoids are still required to quickly understand human instructions and react based on human interaction signals to become valuable assistants in human daily life. Unfortunately, most existing works only focus on multi-stage interactions, treating each task separately, and neglecting real-time feedback. In this work, we aim to empower humanoid robots with real-time reaction abilities to achieve various tasks, allowing human to interrupt robots at any time, and making robots respond to humans immediately. To support such abilities, we propose a general humanoid-human-object interaction framework, named RHINO, i.e., Real-time Humanoid-human Interaction and Object manipulation. RHINO provides a unified view of reactive motion, instruction-based manipulation, and safety concerns, over multiple human signal modalities, such as languages, images, and motions. RHINO is a hierarchical learning framework, enabling humanoids to learn reaction skills from human-human-object demonstrations and teleoperation data. In particular, it decouples the interaction process into two levels: 1) a high-level planner inferring human intentions from real-time human behaviors; and 2) a low-level controller achieving reactive motion behaviors and object manipulation skills based on the predicted intentions. We evaluate the proposed framework on a real humanoid robot and demonstrate its effectiveness, flexibility, and safety in various scenarios.**

I. Introduction

Humanoid robots are increasingly being explored to perform tasks in diverse environments [5, 17, 19]. Their human-like morphology provides a potential for acting with human-like dexterity, making them ideal for general-purpose daily-life human assistants. However, most recent progresses only focus on learning basic abilities such as locomotion [30], object manipulation [10], and expressive motion [9].

Considering how we as humans react to our friends, a practically helpful humanoid assistant should possess three fundamental capabilities: 1) skill proficiency, equipped with diverse and essential skills to achieve various tasks; 2) intention recognition, capable of discerning human intentions, from either motion or language; and 3) instant feedback, able to respond in real-time with feasible actions. Nonetheless, most studies on human-robot interaction only focus on only one or two of these aspects. For instance, a significant body of work on human-robot interaction focuses on object handover [32, 34], or interactive motion generation [28, 7, 21, 22, 25], lacking the ability to switch between different tasks in real-time. Some others focus on recognizing human intentions [12, 13, 14, 24, 31], which simplify the diversity of reaction and treat the interaction as an alternated two-stage process. The robot cannot be interrupted once a task is in progress, and further human commands can only be executed after the completion of the robot's current task. Many recent works have attempted to combine the ability of general foundation models to enable robots to understand the complexity of human interactions [33, 37], but they often suffer from high latency and are not suitable for real-time interaction tasks. These limitations hinder robots from rapid interventions and robust, multi-step interactions in human-centered tasks. Therefore, a unified framework that masters human-robot interaction with real-time intention recognition and various skills is urgently needed to tackle the above challenges.

To achieve this goal, we propose RHINO, a hierarchical learning framework for Reactive Humanoid-human INteraction and Object Manipulation. RHINO decouples the interaction process into two levels: a high-level planner that infers human intentions from real-time human behaviors, and a low-level controller that achieves reactive motion behaviors and object manipulation skills based on predicted intentions. The high-level planner updates at high frequency, and the low-level controller is designed to be interruptable, enabling it to react to high-level commands at any time. To ensure the scalability of RHINO across a wide range of skills, we design a pipeline for learning the interactions from human-object-human demonstration and teleoperation data, which can be easily extended to different tasks and scenarios. We implement RHINO on a real humanoid robot and demonstrate its effectiveness, flexibility, and safety in various scenarios (see Figure 1). Although this work only focuses on the upper body of a humanoid including the head, arms, and hands, it has the potential to be extended to whole-body humanoid interaction with a unified humanoid controller, and finally brings robots, especially humanoids, closer to our daily lives.

Our main contributions are in the following aspects:

- We propose the first humanoid learning architecture that seamlessly integrates intention recognition with real-time human-object-humanoid interaction skills, enabling the robot to respond to human instruction and switch between different tasks immediately.
- We design a pipeline for learning the interactions from human demonstrations, which can easily scale to different tasks and scenarios.
- We implement RHINO on the Unitree H1 humanoid robot and demonstrate its effectiveness, flexibility, and safety in 2 scenarios with over 20 tasks, and open-source the code and datasets to facilitate future research.

II. Related Works

Recent progress in building a human assistant robot can be divided into three categories: 1) recognition of human intention,n, 2) basic skills, and 3) unified interaction framework, as shown in Figure 2(b). We summarize related works in each category and highlight the differences between our work.

A. Human Intention Recognition.

Humanoid robots need to estimate the human physical and mental states to provide appropriate assistance [35]. More specifically, many signals can be used to infer human intentions, such as whole-body motion [34, 39], forces [5], gaze [12, 31], and language [33]. Object information in the environment also plays an important role in predicting human intention by combining it with human motion. Human-object interaction, such as pointing gestures [14] and grabbing objects [24], provides a broader semantic space for human actions. Most works on human intention recognition treat the interaction as a two-stage process, where the robot first predicts the human intention and then executes the task. This design simplifies the diversity of reactions and neglects the real-time reaction ability of the robot. Our work aims to react to human signals in real time, enabling the downstream tasks to be interrupted at any time.

B. Basic Skills

Interactive motion synthesis. In human-robot interaction (HRI), learning to generate interactive and expressive motions, such as shaking hands and waving, are fundamental skills. The human-like morphology of humanoid robots provides a unique opportunity to learn natural motion from retargeted human motion data [15]. Human motion data can be collected from motion capture systems or network videos. Compared to collecting robot motion data, it has a lower cost and higher scalability. Recent works [21, 38] collect multi-human motion data, capturing real-time interaction and reaction between humans. Building on this, studies encode social scenes [25], simulate reactions [22], or deploy interaction models on robots [28]. Our work focuses on learning interactive motion from human-human-object interaction data.

Object manipulation. The ability to manipulate objects is another fundamental skill for a humanoid assistant robot, which requires more precise control of the robot's end-effector. Limited by the dexterity of the robot, especially the degree of freedom of our humanoid robot's arm and hand, imitating learning from real-world teleoperation data [11, 26, 23] is a more practical way to ensure success, compared to learning from human data [36, 29, 42]. Open-Television [10] developed a teleoperation system with a VR device to control the arms and neck of humanoid robots and show the effectiveness and efficiency of learning manipulation skills with ACT [41] policy. Our work learns the manipulation skills based on the teleoperation data.

C. Unified Interaction Framework

Recent works have attempted to leverage the capacity of general foundation models, such as large language models (LLMs) or vison-language models (VLMs), to enable robots to understand human intention in the format of text-based

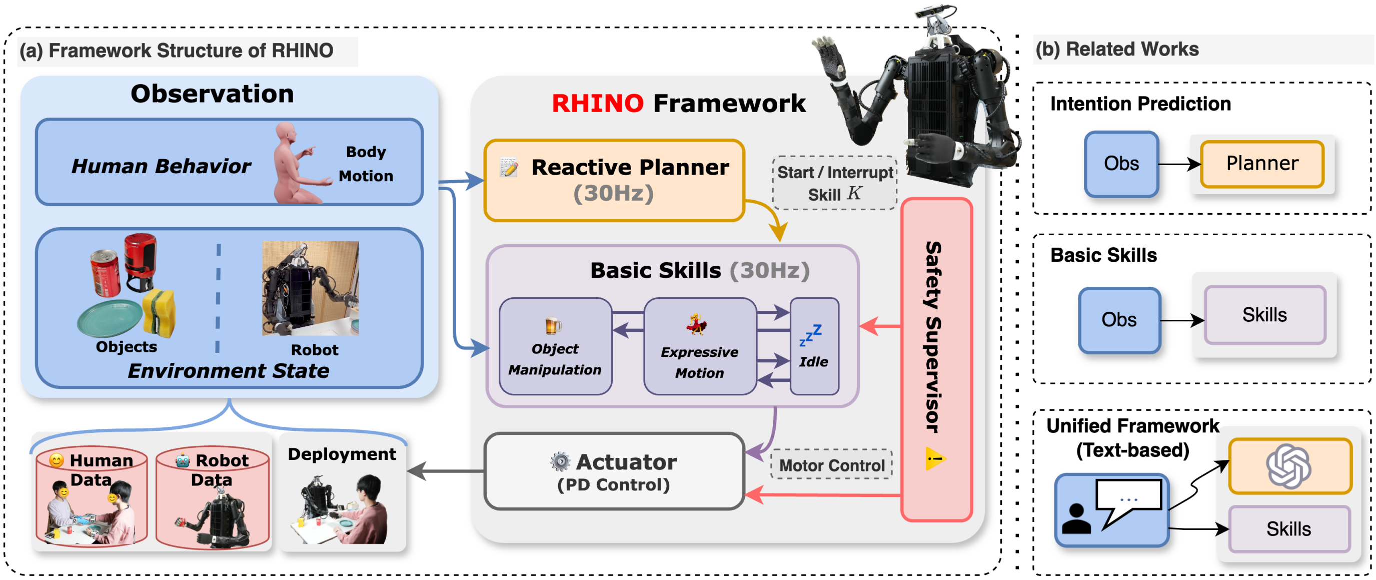


Fig. 2: **Components of the RHINO framework and related works.** (a) Illustration of the RHINO framework, including the reactive planner, motion generation skills, object interaction skills and safety supervisor. (b) Classification of related works based on the components of the RHINO framework.

instructions [33]. However, such interaction is often high-latency and not suitable for real-time environments, limiting the potential for natural and effective human-robot collaboration, particularly in scenarios that require immediate response or adaptation to changing human needs. Asfour et al. [6] designed rules of the real-time human-robot interaction, which is hard to scale up. Cardenas-Perez et al. [8] tries to learn an end-to-end model by imitation to achieve real-time interaction with 5 different tasks. Limited by the sample efficiency, this end-to-end paradigm makes it difficult to scale to more tasks.

Our framework decouples the interaction process and enables each module to model the interaction with different observation spaces, which is more sample-efficient and scalable. We also deploy the framework on a real humanoid robot and demonstrate its effectiveness, flexibility, and safety in two different scenarios and more than 20 tasks.

III. Problem Formulation

In this work, we consider the interaction as a leader-follower formulation [35], where the human is the leader and the humanoid robot is the follower. Define $\mathcal{I}$ as the set of human intentions and $\mathcal{K}$ as the set of robot skills. At time step t, the leader shows an intention $I_t \in \mathcal{I}$ for the follower to perform a skill $K_t \in \mathcal{K}$, such as picking up a can, brushing a plate, or stamping a file. We assume one intention corresponds to at most one skill, and the robot should be able to switch between skills in real time. The map function is defined as $f : \mathcal{I} \to \mathcal{K}$.

The skills of the robot can be categorized into three types: interactive motion, manipulation, and idle. The interactive motion skills require the robot to perform expressive and diverse behavior, and the manipulation skills require the robot to interact with objects in the environment precisely. When the human leader does not show any intention, the robot will be in an idle state and do nothing. The real-time interaction design requires the robot to respond to the leader's intentions with low latency and in-skill reflection to the human leader. Low latency requires the robot to predict the leader's intention in real time and interrupt the current skill when the leader shows a new intention. In-skill reflection requires the robot to react to human motion and environment even if the human intention is not changed, such as pausing current movement if the robot collides with the human or the target object is not reachable.

We formulate the observation space $\mathcal{O}$ of a humanoid robot as the combination of the environment state $\mathcal{E}$ and the human behavior $\mathcal{H}$, i.e., $\mathcal{O} = \mathcal{E} \bigodot \mathcal{H}$. The environment state $\mathcal{E}$ includes the robot's proprioception and the object state. The human behavior $\mathcal{H}$ consists of the leader's intention and the leader's behavior. To reduce the complexity of observation, our framework decomposes interaction policy into several sub-modules and separately designs the observation space for each module. Compared to end-to-end models, this decomposed design allows the robot to learn humanoid-human-object interaction from human-object-human demonstration data, which is more sample-efficient and scalable.

IV. RHINO Framework

In RHINO, the humanoid robot acts as the follower and learns to predict the human intention I with the reactive planner, then utilizes the corresponding skill K to finish the interaction. Those skills are classified into interactive motion, manipulation, and idle. Interactive motion skills, simply called motion skills, enable the robot to react to the leader's intentions with real-time motion. Manipulation skills enable the robot to handle objects based on the predicted intentions. Idle refers to the robot maintaining its joints in a default state. Figure 2 illustrates the framework of RHINO, and Figure 3 provides the detailed network architecture of each sub-module in our implementation.

A. Data Collection

Human-object-human interaction data. To learn the interaction between humans and robots, we first collect a dataset of human-object-human interaction [40], where two people perform a series of daily interaction tasks with various objects. In comparison to human-robot interaction data, human-object-human interaction data can be collected without a real robot, which is cheaper to collect and easier to scale to more skills in various of scenarios. The dataset contains interaction between two people in two different scenarios, dining and office. The dataset is recorded with a simple motion capture system, and a stereo RGB-D camera in the first-person view of the follower. The motion capture system that collects the follower's behavior is described in Appendix A. Motion data is retargeted to the humanoid robot and used by imitation learning algorithms to construct the reactive motion skills. The stereo RGB-D camera records the leader's behavior $\mathcal{H}$ and the environment state $\mathcal{E}$, which is used to predict the leader's intention I.

We label each frame t in the interaction data with the leader's intention I_t and the follower's skill $K^{(t)}$, which are represented as ID integers of intentions and skills. We add additional labels for the occupancy $p \in \mathcal{P}$ of the robot, indicating whether the end-effector (i.e., the hands) is empty or interacting with an object, with distinct labels assigned to different objects.

Teleoperation data. Different from the interactive motion skills, certain skills, such as picking up a cup, require more precise control of the robot's end-effector and the manipulation of some objects. To ensure the success of those skills, we collect demonstrations with a teleoperation system [10], where the human's motion is captured with a VR device, and the robot's joint positions are set by retargeting the human's motion. This system records the control commands, the robot's proprioception, and stereo videos from a camera on the robot's head to perceive the environment. We also label the frames where the skill showcase is completed successfully, referred to as the success signal, which is used to learn the finish condition of the manipulation skills.

Figure 4 shows information collected in the interaction and teleoperation dataset and the requirement of real robot deployment. More details are described in Appendix D.

B. Reactive Planner

The reactive planner is designed to infer the leader's intention I_t from the real-time observation O_t and decide the next skill $K^{(t+1)}$ of the robot. The planner is a Transformer model, which takes the human's motion and the environment information as input, and predicts the leader's intention at a 30Hz frequency. To enhance the generalization of the model, we do not input observed images directly, but extract the human's body and hand postures, the human's hand and head position, and the nearest object to hands from the RGB-D images, as well as the robot's hand occupancy p_t. We retarget human hand postures to a robot hand with 6 degrees of freedom (DoF) and represent the hand posture as the position of each joint.

There are two types of skills that correspond to the leader's intentions: motion skills and manipulation skills. Each skill K has a start condition $s_K \in \mathcal{P}$ and an end transition $e_K \in \mathcal{P}$. The start condition shows the required hand occupancy of each hand to start the skill. For example, the skill to cheer with the leader requires the humanoid to hold a can of drink in the right hand. The end transition determines the change of hand occupancy after finishing this skill successfully. For example, for the skill of picking up a can, the start condition and end transition are `[empty, empty]` and `[empty, can]` respectively. A comprehensive description of all skills is shown in Appendix B.

The humanoid robot starts from an idle state. If the reactive planner predicts a human intention I consistently for n_r time steps, the humanoid robot switches to the corresponding skill $T = f(I)$. The human intentions are meant to "start" the execution of a skill, rather than "keep" the current execution. For example, the leader only needs to point at the can for a while at the beginning to get the robot to pick the can on the table, instead of pointing all the time.

After a skill is initiated, the motion skill persists until a change in human intention occurs, while the manipulation skill persists until the execution is judged successful or exceeds a time limit. When a skill is complete, the robot returns to the idle state.

To enable low-latency interaction, the application of most of the skills can be interrupted by another skill when a different human intention lasts k_2 steps. The motion skills can be easily undone by immediately moving to an idle pose, while the interruption of the manipulation skills is complicated as it requires reversing the object state. We use a corresponding reverse skill to interrupt each interruptible manipulation skill. For example, the skill of placing the can is a reversal of the skill of picking the can. We show the detailed transitions between skills in Appendix D.

When the current occupancy is not satisfied with the start condition of a skill $p_t \neq s_K$, the skill is not able to start. To satisfy the requirement, we We build a directed graph of occupancy transition. The node $n \in \mathcal{P}$ is hand occupancy and the edge $e \in \mathcal{T}$ is skills. Before starting to demonstrate a skill K with an unsatisfied condition, we find the shortest path from current occupancy p_t to the start condition s_K, and execute the skill series $\{K_1, K_2, \ldots\}$ in order. After the operations mentioned above are done, the target skill K can be utilized. The occupancy graphs of 2 scenarios are shown in Appendix D.

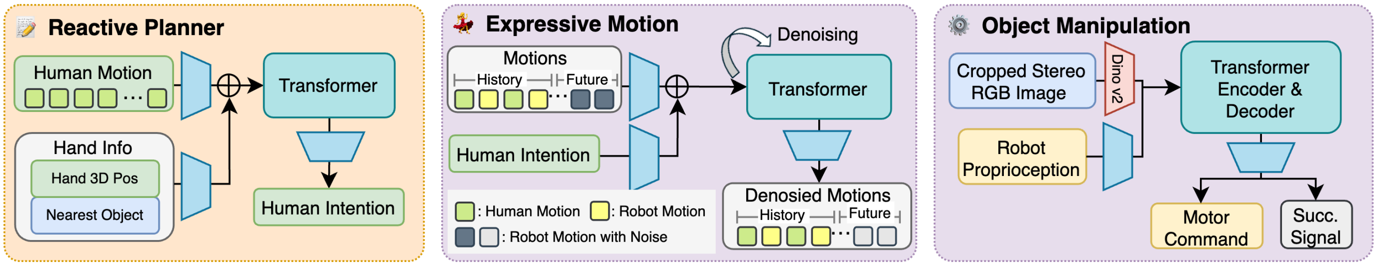


Fig. 3: **Network architecture of RHINO modules**, including the reactive planner, motion generation, and manipulation skills.

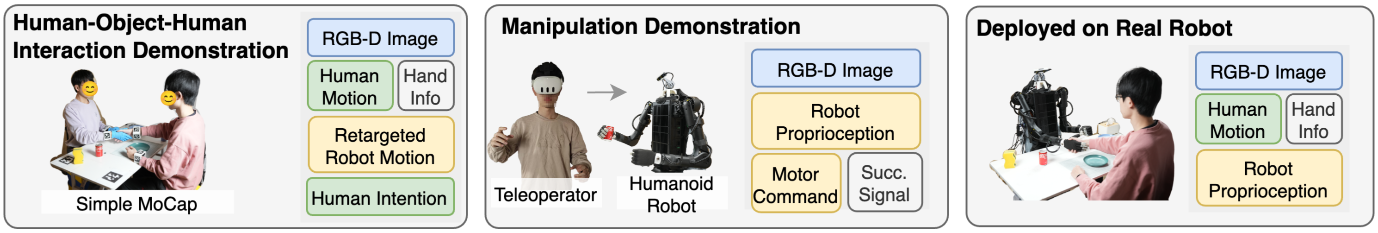


Fig. 4: **Information collected in the dataset and the requirement of real robot deployment.** The left box shows our Human-Object-Human Interaction Demonstration Data, which contains human motion and hand position information collected from RGB-D images and a simple motion capture system. The box in the middle illustrates our Manipulation Demonstration Data, which is collected through teleoperation. The right box shows the information required in real robot deployment.

C. Interactive Motion Skills

In humanoid-human interactions that do not involve complex object manipulation, the primary objective of the humanoid robot is to produce smooth, consistent motions while providing robust real-time feedback on human behavior. To accomplish this, we employ a multi-body motion diffusion model [20] to generate low-level interactive motion skills.

Different from multi-person motion generation, the humanoid and human are heterogeneous and asymmetric in the humanoid-human interaction. We represent the human motion m_t^1 as a 6D rotation vector for each joint, and the humanoid motion m_t^2 as the target of humanoid robot joint positions. Both motions are simplified to arm and hand joints. We also add hand occupancy p_t and human intention I_t as input to the model, to ensure the robot's motion is consistent with the human's intention. Details can be found in Appendix D.

Our model predicts the future motion of the humanoid robot $m_{t+1:t+5}^2$ based on the history of human motion $m_{t-30:t}^1$ and humanoid motion $m_{t-30:t}^2$. The model predicts 5 future frames of humanoid motion with a 3 Hz frequency, which generates 30 frames of motion in one second.

The network structure is a Transformer-based model, which takes the loss of reconstructing the humanoid motion as the main loss, and the velocity loss of motion as an auxiliary loss.

D. Manipulation Skills

Imitation learning. To enhance the smoothness and robustness of interactive motion reactions, the motion generation model described in Section IV-C omits RGB images as input. However, this design makes the model insufficient for dexterous object manipulation, resulting in a lower success rate in practice. Meanwhile, as mentioned earlier, the retargeting inevitably introduces deviations between the humanoid's end-effector poses and original human motions, further leading to manipulation failures.

As a result, we train independent Action Chunking Transformer (ACT) [41] models for each low-level manipulation skill. The ACT model enables 30Hz real-time inference of the robot's joint positions. Demonstrations collected by teleoperation are manually segmented and labeled as distinct skills for model training. To satisfy low-latency requirements, we trained a paired reverse skill model for each manipulation skill, enabling the robot to handle human interruptions properly.

Learning terminal conditions. In our multi-skill interactive manipulation framework, the model must recognize when a current skill is completed in order to transition to the next skill. In addition to desired humanoid joint positions, each manipulation policy predicts an additional success signal. The signal is an indicator of whether the skill is completed, and we add an extra cross-entropy loss to train the 0/1 classification.

Robust and safe manipulation. We crop the image to the region of robot-object interaction as the manipulation model input. The cropped image input removes the leader human's body and only keeps the hand information, which helps the model focus on the manipulation skill and be robust to human appearance and behavior changes. For skills with a single arm, the input/output of the model only includes the corresponding arm information. We also collect in-skill interruption data, where the robot pauses or withdraws its current movement if it collides with the human or the target object is unreachable. Such data helps the robot to exhibit safe behavior and in-skill reflection to the change of human behavior or environment, even if the human intention is not changed.

E. Safety Supervisor

A safety supervisor serves as a global guarantee of safe robot actions, which forces the robot to pause immediately when potential harm is detected. In this module, the collision box of the robot is calculated based on several selected key points on the arms. They are updated by forward kinematics as the movement of the arm. Meanwhile, global coordinates of human hands are obtained by the depth camera.

The safety module judges whether the robot collision box is to collide with human hands. In case the distance between them is too close, the safe supervisor sends an unsafe signal to halt the robot at the current pose until the distance recovers into a safe range. We use the Euclidean distance from human hand key points to robot arm key points as a simple but effective approach to calculating the collision box.

The safe supervisor takes strong aids to ensure the robot would not hurt humans by avoiding collision, especially in skills where they should contact at a rather close distance such as handshake and handover the plate.

F. Real Robot Platform

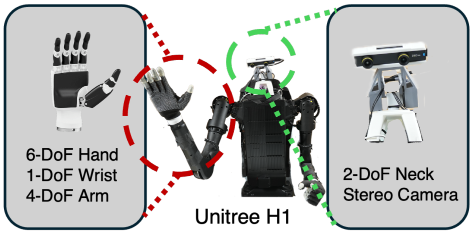


Fig. 5: The humanoid robot platform.

We implement RHINO on a Unitree H1 humanoid robot with an active head, equipped with a RGB-D stereo camera ZED mini, illustrated in Figure 5. The human body posture is detected from a body detection model in ZED API with 38 joints and represented as 6D rotation vectors of joints. We capture the leader human's intention and the state of objects from the RGB-D images. The hand posture is detected by a hand reconstruction model, HaMeR [27]. We record the information of nearest objects to hands, including the object ID, distance to the hand, and the IOU of the object bounding box and the hand bounding box. The 3D bounding box of objects and hands is detected from a stereo RGB-D camera, ZED-mini, with a fine-tuned YoloV11 [18] model and a body detection model in ZED API. The RGB-D stereo camera enables retrieval of the 3D position of any 2D pixel in the image, represented in world coordinates. The orientation of the humanoid's body determines the X and Y axes, while gravity defines the Z axis.

V. Experiment

In experiments, we evaluate the performance of RHINO on two different scenarios: a dining waiter scenario and an office assistant scenario, also called Scene 1 and Scene 2. The details of the skills in each scenario are shown in Appendix B. The robot should react with its arms, hands, and active head. We first evaluate the performance of each module in RHINO, including human intention prediction, motion generation, and manipulation skills. To show the effectiveness of our framework, we compare it with end-to-end models on different numbers of skills. The results show that all modules in RHINO perform well in the skills. The framework outperforms the end-to-end models and is more robust to out-of-distribution data. We also analyze the failure of the system and show the crucial challenges in the real-world deployment of humanoid robots.

A. Framework Performance

1) Human Intention Prediction

We evaluate the performance of our human intention prediction module by calculating the mAP score on datasets. We split the training data into 80% for training and 20% for validation, and also collected a test set under the robot-human-interaction deployment setting. The test set contains 3 different people, each of whom performs all intentions of 2 scenarios, which ensures the diversity of the test set. Table I shows that although deploying the model to the real world leads to a decrease in performance, but our model still outperforms all baselines.

Our method takes both *human motion* and *hand details* including hand 3D position and object nearest to hand as input. Compared to the baseline that only inputs human motion, our method performs better in our test set, demonstrating that *hand details* is necessary information to differentiate between different human intentions. We also test the performance of VLMs on human intention prediction. Qwen2-VL-2B-Instruct can infer at a frequency of 30Hz, but its performance is poor even after being finetuned on our training set, probably due to a relatively small amount of training data. While GPT-4o-mini can perform quite well on our test set, it takes too long for an inference which leads to a slow reaction of the robot. When calculating mAP for the VLMs, we assume that the probability of the class output by VLM is 1, while the probabilities of all other classes are 0.

2) Motion Generation

We compare our motion generation module with three baselines on all the motions involved in our skills (*handshake*, *wave*, *cheers*, *thumbup*, *spread hand*, *take photo*). The baselines are:

- **Zero Velocity**: the repetition of the last pose observed, as a simplest baseline.
- **Ours (w/o diffusion)**: Generate the motions directly with a Transformer model with the same structure as the denoiser used in our Diffusion-based method, without the full diffusion framework.
- **Ours (w/o human motion)**: Generate humanoid motions only conditioning on the human intention label and history of humanoid motions, without the guidance of detailed human motion.

The metrics to evaluate the performance of the motion generation modules are:

TABLE I: **Performance of the interaction planner.**

Method	mAP ↑				Inference Frequency ↑
	Validation Data		Test Data		
	Scene 1	Scene 2	Scene 1	Scene 2	
Ours	0.982	**1.0**	**0.787**	**0.643**	**30 Hz**
Ours (w/o hand details)	**0.999**	0.925	0.729	0.587	**30 Hz**
Qwen2-VL-2B-In.	-	-	0.213	0.167	**30 Hz**
Finetuned Qwen2-VL-2B-In.	0.284	0.322	0.228	0.159	**30 Hz**
ChatGPT-4o-mini	-	-	0.573	0.564	≈ 0.46 Hz

TABLE II: **Performance of motion generation.**

Method	FID↓	JPE(mm)↓	Diversity	MModality↑
Real	-	-	3.74 ±0.05	-
Zero Velocity	43.22 ±0.01	84.82 ±0.01	2.85 ±0.09	-
Ours	**10.67 ±0.01**	**48.79 ±0.00**	**3.68 ±0.06**	0.02 ±0.00
Ours (w/o diffusion)	38.50 ±0.01	142.85 ±0.02	2.91 ±0.04	-
Ours (w/o human motion)	17.34 ±0.05	60.52 ±0.04	3.59 ±0.08	**0.06 ±0.01**

- **FID**: The FID score [16] is leveraged to assess the similarity between synthesized and real motions quantitatively.
- **JPE**: We calculate the Joint Position Error (JPE) based on the forward kinematic results of the generated robot joint 3D position to measure the poses of all the individuals. JPE is averaged over all hand joints and finger joints.
- **Diversity**: We calculate the average Euclidean distances of 300 randomly sampled pairs of motions in latent space to measure motion diversity in the generated motion dataset. The Diversity of motions generated by the model is expected to be closer to the Diversity of Real Data.
- **MModality**: MModality captures the ability of the model to generate diverse motions for the same human intention label and human motion sample. We sample 20 motions within one fixed human intention label and one fixed sample of human motion to form 10 pairs, and measure the average latent Euclidean distances of the pairs. The average overall human intention and human motion pairs are reported.

Table II shows the performance of the motion generation module, where RHINO significantly outperforms all baselines on FID and JPE, demonstrating better quality of generated motions. The baseline without the diffusion process gets the lowest score, indicating that the sampling process of the diffusion model helps to synthesize better motions. Diversity also matters since different people may react differently to the same motion, and we expect our robot to obtain diverse behavior as well. As is shown in Table II, our approach outperforms the baseline without diffusion in terms of Diversity, thanks to the stochasticity introduced by the diffusion process and the ability of diffusion models to fit high-dimensional distributions. The baseline model without human motion inputs gets a higher MModality score, because it may generate quite different motions given only a human intention label. Yet it performs poorer on FID and JPE, since human intention alone can not guide the model to generate expressive motions with accurate and desired reactive meaning. For instance, the policy may give various angles and poses of a stretched hand given only the intention `Shake Hands`; in contrast, given the exact human pose and hand positions, our model can stretch the hand to the exact position and shake hand with the human, highlighting its real-time reactive capability.

3) Objects Manipulation

We further test the manipulation performance, and collect the results in Table III, comparing with statistics recorded from human teleoperation. We compute the success rate and the averaged time based on 20 independent tests, on the main object in two different scenes. The detailed statistics of each certain skill can be found in Appendix E.

Our manipulation module shows good performance aligned with human teleoperators. As only success data of human teleoperation is used in training, the module even outperforms humans by success rate on groups with rather simple motions such as *can*, *tissue*, *book* and *lamp*. In skills requiring a more delicate operation, the trained model slightly falls behind. We mainly consequence the fault for the heterogeneity between humans and the robot. The inadequate DoFs of the robot arms and lack of haptic sensing on the dexterous hand add great difficulty to some of the skills. To be specific, the former would lead to the failure of a smooth trajectory, ending with the cap sticking on the hatstand (in group *cap*), and the latter makes it challenging for the robot to determine whether the stamp is pressed to a fair location (in group *stamp*).

It is also noteworthy that the average operation time of our manipulation module is slightly longer on most of the skills than that of human data. This is because (1) some skills have a periodical motion, making the progress predictor output ending progress value later than the ground truth. (2) the robot would slowly move back to a fair initialized joint position range at the start to avoid violent actions, this adds to lags as a trade-off.

To determine the effectiveness of the in-skill interruption data, we analyze the impact of training data with human disturbance on three typical manipulation motions. With a fixed amount of total training data, 1%, 10%, or 20% of them are replaced by data where the robot motion is disturbed. Take the skill *Pick Can* as an example: the robot finds a human is looting the can when it intends to pick, then the robot should withdraw its motion and try to pick again when the human returns the can.

The success rate of a proper motion is shown in Table IV. The result shows that with an increasing ratio of disturbance data, the success rate of dealing with human violations is getting higher. Models with few disturbance data are not capable of a withdraw action, and only mixing as a ratio of 20% could make a success rate of 85% in simple skills.

4) Framework Structure

To highlight the accessibility to multiple skills of our **two-level** framework, we also compare end-to-end (E2E) baselines, similar to the setting of Cardenas-Perez et al. [8]. We implement an ACT policy [41, 10], leaving the input image not cropped to capture the human intention and robot motion information correctly. As a concise case, the baselines are trained on data of only 1 motions (*cheers*), 3 skills (along with *pick*, *place*, meaning pick or place the can) and 5 skills (along with *handshake*, *wave* also) in a simple scene, and are named

TABLE III: **Performance of manipulation across objects.**

Metrics	**Scene 1**				**Scene 2**			
	Can	***Plate***	***Sponge***	**Tissue**	***Cap***	***Book***	***Stamp***	***Lamp***
Success Rate	1.00	0.96	0.90	0.95	0.93	0.95	0.93	1.00
Average Time	9.41	29.59	23.69	9.43	16.14	10.81	15.17	5.06
Success Rate (Human)	0.97	0.98	0.99	0.91	0.91	0.93	0.92	0.96
Average Time (Human)	10.42	25.77	17.04	9.54	18.98	10.21	11.84	3.53

TABLE IV: **Success rate of manipulation** with different ratios of interrupted data.

Ratio of data with interruption	**Pick Can**	**Stamp the Paper**	**Place Plate to Stack**
1%	0.00	0.00	0.00
10%	0.05	0.15	0.30
20%	0.85	0.60	0.90

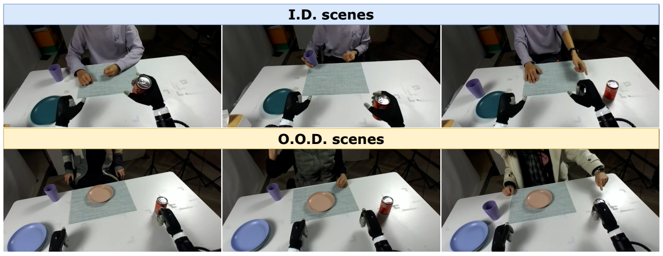


Fig. 6: **Examples of I.D. and O.O.D. scenes comparing RHINO with the End-to-End Model from the robot camera view.** The first row (I.D. scenes) shows the leader human wearing consistent clothing, with only minor adjustments in the arrangement of objects on the table. The second row (O.O.D. scenes) presents leader humans in different clothing and entirely different object arrangements, representing diverse scenarios encountered during deployment.

TABLE V: **Success rates of RHINO and end-to-end model** with a different number of skills.

Number of tasks	**1 skill**	**3 skills**	**5 skills**
Ours	**1.00**	**0.95**	**0.97**
End2End (I.D. Scene)	**1.00**	0.70	0.84
End2End (O.O.D. Scene)	0.95	0.57	0.67

E2E/1, E2E/3 and E2E/5 separately. In dataset collection, 100 slices for each skill are collected separately, the same as an average number of that in the manipulation skills. In deployment, we test the E2E model on the in-distribution (I.D.) scene in which human clothing and object arrangement are the same as the recorded datasets, and in the out-of-distribution (O.O.D.) scene these conditions vary.

In detail, the I.D. scene features a leader human wearing a light purple shirt, with a Coke can placed to the right of the robot, and a dark green plate on the table. The O.O.D. scenes include, but are not limited to, the following changes: replacing the Coke can (red) with a Sprite can (green), swapping the dark green plate for a light red or light blue one, adding more plates in front of the leader human, changing the leader human's outfit, for example, to a down jacket, or replacing the leader human with another character, such as one in a striped sweater or a white coat. Figure 6 shows an example highlighting the scene difference with respect to the leader human.

The success rates on average of different numbers of skills are shown in Table V, where we can observe RHINO outperforms the E2E baselines on the skills of better prediction of human intention and robustness to O.O.D. data. When trained with data of only one skill, the E2E framework is able to predict human intention and perform a successful motion. However, when more different skills are used, the E2E framework struggles to predict the correct intention. Meanwhile, without the information on hand occupancy, the model is likely to fail to tell the difference among skills with similar camera view, i.e. *cheers*, *pick*, and *place*. Moreover, the coupling of prediction, generation, and manipulation leads to uncropped images as input. The high dimension of images and noise cripple the robustness of the prediction module. Interacting with O.O.D. human body and clothing, or under some O.O.D. table arrangement, the model fails to generate proper motion to finish the skill showcase. The detailed success rate is shown in Appendix E. .

B. Analysis of System Failure

As a framework of multiple modules, the failure of the system could be caused by various reasons.

Error and limitation of sensors. Most of the perception of our implementation of RHINO is based on one RGB-D camera. However, the estimation of 3D position often shifts with time and missing when the estimated object is occluded by other objects or the robot arms. The cumulative error of the sensors leads to a misunderstanding of human intention and incorrect judgment of the safety supervisor.

Stability of hardware. The zero position of the robot arm may have shifted in a small range, which leads to the incorrect proprioception. Also, the robot's electronics age over time, which causes errors that require precise control of the robot's end-effector.

Failure of Model Generalization. Due to the limited data collection, the model may fail to generalize to extreme out-of-distribution scenarios, although we mitigate this issue by cropping the image to the region of interest for manipulation skills and using extracted information rather than raw images for human intention prediction. Some unseen human clothing or unexpected object arrangement may lead to the failure of the manipulation. Also, non-standard sitting posture or body shape can also have an impact on the prediction of human posture and intention, which leads to misunderstanding of human intention. Fortunately, human leaders can intervene to correct the robot's behavior in RHINO, which helps prevent a complete breakdown of the system.

VI. Conclusion and Limitations

In this work, we presented RHINO, a hierarchical learning framework designed to enable humanoid robots to engage in real-time humanoid-human-object interactions. By decoupling the interaction process into high-level planning and low-level reactive control, RHINO allows humanoid robots to quickly adapt to the change of human intentions and interrupt ongoing tasks without delays. This framework incorporates a wide variety of skills, from object manipulation to expressive motion generation. We implemented RHINO on a real humanoid robot and demonstrated its effectiveness, flexibility, and safety in various dynamic environments.

The proposed framework provides a significant step toward making humanoid robots autonomous and responsive in real-world applications, such as assisting with daily life tasks, disaster response, and industrial automation. By allowing continuous humanoid-human-object interaction, RHINO provide immediate and adaptive responses, making humanoid robots suited for seamless integration into human environments.

Despite promising results, several limitations remain. First, while RHINO is designed to be scalable, the current implementation is constrained by the availability of high-quality training data. The generalization of the system across a broader range of tasks and environments is still a challenge, as it heavily relies on human demonstrations and teleoperation data, which is time-consuming to collect. Future work will focus on utilizing existing datasets and simulation environments to improve the scalability and generalization of the framework. Additionally, the current implementation of RHINO is limited to the upper body at a fixed workspace, but a humanoid assistant should have locomotion and navigation abilities in a dynamic environment, and react with whole-body behaviors. Future work should integrate a whole-body controller to extend the framework to whole-body interaction for humanoid robots, and more general tasks with varying levels of human intervention.

References

[1] Universal humanoid robot H1_Bipedal Robot_Humanoid Intelligent Robot Company | Unitree Robotics. https://www.unitree.com/h1/.

[2] The Dexterous Hands. https://inspire-robots.store/collections/the-dexterous-hands.

[3] DYNAMIXEL XL330-M288-T. https://www.robotis.us/dynamixel-xl330-m288-t/.

[4] ZED Mini Stereo Camera | Stereolabs. https://www.stereolabs.com/store/products/zed-mini.

[5] Don Joven Agravante, Andrea Cherubini, Alexander Sherikov, Pierre-Brice Wieber, and Abderrahmane Kheddar. Human-humanoid collaborative carrying. *IEEE Transactions on Robotics*, 35(4):833–846, 2019.

[6] Tamim Asfour, Fabian Paus, Mirko Waechter, Lukas Kaul, Samuel Rader, Pascal Weiner, Simon Ottenhaus, Raphael Grimm, You Zhou, and Markus Grotz. ARMAR-6: A High-Performance Humanoid for Human-Robot Collaboration in Real-World Scenarios. *IEEE Robotics & Automation Magazine*, 26(4):108–121, December 2019. ISSN 1070-9932, 1558-223X. doi: 10.1109/MRA.2019.2941246.

[7] Judith Bütepage, Ali Ghadirzadeh, Özge Öztimur Karadag, Mårten Björkman, and Danica Kragic. Imitating by generating: Deep generative models for imitation of interactive tasks, October 2019.

[8] Carlos Cardenas-Perez, Giulio Romualdi, Mohamed Elobaid, Stefano Dafarra, Giuseppe L'Erario, Silvio Traversaro, Pietro Morerio, Alessio Del Bue, and Daniele Pucci. Xbg: End-to-end imitation learning for autonomous behaviour in human-robot interaction and collaboration. *IEEE Robotics and Automation Letters*, 2024.

[9] Xuxin Cheng, Yandong Ji, Junming Chen, Ruihan Yang, Ge Yang, and Xiaolong Wang. Expressive whole-body control for humanoid robots. *arXiv preprint arXiv:2402.16796*, 2024.

[10] Xuxin Cheng, Jialong Li, Shiqi Yang, Ge Yang, and Xiaolong Wang. Open-television: Teleoperation with immersive active visual feedback. *arXiv preprint arXiv:2407.01512*, 2024.

[11] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023.

[12] Nuno Ferreira Duarte, Jovica Tasevski, Moreno Coco, Mirko Raković, Aude Billard, and José Santos-Victor. Action Anticipation: Reading the Intentions of Humans and Robots. *IEEE Robotics and Automation Letters*, 3(4):4132–4139, October 2018. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2018.2861569.

[13] Sadman Sakib Enan, Michael Fulton, and Junaed Sattar. Robotic Detection of a Human-Comprehensible Gestural Language for Underwater Multi-Human-Robot Collaboration, July 2022.

[14] Irving Fang, Yuzhong Chen, Yifan Wang, Jianghan Zhang, Qiushi Zhang, Jiali Xu, Xibo He, Weibo Gao, Hao Su, Yiming Li, and Chen Feng. EgoPAT3Dv2: Predicting 3D Action Target from 2D Egocentric Vision for Human-Robot Interaction, March 2024.

[15] Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. HumanPlus: Humanoid Shadowing and Imitation from Humans, June 2024.

[16] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[17] Matthew Johnson, Brandon Shrewsbury, Sylvain Bertrand, Tingfan Wu, Daniel Duran, Marshall Floyd, Peter Abeles, Douglas Stephen, Nathan Mertins, Alex Lesman, John Carff, William Rifenburgh, Pushyami Kaveti, Wessel Straatman, Jesper Smith, Maarten Griffioen, Brooke Layton, Tomas de Boer, Twan Koolen, Peter Neuhaus, and Jerry Pratt. Team ihmc's lessons learned from the darpa robotics challenge trials. *Journal of Field Robotics*, 32(2):192–208, 2015. doi: https://doi.org/10.1002/rob.21571. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21571.

[18] Rahima Khanam and Muhammad Hussain. Yolov11: An overview of the key architectural enhancements. *arXiv preprint arXiv:2410.17725*, 2024.

[19] Abderrahmane Kheddar, Stephane Caron, Pierre Gergondet, Andrew Comport, Arnaud Tanguy, Christian Ott, Bernd Henze, George Mesesan, Johannes Englsberger, Máximo A. Roa, Pierre-Brice Wieber, François Chaumette, Fabien Spindler, Giuseppe Oriolo, Leonardo Lanari, Adrien Escande, Kevin Chappellet, Fumio Kanehiro, and Patrice Rabaté. Humanoid robots in aircraft manufacturing: The airbus use cases. *IEEE Robotics & Automation Magazine*, 26(4):30–45, 2019. doi: 10.1109/MRA.2019.2943395.

[20] Han Liang, Wenqian Zhang, Wenxuan Li, Jingyi Yu, and Lan Xu. Intergen: Diffusion-based multi-human motion generation under complex interactions. *International Journal of Computer Vision*, pages 1–21, 2024.

[21] Han Liang, Wenqian Zhang, Wenxuan Li, Jingyi Yu, and Lan Xu. InterGen: Diffusion-based Multi-human Motion Generation under Complex Interactions. *International Journal of Computer Vision*, March 2024. ISSN 0920-5691, 1573-1405. doi: 10.1007/s11263-024-02042-6.

[22] Yunze Liu, Changxi Chen, Chenjing Ding, and Li Yi. PhysReaction: Physically Plausible Real-Time Humanoid Reaction Synthesis via Forward Dynamics Guided 4D Imitation, April 2024.

[23] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. *arXiv preprint arXiv:2108.03298*, 2021.

[24] Esteve Valls Mascaro, Daniel Sliwowski, and Dongheui Lee. HOI4ABOT: Human-Object Interaction Anticipation for Human Intention Reading Collaborative roBOTs. *arXiv preprint arXiv:2309.16524*, 2023. URL https://arxiv.org/abs/2309.16524.

[25] Esteve Valls Mascaro, Yashuai Yan, and Dongheui Lee. Robot Interaction Behavior Generation based on Social Motion Forecasting for Human-Robot Interaction, April 2024.

[26] Jyothish Pari, Nur Muhammad Shafiullah, Sridhar Pandian Arunachalam, and Lerrel Pinto. The surprising effectiveness of representation learning for visual imitation. *arXiv preprint arXiv:2112.01511*, 2021.

[27] Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3D with transformers. In *CVPR*, 2024.

[28] Vignesh Prasad, Alap Kshirsagar, Dorothea Koert, Ruth Stock-Homburg, Jan Peters, and Georgia Chalvatzaki. MoVEInt: Mixture of Variational Experts for Learning Human-Robot Interactions from Demonstrations. *IEEE Robotics and Automation Letters*, 9(7):6043–6050, July 2024. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2024.3396074.

[29] Yuzhe Qin, Wei Yang, Binghao Huang, Karl Van Wyk, Hao Su, Xiaolong Wang, Yu-Wei Chao, and Dieter Fox. Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system, 2024. URL https://arxiv.org/abs/2307.04577.

[30] Ilija Radosavovic, Tete Xiao, Bike Zhang, Trevor Darrell, Jitendra Malik, and Koushil Sreenath. Learning humanoid locomotion with transformers. *arXiv:2303.03381*, 2023.

[31] Lisa Scherf, Lisa Alina Gasche, Eya Chemangui, and Dorothea Koert. Are You Sure? - Multi-Modal Human Decision Uncertainty Detection in Human-Robot Interaction. In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, pages 621–629, Boulder CO USA, March 2024. ACM. ISBN 9798400703225. doi: 10.1145/3610977.3634926.

[32] Kyle Wayne Strabala, Min Kyung Lee, Anca Diana Dragan, Jodi Lee Forlizzi, Siddhartha Srinivasa, Maya Cakmak, and Vincenzo Micelli. Towards Seamless Human-Robot Handovers. *Journal of Human-Robot Interaction*, 2(1):112–132, March 2013. ISSN 21630364. doi: 10.5898/JHRI.2.1.Strabala.

[33] Daniel Tanneberg, Felix Ocker, Stephan Hasler, Joerg Deigmoeller, Anna Belardinelli, Chao Wang, Heiko Wersing, Bernhard Sendhoff, and Michael Gienger. To help or not to help: Llm-based attentive support for human-robot group interactions. *arXiv preprint arXiv:2403.12533*, 2024.

[34] Andreea Tulbure, Firas Abi-Farraj, and Marco Hutter. Fast Perception for Human-Robot Handovers with Legged Manipulators. In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, pages 734–742, Boulder CO USA, March 2024. ACM. ISBN 9798400703225. doi: 10.1145/3610977.3634958.

[35] Lorenzo Vianello, Luigi Penco, Waldez Gomes, Yang You, Salvatore Maria Anzalone, Pauline Maurice, Vincent Thomas, and Serena Ivaldi. Human-Humanoid Interaction and Cooperation: A Review. *Current Robotics Reports*, 2(4):441–454, December 2021. ISSN 2662-4087. doi: 10.1007/s43154-021-00068-z.

[36] Chen Wang, Linxi Fan, Jiankai Sun, Ruohan Zhang, Li Fei-Fei, Danfei Xu, Yuke Zhu, and Anima Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play. *arXiv preprint arXiv:2302.12422*, 2023.

[37] Zhen Wu, Jiaman Li, and C. Karen Liu. Human-Object Interaction from Human-Level Instructions, June 2024.

[38] Liang Xu, Xintao Lv, Yichao Yan, Xin Jin, Shuwen Wu, Congsheng Xu, Yifan Liu, Yizhou Zhou, Fengyun Rao, Xingdong Sheng, Yunhui Liu, Wenjun Zeng, and Xiaokang Yang. Inter-X: Towards Versatile Human-Human Interaction Analysis, December 2023.

[39] Wei Yang, Chris Paxton, Arsalan Mousavian, Yu-Wei Chao, Maya Cakmak, and Dieter Fox. Reactive Human-to-Robot Handovers of Arbitrary Objects, June 2021.

[40] Chengwen Zhang, Yun Liu, Ruofan Xing, Bingda Tang, and Li Yi. Core4d: A 4d human-object-human interaction dataset for collaborative object rearrangement. *arXiv preprint arXiv:2406.19353*, 2024.

[41] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

[42] Yifeng Zhu, Arisrei Lim, Peter Stone, and Yuke Zhu. Vision-based manipulation from single human video with open-world object graphs. *arXiv preprint arXiv:2405.20321*, 2024.

APPENDIX

A. Real World Setup

Deployment hardware The humanoid robot on which we deploy RHINO is Unitree H1 [1]. Following Cheng et al. [10], we assembled two DYNAMIXEL XL330-M288-T motors [3] with 3D printed gimble parts and a ZED Mini stereo camera [4] for two-DoF (yaw and pitch) active sensing. Each arm of H1 has 5 DoFs and a 6-DoF end-effector from [2], and other DoFs on the robot are not used.

Motion capture system We use ArUco markers and two cameras to build a simple motion capture system. We put four ArUco markers on the four corners of the workspace table to locate the cameras. Each human in the workspace wears two 3D-printed wristbands with four ArUco markers on each of them, illustrated in Figure 7. The cameras are calibrated and located using the OpenCV library and capture the human wristband's position in real time. Each wristband has an additional IMU sensor to capture the orientation of the human's wrist. To reduce the noise in the collected data, we use a Kalman filter to smooth the data.

Motion Detection and Object Detection As is illustrated in Section IV-F, we use the Body Tracking feature in ZED API to detect the body motion of the human and a fine-tuned YoloV11 [18] model to detect objects on the table. For hand detection, we use HaMeR[27] to obtain the human hand motion and then retarget[29] it to the 6-DoF robot hand. The visualization results are shown in Figure 8.

B. Skill Descriptions

In this section, we describe the skills that we deploy on the humanoid robot. The details include the description, success condition, reverse skill (if exists), and the human intention related to the skill. Note that the intention is inferred mainly from human behavior, hand positions, and the relative location of objects. The latter two can be concluded trivially to the start condition and end transition, and are shown by skill in Table VI. We concern the intention mainly on the human body motion in the narration as follows:

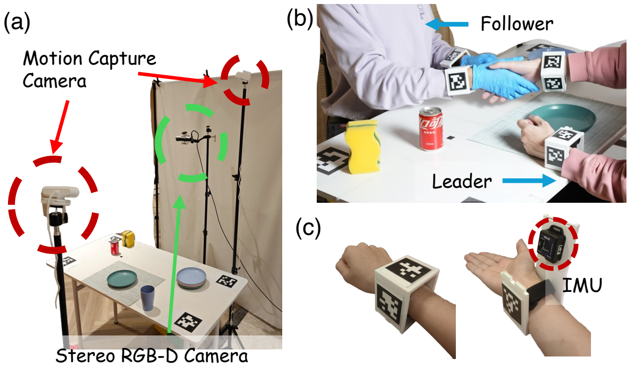


Fig. 7: **Setup of motion capture system.** a) The two Motion Capture Cameras are used to detect the ArUco markers. The video recorded by the RGB-D Camera is used to process human motion and human hand details of the Leader. b) Follower and Leader both wear wristbands with ArUco markers for hand position detection. c) We 3D-print our wristbands with 4 ArUco markers on 4 surfaces and embed a IMU beneath the upper surface.

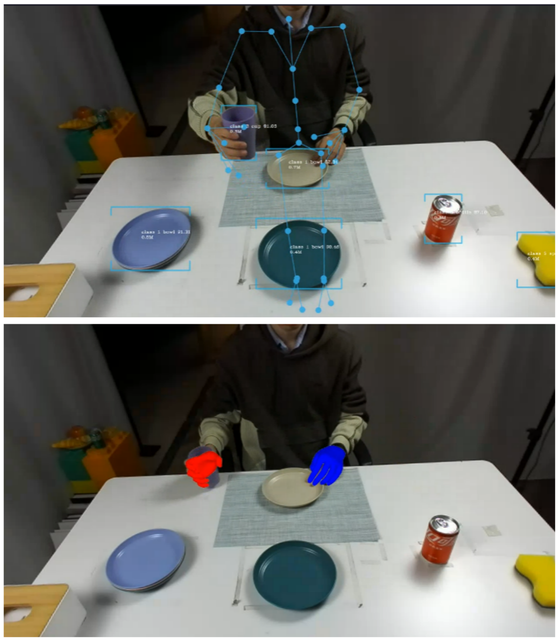


Fig. 8: **Visualization results of human body motion detection, object detection, and human hand detection.** The upper image demonstrates the human motion detection (only upper body is used) and object detection (in Dining scenario). The lower image demonstrate the human hand detection.

1) Scenario 1: Humanoid as a Dining Waiter

In this scenario, the leader human and the robot sit face-to-face at the side of a dining table. There are plates with food, a Coke can, a tissue box, and a sponge on the table. In the following skill descriptions, the humanoid robot takes the role of a helpful waiter and serves the leader human a meal with these objects. 10 skills related to 4 objects are listed below:

- *Pick Can*: The robot picks up a can with its right arm from the table. The success condition is that the can is lifted off the table. The interruption data includes the human taking the can away or the human putting his hand on the can. *Place Can* is the corresponding reverse skill. The leader human shows the intention by pointing to the can when the right robot arm is empty.
- *Place Can*: The robot places the can back on the table with its right arm. The success condition is that the can is placed on the table and the robot's hand is lifted off the can. The intention of this skill is shown by the leader human pointing to the place on the table where the can was placed before.
- *Get Plate from Human*: The robot fetches a plate from the hand of the human with its left arm. The success condition is the plate in the dexterous hand when the human loosens the grip of the plate. The leader human shows the intention by handing a plate forward.
- *Place Plate to Stack*: The robot stacks the plate in its right hand onto a pile of plates on the table. The success condition

is the plate settled on the top of the plate pile without slipping. The intention is given by the leader human pointing to the stack. Interruption data, in which the human touches the plate to stick the motion, is added to the collected dataset.

- *Pick Place from Table*: The robot lifts a plate on the table by both arms and holds it in the left hand. Application of this skill succeeds if there is no slippage in the motion until the plate is held. The leader human points to the plate to show the intention.
- *Handover Plate*: The robot protracts its left arm to give the leader human the plate on it. The showcase of this skill ends when the plate is put into the human hand. The leader human simply stretches the right hand out to show the intention. It is the reverse skill of *Get Plate from Human*.
- *Pick Sponge*: The robot picks up the sponge with its right arm. The sponge is placed beside the can. When it is lifted off the table the skill showcase succeeds. The leader human shows intention by mimicking washing. Data where the human snatches the sponge before the robot reaches it adds to the dataset. In case this happens during deployment, the robot withdraws its hand to the idle state.
- *Brush with Sponge*: It is a complex skill using both arms. The start condition is a plate in the left hand and a sponge in the right one when the leader human makes the washing gesture (same as that in *Pick Sponge*) again. To apply this skill, the robot moves the sponge close to the plate and rubs the sponge on the plate to brush it. The success condition is the robot keeps the periodic brushing motion for over 10 seconds.
- *Place Sponge*: The reverse skill of *Pick Sponge*. The robot puts the sponge in the right hand back onto the table to complete the skill demonstration. The intention is shown by the leader human pointing to the place on the table where the sponge was placed before (similar to skill *Place Can*).
- *Pick a Piece of Tissue*: The leader human points to the tissue box to express the intention. Then the robot uses its left hand to pull a tissue from the tissue box placed on the table corner and gives it to the leader human. The skill showcase succeeds when the leader human receives the tissue.

2) Scenario 2: Humanoid as an Office Assistant

In this scenario, the leader human sits across the humanoid robot at an office table. This time the robot transforms into an office assistant and deals with complicated cases such as stamping paper for approval, settling a baseball cap on the rack, picking and handing a book over, and reacting properly if the human takes a snap in working. There are 7 skills related to 4 objects in this scenario.

- *Settle Cap*: The robot gets a cap from the leader human's hand and settles it on a hat rack with its right arm. The skill showcase begins with the human holding the cap with both hands and ends with the robot pulling its hand back from the hat rack.
- *Handover Cap*: The robot takes the cap off the hat rack and sends it to the leader human. It is the reverse skill of *Settle Cap*. The related intention is inferred when seeing the human pointing to the rack. The success condition is that the human has received the cap.
- *Pick Book*: The robot picks a book from the shelf and hands it over. The skill begins with the human gesturing toward the book. When the human takes the book, this skill is completed successfully.
- *Pick Stamp*: The robot picks up the stamp on the table with its right hand. The skill succeeds when the stamp is lifted near the hand in an idle posture. The leader human instructs the execution of this skill by passing along the paper.
- *Stamp the Paper*: It is a delicate operation to make an issue for approval. The robot presses the stamp down onto the paper to mark a sign. This skill is considered successful only if one mark is imprinted. It is noteworthy that printing more than one mark in a single execution means that the model fails to predict the ending, thus being treated as a failure case. The sign of the related intention is the leader human pointing at the paper. To make an in-skill interruption, the human covers the paper with a hand to make the robot withdraw its hand if the pressing is not done.
- *Place Stamp*: The robot places the stamp back with its right hand. It is the reverse skill of *Pick Stamp* and is triggered by withdrawing the paper.
- *Turn off/on the Lamp*: Turning on and off the lamp share the same motion, and thus are trained as one skill. When the human slumps over the office desk to take a nap, the robot taps the switch of the lamp to turn off it. And when the human wakes up and lifts the head, the robot operates the same motion to turn on the lamp.

3) Interactive Motion Skills

Some skills are not involved with object operation and, thus, are not trained as manipulation skills. They are noted as motion skills. These skills are considered successful when the robot performs the motion properly as the human shows the intention and recovers to the idle posture when the intention no longer sustains.

- *Cheers*: The robot reaches out the right hand to touch the bottle held by the right hand of the human. Though holding a Coke can in the right hand during deployment, the robot does not manipulate the object. For this reason, this skill is not trained in a manipulation demonstration model.
- *Wave*: The robot lifts up the right hand and waves the right hand when the leader human is waving also.
- *Shake Hands*: The robot stretches its right hand out to touch the hand of the leader human with a handshaking posture.
- *Take Photo*: The robot lifts up the right hand and makes a V-sign when the human raises the phone to take a photo, and puts the hand done as the human puts away the phone.
- *Thumb Up*: The robot reaches both hands out with the thumbs up as the human gives it a thumb-up. Human intention with the left hand, right hand, or both is approved.
- *Spread out Hands*: The robot stretches its arms out to the sides with palms up when the leader human spreads its hands out.

TABLE VI: Description of the skills. Notes: The **Start Condition** or **End Transition** `[A, B]` means that object `A` is in the left hand of the robot and `B` is in the right hand. `empty` for this hand must be empty, `any` for this hand could hold any object or be empty, and − for this object remains unchanged after the skill is completed.

Scenarios	Object	Skill Name	Start Condition	End Transition	Num. of Data	Arm
Scenario 1 Dining Waiter	can	Pick Can	[any, empty]	[-, can]	107	Right
		Place Can	[any, can]	[-, empty]	100	Right
	plate	Get Plate from Human	[empty, any]	[plate, -]	100	Left
		Place Plate to Stack	[plate, any]	[empty, -]	98	Left
		Pick Plate from Table	[empty, empty]	[empty, plate]	115	Dual-Arm
		Handover Plate	[plate, any]	[empty, -]	115	Left
	sponge	Pick Sponge	[any, empty]	[-, sponge]	89	Right
		Brush with Sponge	[plate, sponge]	[-, -]	81	Dual-Arm
		Place Sponge	[any, sponge]	[-, empty]	82	Right
	tissue	Pick a Piece of Tissue	[empty, any]	[-, -]	105	Left
Scenario 2 Office Assistant	cap	Settle Cap	[any, empty]	[-, -]	111	Right
		Handover Cap	[any, empty]	[-, -]	110	Right
	book	Pick Book	[empty, any]	[-, -]	115	Left
	stamp	Pick Stamp	[any, empty]	[-, stamp]	92	Right
		Stamp the Paper	[any, stamp]	[-, -]	87	Right
		Place Stamp	[any, stamp]	[-, empty]	89	Right
	lamp	Turn off/on the Lamp	[empty, any]	[-, -]	85	Left
Expressive Motions	None	Cheers	[any, can]	[-, -]	66	Dual-Arm
		Wave	[any, empty]	[-, -]	39	Dual-Arm
		Shake Hands	[any, empty]	[-, -]	51	Dual-Arm
		Take Photo	[any, empty]	[-, -]	31	Dual-Arm
		Thumb Up	[empty, empty]	[-, -]	22	Dual-Arm
		Spread out Hands	[empty, empty]	[-, -]	26	Dual-Arm

C. Prompt for VLMs

Here are the prompts we give to Qwen and GPT-4o-mini in evaluation of the intention prediction module.

Prompt for the **Dining** scenario

You are a humanoid robot sitting in front of a human and equipped with a camera slightly tilted downward on your head, providing a first-person perspective. I am assigning you a new task to recognize to human gestures in front of you. Remember, the person is sitting facing you, so be mindful of their gestures. If the person is holding a cup to you and trying to cheer with you, answer ‘Cheers’. If the person is giving you a thumbs-up, answer ‘Thumbup’. If the person extends their right hand to shake hands with you, answer ‘ShakingHand’.If the person is waving to you with the right hand, answer ‘Waving‘. If the person is taking a photo of you with a cellphone, answer ‘Taking Photo‘. If the person is spreading out both hands in a gesture of resignation, answer ‘Spreading Hands‘. If the person is pointing to a Coke can in the middle of the table (on your right side), answer ‘Pointing Can’. If the person is pointing to an empty spot on the table with no objects (on your right side), answer ‘Pointing Table‘. If the person is pointing to a tissue box at the far left of the table, answer ‘Pointing TissueBox’. If the person is pointing to a plate in the middle of the table (just in front of you), answer ‘Pointing Plate’. If the person is holding out the right hand with the palm open toward you, answer ‘Palmup’. If the person is handing you a plate, answer ‘Handing Plate’. If the person is clenching their right fist, holding their left hand open and upward, and placing their right hand above the left as if pretending to wash a plate, answer ‘Washing’. If the person is pointing at a stack of plates on the left side of the table, answer ‘Pointing Plates’. If the person is pointing at a sponge on the right side of the table, answer ‘Pointing Sponge’. If the person is crossing his arms to form an X shape, answer ‘Cancel’. If no significant gestures are made, answer ‘Idle’. Respond directly with the corresponding options [Cheers, Thumbup, ShakingHand, Pointing Can, Pointing TissueBox, Pointing Plate, Palmup, Handing Plate, Washing, Pointing Plates, Pointing Sponge, Cancel, Idle] based on the current image and observed gestures. Directly reply with the chosen answer only, without any additional characters.

Prompt for the **Office** scenario

You are a humanoid robot sitting in front of a human and equipped with a camera slightly tilted downward on your head, providing a first-person perspective. I am assigning you a new task to recognize human gestures in front of you. Remember, the person is sitting facing you, so be mindful of their gestures. If the person is giving you a thumbs-up, answer ‘Thumbup‘. If the person extends their right hand to shake hands with you, answer ‘ShakingHand‘. If the person is waving to you with the right hand, answer ‘Waving‘. If the person is taking a photo of you with a cellphone, answer ‘Taking Photo‘. If the person is spreading out both hands in a gesture of resignation, answer ‘Spreading Hands‘. If the person is handing you a cap, answer ‘Handing Cap‘. If the person is pointing at a cap place on the right of the table, answer ‘Pointing Cap‘. If the person is handing a document to you with both hands and you are NOT holding

> *a stamp, answer ‘Handing File‘. If a document is placed in the center of the table in front of you, and the person is pointing to it with the right hand, answer ‘Pointing File‘. If the person retrieves the document from your side of the table to the other side, directly across from you, and you are still holding the stamp, answer ‘Retrieve File‘. If the person is lying down on the table and the lamp is ON, answer ‘Lie Down‘. If the person is sitting up from the table and the lamp is OFF, answer ‘Sit up‘. If the person is pointing at the books standing in the top left corner of the table, answer ‘Pointing Book‘. If the person is crossing the arms to form an X shape, answer ‘Cancel‘. If no significant gestures are made, answer ‘Idle‘. You are NOT holding a stamp right now and the lamp is now ON, observe the image and gestures carefully. Respond directly with the corresponding options [Pointing Book, Handing Cap, Pointing Cap, Handing File, Pointing File, Retrieve File, Lie Down, Sit up, ShakingHand, Thumbup, Cancel, Idle]. Directly reply with the chosen answer ONLY, without any additional characters.*

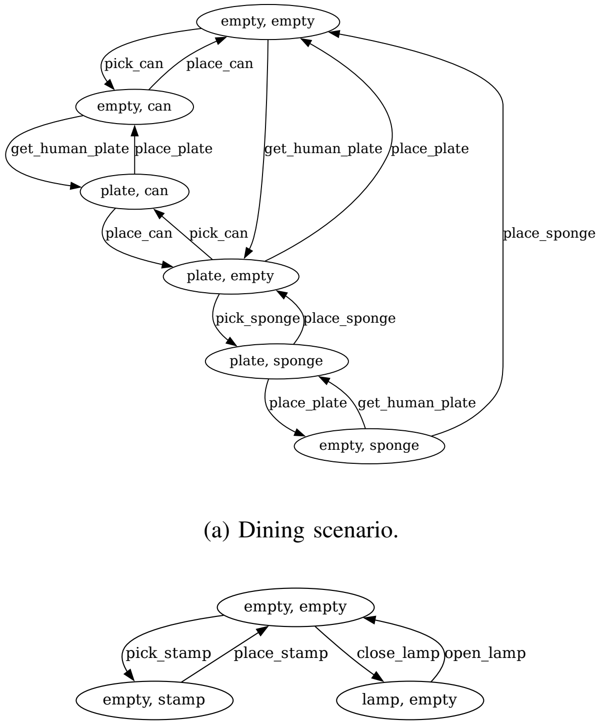
The sentence ‘*You are NOT holding a stamp right now and the lamp is now ON*’ is modified at each query according to the current situation (whether the robot is holding a stamp and whether the lamp is on).

D. Implementation of RHINO Modules

1) Reactive Planner

Algorithm 1 Pseudo-code for Skill Transitions of Reactive Planner.

```
$Skill$ ← Idle
while $true$ do
    $human\_intention$ ← Recognize_Human_Intention()
    if human intention is stable for $k$ frames and human intention != current intention then
        if human intention = Idle and Skill = Manipulation then
            Continue
        if Skill = Manipulation and interruptionAllowed then
            $Skill$ ← Reverse_Skill($Skill$)
        if Start_Condition($human\_intention$) is not satisfied by hand occupancy then
            $path$ ← FindPath(occupancy, StartCondition ($human\_intention$))
            $Skill$ ← Execute_Path($path$)
        else
            $Skill$ ← Corresponding Skill($human\_intention$)
    else if SkillSucceeded(Skill) or SkillTimeout(Skill) then
        $Skill$ ← Idle
        if SkillSucceeded(Skill) then
            Hand Occupancy ← End_Transition($Skill$)
```



(a) Dining scenario.

(b) Office scenario.

Fig. 9: Occupancy graph of two different scenarios.

The switching logic of the skill planner is listed in Algorithm 1. The directed graphs of occupancy are shown in Figure 9. Here we further explain the `Recognize_Human_Intention()` function in detail, which is implemented as a transformer-based classifier. The model input includes:

- **Upper Body Human Posture**: a 36-dim human upper body skeleton, namely the 6D rotation of the wrist, elbow and shoulder joints for each arm.
- **Human Hand Pose**: a 12-dim human hand pose vector. For each hand, we retarget the detected human hand pose to our robot hand with IK, and take the 6 joint pos as human hand pose vector.
- **Robot Hand Occupancy**: a 10-dim robot hand occupancy label. Since we have at most 5 objects in total (Can, Cup, Plate, Sponge, Tissue), we use a 5-dim one-hot label for each hand to represent the object held in the robot’s hand. If the robot is not holding anything, the label will be all-zeros.
- **Human Details**: a 19-dim vector, including the x and y-axis of each human hand position, the z-axis (height) of the human head position, and a 7-dim label for the nearest object to each hand. The nearest object label is concatenated by a 5-dim one-hot label of the object type, the distance from the object to the human hand, and the average of IOU and IOFs of the object bounding box and the human hand bounding box.

We use an MLP encoder to encode the concatenated vector of **Upper Body Human Posture**, **Human Hand Pose** and **Robot Hand Occupancy**, and another MLP to encode

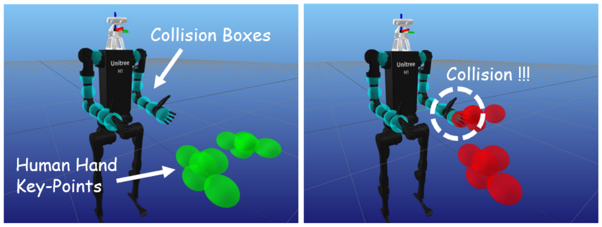


Fig. 10: The interface displays the work of the safety supervisor, with sphere markers representing the collision boxes of the human hands and robot arms. These markers move in sync with the interaction. When an unsafe collision is detected, the human hand markers change color from green to red.

Human Details to latent dimension. The concatenated latent vector is processed by a Transformer backbone, followed by a final MLP layer to predict the human intention class. The hyper-parameters of the Transformer backbone are listed in Table VII.

2) Interactive Motion Generation

For interactive motion generation, we use a transformer-based diffusion model, which denoises the past 30 frames of human and robot motions and future 5 frames of robot motions. Both human motion and robot motion consist of upper-body motion (36-dim for humans and 10-dim for humanoid), hand motion (6-dim for each hand,) and hand occupancy label (5-dim one-hot label for each hand). Besides, the predicted human intention label is also conditioned during the diffusion process. The hyper-parameters of our model are listed in Table VIII.

3) Manipulation Skills

Thanks to the stability of model training, most of the hyper-parameters are basically consistent across all skills. The volume of data for training each skill is shown as a column in Table VI. The hyper-parameters in training ACT models [41] are shown as Table IX.

For the prediction of the success signal, we marked the last n_s frames of the recorded data as 1 (completed) and other frames as 0 to generate a 0/1 label. n_s is set to 25 in most of the skills and shifted to 10 in three of them of which the ending frames changed sharply in motion. The special skills are *Pick Stamp*, *Stamp the Paper*, and *Place Stamp*.

4) Safety Supervisor

The collision box is calculated using 14 key points across each arm. The key points at specific joints and their midpoints are identified as follows:

- The origins of the shoulder pitch, shoulder yaw, elbow, and wrist joints are defined as key points.
- Additional key points include the midpoints between the shoulder yaw and elbow joints, and between the elbow and wrist joints.
- A further key point is defined at one-third the distance beyond the elbow towards the wrist, extending from the segment between these two joints.

This structured delineation allows for precise calculations pertinent to robotic arm movements within a predefined spatial configuration.

The human hands are shaped by the detected key points from body detection model of ZED API, from which each hand is reconstructed as 5 points. Once one of the points is close to any robot key point in 0.1 meters, an unsafe signal is broadcast to pause the robot control.

We also provide the visualization of the safety supervisor, of which the interface shown in Figure 10. When the human hand key-points collide with any collision box, the supervisor will send an unsafe signal to halt the robot. Our safety supervisor runs at 30Hz.

E. Detailed Experiment Results

1) Planner

We use confusion matrices to show the classification performance of our planner on the test dataset. The confusion matrices for our model, our model without human details and GPT-4o-mini on the test datasets of the dining and office scenarios are shown in Figure 11.

As is shown in the confusion matrices, although the model mainly relies on human body motion and human hand motion input for classification, human details can help the model better deal with certain situations, such as avoiding misclassification into Idle.

2) Objects Manipulation

The detailed success rates and average execution times across skills are presented in Table X, from which the statistics in Table III are derived.

In most skills, the manipulation module of RHINO autonomously executes motions following the patterns of teleoperation data within a comparable time frame. Trained exclusively on successful human teleoperation cases, the module demonstrates both effectiveness and robustness to slight scene variations during deployment. As a result, it achieves higher success rates in skills involving simple motions with abundant training data, such as *Pick Can*, *Handover Plate*, and *Place Stamp*.

However, certain skills pose challenges for the manipulation module. In *Place Plate to Stack* and *Stamp the Paper*, the robot hesitates to drop the plate or press the stamp due to prediction noise. In *Pick Plate from Table*, it must overcome increased friction against the table when joint positions deviate from those in the collected data. Another challenge arises in *Brush with Sponge*, where the success signal predictor struggles to assess the progress of the periodic motion accurately. As a result, termination is constrained by a 10-second timeout. These various factors contribute to a longer average execution time for these four skills compared to human performance.

Referring to the experiment on in-skill interruption data presented in Table IV, we select *Pick Can*, *Stamp the Paper*, and *Place Plate to Stack* as representative skills for interruptions occurring during the stages of fetching an object, operating with the object, and returning the object, respectively.

To ensure a controlled data volume across different interruption ratios, we assign a fixed data amount of M to

TABLE VII: Hyper-parameters of the Reactive Planner.

hyper-parameter	value
latent dimension	128
num head	8
num layers	3
batch size	256
feed-forward dimension	128
maximum epoch	300
learning rate	0.0001

TABLE VIII: Hyper-parameters of the Motion Generation Model.

hyper-parameter	value
latent dimension	256
num head	8
num layers	4
feed-forward dimension	256
diffusion steps	300
sampling steps	30
batch size	512
maximum epoch	4000
learning rate	0.0001

TABLE IX: Hyper-parameters of the ACT model for manipulation skills.

hyper-parameter	value
KL weight	10
Cross-entropy weight	1
chunk size	30
hidden dimension	512
batch size	45
feed-forward dimension	3200
maximum epoch	50000
learning rate	0.00005

TABLE X: Performance of manipulation module across manipulation skills.

Scenarios	**Object**	**Skill Name**	**Success Rate**	**Average Time**	**Success Rate (Human)**	**Average Time (Human)**
Scenario 1 Dining Waiter	can	Pick Can	1.00	5.31	1.00	5.77
		Place Can	1.00	4.10	0.93	4.65
	plate	Get Plate from Human	1.00	4.86	0.98	5.12
		Place Plate to Stack	0.95	8.19	0.97	6.91
		Pick Plate from Table	0.90	10.75	0.96	8.60
		Handover Plate	1.00	5.79	1.00	5.14
	sponge	Pick Sponge	0.95	8.19	1.00	7.45
		Brush with Sponge	0.90	10.02	1.00	4.18
		Place Sponge	0.85	5.57	0.98	5.41
	tissue	Pick a Piece of Tissue	0.95	9.43	0.91	9.54
Scenario 2 Office Assistant	cap	Settle Cap	1.00	7.50	0.91	8.50
		Handover Cap	0.85	8.64	0.90	10.48
	book	Pick Book	0.95	10.81	0.93	10.21
	stamp	Pick Stamp	1.00	4.80	0.92	3.91
		Stamp the Paper	0.80	5.64	0.92	3.11
		Place Stamp	1.00	4.74	0.93	4.83
	lamp	Turn off/on the Lamp	1.00	5.06	0.96	3.53

each skill. In a full data collection for any given skill, the total data amount is N, with N_{in} representing the portion containing in-skill interruptions. The ratio of interrupted data in a selected subset is denoted as α, meaning that $\lceil \alpha M \rceil$ slices contain interruptions. To maintain this proportion, we set $M = N - N_{in} + 1$. Specifically, M is set to 69, 66, and 76 for the three skills, respectively.

It is worth noting that the assigned data amount is smaller than that used in full data collection models (see Table VI), which results in a decrease in skill success rate and interrupt success rate compared to the final model.

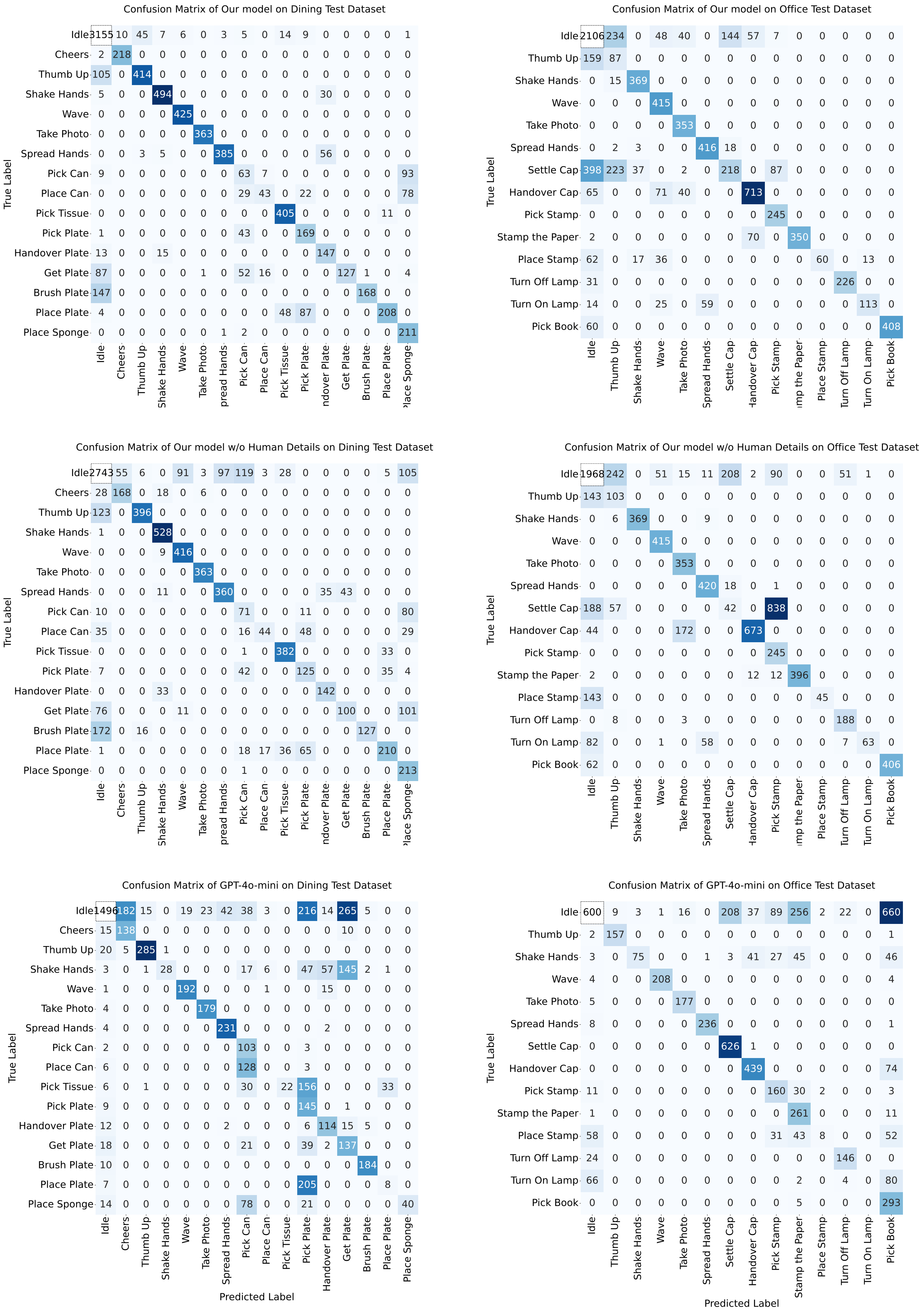
Each ACT model in this experiment uses the same hyper-parameters as those employed for the corresponding skill in both training and deployment with the full data collection.

3) End2End Policy

The detailed result to derive Table V is shown in Table XI. For each single skill, we collected 100 slices of motion in the dataset, which is close to the average volume of all the manipulation skills. As the small model is not capable of the skills which are totally unseen, we only examine each E2E model with the success rate of skills in the training data.

TABLE XI: Detailed success rate of RHINO framework and end-to-end model on different skills.

Method		**cheers**	**pick**	**place**	**handshake**	**wave**
Ours		1.00	1.00	0.85	1.00	1.00
E2E/1	I.D.	1.00	-	-	-	-
	O.O.D.	0.95	-	-	-	-
E2E/3	I.D.	1.00	0.75	0.35	-	-
	O.O.D.	0.95	0.30	0.45	-	-
E2E/5	I.D.	0.95	0.70	0.80	0.85	0.90
	O.O.D.	1.00	0.55	0.35	0.60	0.85



Confusion Matrix of Our model on Dining Test Dataset

	Idle	Cheers	Thumb Up	Shake Hands	Wave	Take Photo	pread Hands	Pick Can	Place Can	Pick Tissue	Pick Plate	ndover Plate	Get Plate	Brush Plate	Place Plate	lace Sponge
Idle	3155	10	45	7	6	0	3	5	0	14	9	0	0	0	0	1
Cheers	2	218	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Thumb Up	105	0	414	0	0	0	0	0	0	0	0	0	0	0	0	0
Shake Hands	5	0	0	494	0	0	0	0	0	0	0	30	0	0	0	0
Wave	0	0	0	0	425	0	0	0	0	0	0	0	0	0	0	0
Take Photo	0	0	0	0	0	363	0	0	0	0	0	0	0	0	0	0
Spread Hands	0	0	3	5	0	0	385	0	0	0	0	56	0	0	0	0
Pick Can	9	0	0	0	0	0	0	63	7	0	0	0	0	0	0	93
Place Can	0	0	0	0	0	0	0	29	43	0	22	0	0	0	0	78
Pick Tissue	0	0	0	0	0	0	0	0	0	405	0	0	0	0	11	0
Pick Plate	1	0	0	0	0	0	0	43	0	0	169	0	0	0	0	0
Handover Plate	13	0	0	15	0	0	0	0	0	0	0	147	0	0	0	0
Get Plate	87	0	0	0	0	1	0	52	16	0	0	0	127	1	0	4
Brush Plate	147	0	0	0	0	0	0	0	0	0	0	0	0	168	0	0
Place Plate	4	0	0	0	0	0	0	0	0	48	87	0	0	0	208	0
Place Sponge	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	211

Confusion Matrix of Our model on Office Test Dataset

	Idle	Thumb Up	Shake Hands	Wave	Take Photo	Spread Hands	Settle Cap	Handover Cap	Pick Stamp	mp the Paper	Place Stamp	Turn Off Lamp	Turn On Lamp	Pick Book
Idle	2106	234	0	48	40	0	144	57	7	0	0	0	0	0
Thumb Up	159	87	0	0	0	0	0	0	0	0	0	0	0	0
Shake Hands	0	15	369	0	0	0	0	0	0	0	0	0	0	0
Wave	0	0	0	415	0	0	0	0	0	0	0	0	0	0
Take Photo	0	0	0	0	353	0	0	0	0	0	0	0	0	0
Spread Hands	0	2	3	0	0	416	18	0	0	0	0	0	0	0
Settle Cap	398	223	37	0	2	0	218	0	87	0	0	0	0	0
Handover Cap	65	0	0	71	40	0	0	713	0	0	0	0	0	0
Pick Stamp	0	0	0	0	0	0	0	0	245	0	0	0	0	0
Stamp the Paper	2	0	0	0	0	0	0	70	0	350	0	0	0	0
Place Stamp	62	0	17	36	0	0	0	0	0	0	60	0	13	0
Turn Off Lamp	31	0	0	0	0	0	0	0	0	0	0	226	0	0
Turn On Lamp	14	0	0	25	0	59	0	0	0	0	0	0	113	0
Pick Book	60	0	0	0	0	0	0	0	0	0	0	0	0	408

Confusion Matrix of Our model w/o Human Details on Dining Test Dataset

	Idle	Cheers	Thumb Up	Shake Hands	Wave	Take Photo	pread Hands	Pick Can	Place Can	Pick Tissue	Pick Plate	ndover Plate	Get Plate	Brush Plate	Place Plate	lace Sponge
Idle	2743	55	6	0	91	3	97	119	3	28	0	0	0	0	5	105
Cheers	28	168	0	18	0	6	0	0	0	0	0	0	0	0	0	0
Thumb Up	123	0	396	0	0	0	0	0	0	0	0	0	0	0	0	0
Shake Hands	1	0	0	528	0	0	0	0	0	0	0	0	0	0	0	0
Wave	0	0	0	9	416	0	0	0	0	0	0	0	0	0	0	0
Take Photo	0	0	0	0	0	363	0	0	0	0	0	0	0	0	0	0
Spread Hands	0	0	0	11	0	0	360	0	0	0	0	35	43	0	0	0
Pick Can	10	0	0	0	0	0	0	71	0	0	11	0	0	0	0	80
Place Can	35	0	0	0	0	0	0	16	44	0	48	0	0	0	0	29
Pick Tissue	0	0	0	0	0	0	0	1	0	382	0	0	0	0	33	0
Pick Plate	7	0	0	0	0	0	0	42	0	0	125	0	0	0	35	4
Handover Plate	0	0	0	33	0	0	0	0	0	0	0	142	0	0	0	0
Get Plate	76	0	0	0	11	0	0	0	0	0	0	0	100	0	0	101
Brush Plate	172	0	16	0	0	0	0	0	0	0	0	0	0	127	0	0
Place Plate	1	0	0	0	0	0	0	18	17	36	65	0	0	0	210	0
Place Sponge	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	213

Confusion Matrix of Our model w/o Human Details on Office Test Dataset

	Idle	Thumb Up	Shake Hands	Wave	Take Photo	Spread Hands	Settle Cap	Handover Cap	Pick Stamp	mp the Paper	Place Stamp	Turn Off Lamp	Turn On Lamp	Pick Book
Idle	1968	242	0	51	15	11	208	2	90	0	0	51	1	0
Thumb Up	143	103	0	0	0	0	0	0	0	0	0	0	0	0
Shake Hands	0	6	369	0	0	9	0	0	0	0	0	0	0	0
Wave	0	0	0	415	0	0	0	0	0	0	0	0	0	0
Take Photo	0	0	0	0	353	0	0	0	0	0	0	0	0	0
Spread Hands	0	0	0	0	0	420	18	0	1	0	0	0	0	0
Settle Cap	188	57	0	0	0	0	42	0	838	0	0	0	0	0
Handover Cap	44	0	0	0	172	0	0	673	0	0	0	0	0	0
Pick Stamp	0	0	0	0	0	0	0	0	245	0	0	0	0	0
Stamp the Paper	2	0	0	0	0	0	0	12	12	396	0	0	0	0
Place Stamp	143	0	0	0	0	0	0	0	0	0	45	0	0	0
Turn Off Lamp	0	8	0	0	3	0	0	0	0	0	0	188	0	0
Turn On Lamp	82	0	0	1	0	58	0	0	0	0	0	7	63	0
Pick Book	62	0	0	0	0	0	0	0	0	0	0	0	0	406

Confusion Matrix of GPT-4o-mini on Dining Test Dataset

	Idle	Cheers	Thumb Up	Shake Hands	Wave	Take Photo	Spread Hands	Pick Can	Place Can	Pick Tissue	Pick Plate	Handover Plate	Get Plate	Brush Plate	Place Plate	Place Sponge
Idle	1496	182	15	0	19	23	42	38	3	0	216	14	265	5	0	0
Cheers	15	138	0	0	0	0	0	0	0	0	0	0	10	0	0	0
Thumb Up	20	5	285	1	0	0	0	0	0	0	0	0	0	0	0	0
Shake Hands	3	0	1	28	0	0	0	17	6	0	47	57	145	2	1	0
Wave	1	0	0	0	192	0	0	0	1	0	0	15	0	0	0	0
Take Photo	4	0	0	0	0	179	0	0	0	0	0	0	0	0	0	0
Spread Hands	4	0	0	0	0	0	231	0	0	0	0	2	0	0	0	0
Pick Can	2	0	0	0	0	0	0	103	0	0	3	0	0	0	0	0
Place Can	6	0	0	0	0	0	0	128	0	0	3	0	0	0	0	0
Pick Tissue	6	0	1	0	0	0	0	30	0	22	156	0	0	0	33	0
Pick Plate	9	0	0	0	0	0	0	0	0	0	145	0	1	0	0	0
Handover Plate	12	0	0	0	0	0	2	0	0	0	6	114	15	5	0	0
Get Plate	18	0	0	0	0	0	0	21	0	0	39	2	137	0	0	0
Brush Plate	10	0	0	0	0	0	0	0	0	0	0	0	0	184	0	0
Place Plate	7	0	0	0	0	0	0	0	0	0	205	0	0	0	8	0
Place Sponge	14	0	0	0	0	0	0	78	0	0	21	0	0	0	0	40

Confusion Matrix of GPT-4o-mini on Office Test Dataset

	Idle	Thumb Up	Shake Hands	Wave	Take Photo	Spread Hands	Settle Cap	Handover Cap	Pick Stamp	Stamp the Paper	Place Stamp	Turn Off Lamp	Turn On Lamp	Pick Book
Idle	600	9	3	1	16	0	208	37	89	256	2	22	0	660
Thumb Up	2	157	0	0	0	0	0	0	0	0	0	0	0	1
Shake Hands	3	0	75	0	0	1	3	41	27	45	0	0	0	46
Wave	4	0	0	208	0	0	0	0	0	0	0	0	0	4
Take Photo	5	0	0	0	177	0	0	0	0	0	0	0	0	0
Spread Hands	8	0	0	0	0	236	0	0	0	0	0	0	0	1
Settle Cap	0	0	0	0	0	0	626	1	0	0	0	0	0	0
Handover Cap	0	0	0	0	0	0	0	439	0	0	0	0	0	74
Pick Stamp	11	0	0	0	0	0	0	0	160	30	2	0	0	3
Stamp the Paper	1	0	0	0	0	0	0	0	0	261	0	0	0	11
Place Stamp	58	0	0	0	0	0	0	0	31	43	8	0	0	52
Turn Off Lamp	24	0	0	0	0	0	0	0	0	0	0	146	0	0
Turn On Lamp	66	0	0	0	0	0	0	0	0	2	0	4	0	80
Pick Book	0	0	0	0	0	0	0	0	0	5	0	0	0	293

Fig. 11: **Confusion Matrices of Our Model (with and without Human Details) and GPT-4o-mini.** To show the results more clearly, we did not color the cell in the top left corner since ”idle” accounts for a significant proportion in the data.