ScrewMimic: Bimanual Imitation from Human Videos with Screw Space Projection

Arpit Bahety, Priyanka Mandikal, Ben Abbatematteo, Roberto Martín-Martín
The University of Texas at Austin

***Abstract*—Bimanual manipulation is a longstanding challenge in robotics due to the large number of degrees of freedom and the strict spatial and temporal synchronization required to generate meaningful behavior. Humans learn bimanual manipulation skills by watching other humans and by refining their abilities through play. In this work, we aim to enable robots to learn bimanual manipulation behaviors from human video demonstrations and fine-tune them through interaction. Inspired by seminal work in psychology and biomechanics, we propose modeling the interaction between two hands as a serial kinematic linkage — as a screw motion, in particular, that we use to define a new action space for bimanual manipulation: screw actions. We introduce ScrewMimic, a framework that leverages this novel action representation to facilitate learning from human demonstration and self-supervised policy fine-tuning. Our experiments demonstrate that ScrewMimic is able to learn several complex bimanual behaviors from a single human video demonstration, and that it outperforms baselines that interpret demonstrations and fine-tune directly in the original space of motion of both arms. For more information and video results, https://robin-lab.cs.utexas.edu/ScrewMimic/**

I. Introduction

Manipulation in human environments often requires coordinating the motion of two arms, e.g., opening a bottle, cutting a block in two pieces, or stirring a pot. In *dexterous bimanual manipulation*, the agent has to generate behavior for both arms that are synchronized spatially and temporally, rendering it even more complex to generate than two independent unimanual manipulations. Due to its complexity, in nature, this kind of behavior is almost unique to higher-level primates [1, 2, 3, 4], and it requires several years to fully develop in humans [5, 6], being mastered only after a significant amount of time of observing expert bimanual agents and practicing through trial-and-error. This work aims to endow robots with novel capabilities to learn bimanual manipulation tasks.

Learning to generate dexterous bimanual manipulation in robots is challenging due to the large state and action spaces resulting from the two arms, and the strict requirements of spatial and temporal synchronization between them to achieve success [7, 8, 9]. As a result, exploring randomly in this space is prohibitively difficult, especially on real robot hardware, limiting some of the successes to simulation [10, 11, 12, 13]. A promising approach to reduce the challenge of searching for a successful bimanual manipulation policy is to observe a human performing a bimanual manipulation and imitate it. However, due to the morphology differences between the human and the robot, the direct execution of the observed bimanual interaction may not be successful, necessitating an exploratory refinement to adapt to the robot's embodiment and capabilities, which reintroduces the challenges of exploring directly in the space of motion of both arms.

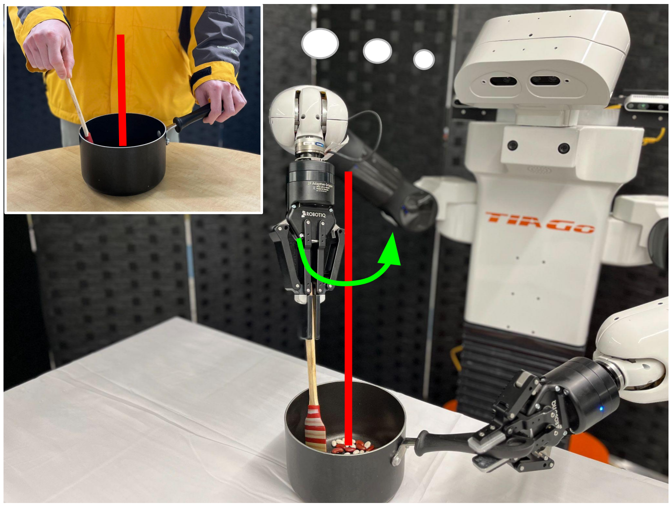


Fig. 1. **Bimanual manipulation** tasks can be represented by a screw axis (red line) constraining and synchronizing the motion of both hands. ScrewMimic maps a single human demonstration into a screw axis, improves it with an iterative interactive exploration procedure, and learns to predict it for new object instances and poses, enabling their manipulation.

The main insight in this work is that for many bimanual manipulation tasks, the relative motion between hands can be explained by a simple one-degree-of-freedom (1-DoF) screw joint. This virtual joint constrains the motion in a way that matches an existing physical constraint in the environment (e.g., when opening a laptop or a bottle with both hands) or just facilitates the manipulation (e.g., when cutting a block or stirring a pot, see Fig. 1). The type of 1-DoF screw joint — prismatic, revolute, screw— captures different modalities of bimanual manipulation, while the screw joint parameters fully specify the motion. This insight works at several levels: in perception, it serves as a prior for interpreting noisy sensor signals and facilitates understanding a human-demonstrated bimanual manipulation. And, in exploration, it provides an action space where both arm motions are coordinated by design, allowing efficient action fine-tuning to find successful behaviors with the real robot's embodiment.

We present a novel method, ScrewMimic, that leverages this insight for one-shot visual imitation learning of bimanual manipulation from a human demonstration. Our method uses

[*] Correspondences to abahety@utexas.edu

a single demonstration as input as an RGB-D video of a human performing a bimanual manipulation task. SCREWMIMIC interprets the demonstration as a screw motion between both hands and uses the perceived bimanual grasp and virtual joint to train a prediction model on 3D point clouds. This model predicts full bimanual manipulation behaviors composed of bimanual grasping strategies and two-arm relative motion in a possibly moving reference frame, for novel views of the object. These predictions form the starting hypothesis for a self-practicing iterative process. Here, the robot engages in bimanual interactions, learning to overcome morphological differences by optimizing a reward signal generated autonomously, resulting in successful bimanual manipulation strategies. The new strategy can then be used in a self-improving loop to retrain a better prediction model that is also able to generalize to new instances of the same object class thanks to a set of geometric augmentations.

We demonstrate the performance of our solution in six challenging bimanual manipulation tasks involving different types of screw motion between both hands, both in objects with physical kinematic constraints and in tasks where the constraints need to be virtually created by the agent. Our experiments indicate that the projection into the screw-axis space is a robust representation for bimanual manipulation—leading to sample efficient exploration and strong performance in executing bimanual tasks.

II. RELATED WORK

SCREWMIMIC is a novel solution to generate and refine autonomous robot bimanual manipulation behavior bootstrapped with a single video of a human demonstration. In the following, we contrast SCREWMIMIC to the most relevant prior work in robot bimanual manipulation and visual imitation learning.

a) Bimanual Manipulation: Early on, robotics researchers acknowledged the need for bimanual manipulators to solve tasks in unstructured environments [14, 15, 16]. Generating coordinated behavior for both arms became a significant challenge [8, 17] that researchers have attempted to solve with planning [7, 18, 19], control [20], reinforcement learning [21], and imitation learning [22, 23]. To generate bimanual manipulation behavior, these solutions have to explore a large action space with strict temporal and spatial synchronization. A common strategy is to coordinate behavior using stable static postures or keypoints [24, 25, 26], or with explicit spatial or temporal constraints typically extracted via kinesthetic teaching [27, 28, 29, 30]. Oftentimes, these approaches necessitate specialized teleoperation hardware like custom devices [31, 32, 33] or motion capture [34, 35] that limit their scalability and availability. In contrast, SCREWMIMIC uses a single RGB-D video of a human demonstration, which is cheaper to acquire, scalable and does not require controlling a robot.

Given the large action space and difficulty of exploration, reinforcement learning approaches to bimanual manipulation are prohibitively costly to train on real robot hardware. As an alternative, researchers have explored sim-to-real approaches [12, 36, 37]. These approaches suffer from the reality gap which is exacerbated in contact-rich manipulation tasks with complicated dynamics [38]. An alternative approach is to employ movement primitives [39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51] which reduce the search space but limit expressiveness and typically require substantial engineering effort. Recently, Grannen et al. [52] proposed a stabilizing-acting bimanual manipulation framework where the stabilizing hand is trained using human annotations and the acting hand is trained using kinesthetic demonstrations. In contrast, SCREWMIMIC leverages a novel action representation that efficiently learns bimanual manipulation policies given only a *single human video* demonstration, and can correct failed actions through a self-supervised policy fine-tuning method.

b) Visual Imitation Learning: Recent work has sought to imbue robots with the ability to learn from large collections of unstructured human videos like Ego4D [53] or YouTube videos [54, 55]. Some works have proposed learning cost functions from video and language data [56, 57, 58, 59], whereas others propose pretraining objectives [60, 61]. More direct approaches generally track human hands in videos (e.g. with FrankMocap [62]), mapping the hand trajectories to the robot's action space [63, 64, 65, 66, 67]. A common approach in these works is to structure video understanding by modelling manipulation using affordances (i.e. detecting contact points [68]) and subsequent interaction trajectories . Since the robot's embodiment differs from a human demonstrator and tracking is generally noisy, interactive fine-tuning is generally necessary to obtain a reliable behavior policy [69, 70, 71]. DEFT [66], for example, trains an affordance prediction model on large-scale data and obtains the interaction trajectory given a human demonstration at test time. This trajectory is then refined through interaction. Inspired by this line of work, we propose a novel formulation of synchronized bimanual manipulation learned solely by watching human-object interactions in video. In contrast to prior work in unimanual manipulation, our focus here is on the action representation. Our unique formulation of bimanual motion in terms of screw joints abstracts complex high-DoF manipulation into a unified framework—enabling efficient imitation learning from video.

III. PRELIMINARIES: SCREW THEORY

SCREWMIMIC models bimanual manipulation as a screw motion between the two hands. Chasles' theorem states that any rigid body motion can be written as the composition of a rotation of the body about a unique line in space and a translation along the same line. This line is referred to as the *screw axis* of that motion. A screw axis $\mathcal{S}$ can be represented as $(q, \hat{s}, h)$ where $q \in \mathbb{R}^3$ is any point on the axis, $\hat{s} \in \mathbb{R}^3$ is a unit vector in the direction of the axis, and $h \in \mathbb{R}_+$ is the pitch of the screw, defining the ratio of linear motion along the screw axis to the rotational motion around the screw axis [73, 74].

Assuming some angular displacement $\theta \in \mathbb{R}$ along a screw axis $\mathcal{S}$, the corresponding rigid body motion in exponential

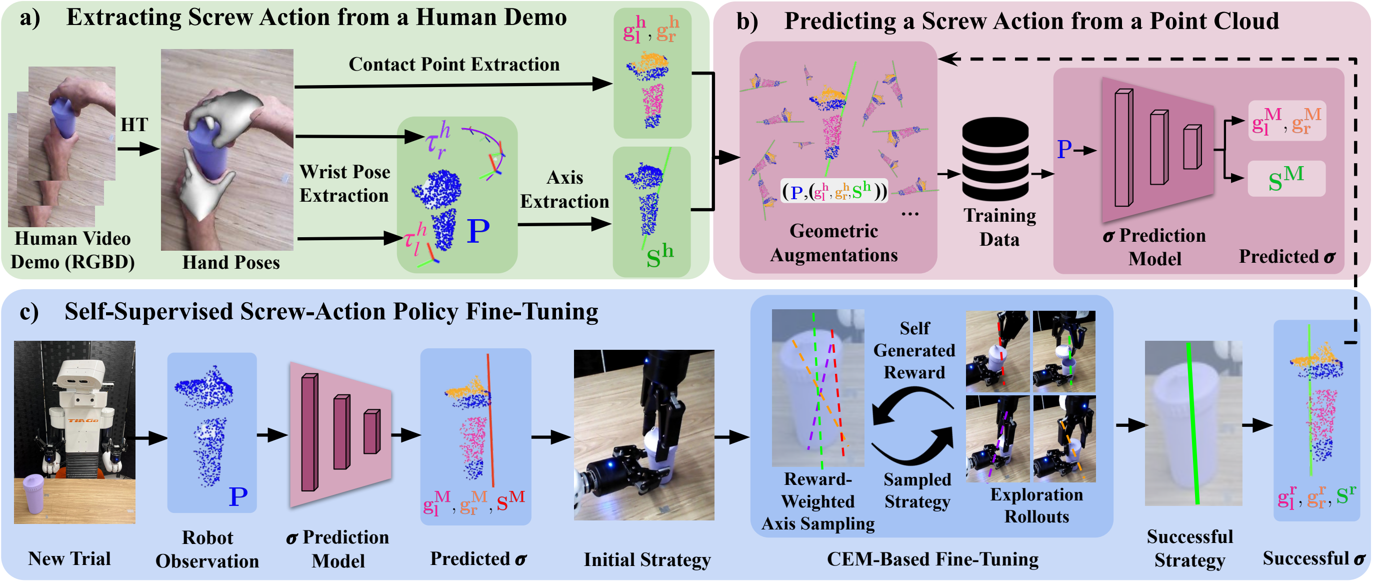


Fig. 2. **Overview of ScrewMimic. a)** Given an RGB-D video of a human performing a bimanual task, we use off-the-shelf hand tracking (HT) models [62, 68] to extract a trajectory of wrist poses τ^h and grasp contact points (g_l^h, g_r^h). ScrewMimic interprets τ^h as a screw motion between both hands to estimate screw axis parameters S^h (Sec. IV-A). **b)** Next, we apply geometric augmentations on the 3D object point cloud to train a PointNet [72] model to estimate screw actions for novel object views (Sec. IV-B). **c)** Finally, the trained model generates an initial hypothesis that the robot executes and iteratively refines using an autonomously generated reward signal. The successful data point is further used to improve the prediction model (Sec. IV-C).

coordinates, $\xi = (\omega, v) \in \mathbb{R}^6$ is given by

$$\xi = \mathcal{S}\theta = \begin{bmatrix} \omega \\ v \end{bmatrix} = \begin{bmatrix} \hat{s}\theta \\ -\hat{s}\theta \times q + h\hat{s}\theta \end{bmatrix} \tag{1}$$

where $\omega \in \mathbb{R}^3$ represents angular motion, and $v \in \mathbb{R}^3$ represents linear motion. To transform this into a homogeneous transformation matrix, $T \in SE(3)$ ($SE(3)$ the Special Euclidean Lie Group) we apply the matrix exponential: $T = \exp([\mathcal{S}]\theta)$, where $[\mathcal{S}]\theta \in se(3)$ ($se(3)$ is the Lie algebra) is the matrix representation of the exponential coordinates:

$$[\mathcal{S}]\theta = \begin{bmatrix} [\omega] & v \\ 0 & 0 \end{bmatrix}. \tag{2}$$

with $[\omega] \in so(3)$ is a skew-symmetric matrix representing orientation.

Conversely, given a rigid body transformation, $T \in SE(3)$, we can compute the corresponding screw axis as follows. In the case of a pure translation ($h = \infty$), the screw axis can be recovered as $\hat{s}$ pointing in the direction of linear motion, and q is any point. In the case of pure rotation ($h = 0$), we can recover the corresponding twist in matrix form $[S]$ using the matrix logarithm: $[\mathcal{S}]\theta = \log(T)$. Applying Eq. 1, we can obtain the screw axis parameters as

$$\hat{s} = \frac{\omega}{||\omega||}, \quad q = \frac{\hat{s} \times v}{||\omega||}. \tag{3}$$

For the general case with $h \notin \{0, \infty\}$ and further details, we refer the reader to Lynch and Park [74].

We will use the definitions above to infer a screw axis from a sequence of relative transformations between human hands, and to generate relative motion between robot hands for a given screw axis. In this work, we will consider three screw types: pure translation ($h = \infty$, `prismatic`), pure rotation ($h = 0$, `revolute`), and rotation with a fixed orientation (`revolute3D`). We describe the axis computation and trajectory generation for each case in Sec. IV below.

IV. ScrewMimic: Policy Learning with Screw Actions

Seminal research in psychology and biomechanics [75] indicates that bimanual behavior in humans can be modeled as if "a serial kinematic chain would connect both hands", where one hand (left) sets a spatial reference frame and the other (right) moves relative to it. Inspired by this work, we propose a novel action space parametrization for robotic bimanual manipulation that we call **screw actions**, that fully specifies the behavior of both hands through a screw joint between the hands. A screw action is defined as, $\sigma = (g_l, g_r, S, \tau_l)$ in its most general form. g_l and g_r are the grasping/placing locations for left and right hands. S is a 1-DoF screw axis describing the relative motion between left and right hands. Finally, τ_l is a possible sequence of left-hand pose changes during the interaction (e.g. moving a pot to the stove while stirring) that can be empty if the left hand just fixates/stabilizes the object.

Given a screw action, the motion of both hands of the robot during the bimanual manipulation is fully specified. Our main hypothesis is that our new action space simplifies robot dexterous bimanual manipulation at two levels: first, it aids in learning from visual human demonstrations by projecting noisy multi-hand motion into a simpler constrained space, and second, it facilitates fine-tuning the perceived motions by providing a constrained space in which real-world exploration is more efficient. We propose a novel solution, ScrewMimic, that leverages this insight to learn bimanual policies. ScrewMimic integrates three modules: a perceptual module to interpret a single human demonstration as a screw action, a prediction model that predicts screw actions based on a point cloud of an object, and a self-supervised iterative fine-tuning algorithm that explores in screw action space to find optimal parameters for bimanual tasks. In the following, we explain each of these modules in detail.

A. Extracting a Screw Action from a Human Demonstration

The first module of ScrewMimic (Fig. 2a) parses an RGB-D video of a human demonstrating a bimanual task into a suitable action representation for robot execution, in our case, a screw action $\sigma^h = (g_l^h, g_r^h, S^h, \tau_l^h)$ (h indicates *human*). ScrewMimic first extracts the grasping/placing location of the human hands (g_l^h and g_r^h) using an off-the-shelf hand-object detector [68], detecting the first intersection of the hand and the object bounding boxes in the RGB image sequence, and projecting it into the 3D point cloud of the object, P, using the information of the depth channel.

ScrewMimic then extracts the 6-DoF trajectories (position and orientation) of the human hands using an off-the-shelf hand-tracking solution (FrankMocap [62]) to detect the wrist poses over time, τ_l^h and τ_r^h. A direct approach would use these trajectories to imitate and fine-tune the bimanual manipulation. However, the original trajectories contain noise from the visual tracker, the motion of both hands is not constrained to be synchronized, and an embodiment gap exists between the human hand and the robot gripper, making it harder to imitate and fine-tune (as shown in Sec. V); ScrewMimic overcomes these limitations by interpreting the trajectories as a screw action.

Inspired by models of human bimanual manipulation [75], we assign an acting and reference role to the right and left hands, respectively, keeping τ_l^h as the trajectory of the left hand. ScrewMimic finds then the screw axis S^h by transforming τ_r^h to the left hand reference frame and analyzing the left-right relative motion to obtain the screw-joint type, m, and parameters, $\hat{s}$ and q. For that, ScrewMimic assumes three possible screw joint types, prismatic, revolute and revolute with fixed orientation, as explained in Sec. III. Assuming $m = \texttt{prismatic}$, ScrewMimic obtains the screw parameters by fitting a 3D line to the trajectory of the right wrist. When $m = \texttt{revolute}$, the screw axis parameters can be obtained by transforming each pose of the right wrist relative to the left wrist to exponential coordinates using the matrix logarithm, applying Eq. 3, and averaging the resulting $\hat{s}$ and q. Finally, if $m = \texttt{revolute3D}$, ScrewMimic first fits a plane to the trajectory of the right wrist; the normal to the plane provides, $\hat{s}$. The right wrist trajectory is then projected onto the plane and ScrewMimic fits a circle to it; the center of the circle provides q. We employ a Maximum a Posteriori Estimation (MAP) method to determine the screw joint type (m) corresponding to a human demonstration. In this case, we want to estimate m based on the hand pose observations, τ_r^h. To do this, ScrewMimic evaluates the likelihood of each joint type (m), by comparing the observed demonstration trajectory τ_r^h with the trajectory computed based on m. This comparison involves evaluating a score function that measures the distance between the two trajectories, considering both positional and angular differences at each waypoint. A lower distance indicates greater similarity between trajectories. ScrewMimic picks the joint type with the highest likelihood. Examples of the extracted screw axes for each type are depicted in Fig. 3.

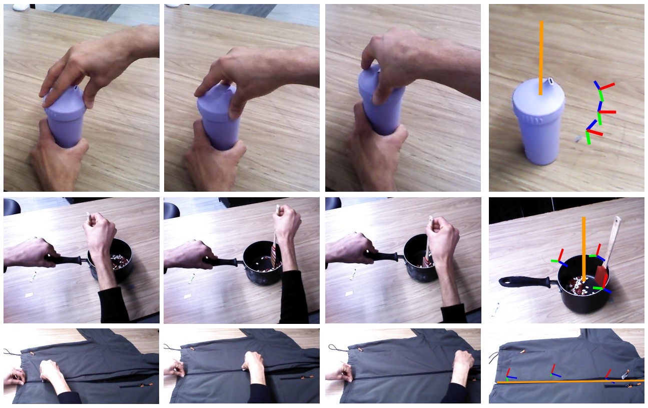


Fig. 3. **Human demonstrations as screw actions.** Three frames of a human demonstration for three bimanual tasks (top row: opening a bottle, $m = \texttt{revolute}$, middle row: stirring a pot, $m = \texttt{revolute3D}$, bottom row: opening a zipper, $m = \texttt{prismatic}$) and the perceived screw axis explaining the motion (fourth column, orange indicates the axis line). Our screw action representation facilitates the interpretation of noisy hand trajectory observations in a bimanual interaction as evidence of a simple 1-DoF constraint between both hands

B. Predicting a Screw Action from a Point Cloud

Once the robot perceives the human demonstration of the bimanual task and represents it in the screw action space, how can it generalize to novel object instances and configurations? To tackle this, the second module of ScrewMimic (Fig. 2b) includes a PointNet [72] based model trained to predict the screw action from an object's point cloud. Concretely, given an RGB-D observation, we use MDETR [76] to segment out the object and extract its partial point cloud, P. The goal is to learn a perception model $M : P \mapsto \sigma^M = (g_l^M, g_r^M, S^M)$ (M indicates *Model*). Here g_l^M, g_r^M refer to the grasp contact points predicted by the perception model for the left and the right grippers respectively and S^M refers to the predicted screw axis. From here on, we omit the left-hand trajectory, τ_l, since our experiments focus on learning the relative motion between hands, but the left-hand motion is enabled by our general formalism, as shown in additional trials in Appendix A and website.

ScrewMimic benefits from the 3D nature of both the input observation and screw action representation that allows for straightforward geometric augmentations of the data: translation, rotation, and scaling. These augmentations are applied to the point cloud, P, and corresponding robot action σ^M. As a result, we generate an extensive training dataset from just a *single* human demonstration. Using PointNet [72] as the backbone, we construct two networks: a regression network trained with MSE loss to predict the axis and a segmentation network trained with negative log likelihood loss for identifying contact points. We train task-specific prediction models. The training of each model is efficient, requiring on average 40 minutes for 2000 epochs on a RTX 4090 GPU. This rapid training cycle enables quick model refinement and incorporation of new data, as we will discuss in the next section.

C. Self-Supervised Screw-Action Policy Fine-Tuning

Given that human pose tracking is inherently noisy, the prediction model trained on the human hand trajectory will necessarily have some error. Thus, if the robot directly executes the bimanual manipulation defined by the predicted screw action, it will probably fail (as evidenced in our experiments). Nevertheless, the predicted screw action produces a behavior close to a successful manipulation and thus can be used as initialization for a fine-tuning procedure through interaction. The third module of ScrewMimic consists of a self-supervised policy improvement algorithm that refines the noisy screw action (Fig. 2c). As our experiments indicate (Sec. V), the use of screw actions as policy parameterization is critical for more efficient bimanual exploration and allows ScrewMimic to achieve success in multiple tasks. In the following, we first explain how a screw action is used by ScrewMimic to generate a bimanual manipulation behavior, and then, we describe the iterative process to fine-tune an initial (failing) screw action into a successful one.

Given a predicted screw action $\sigma^M = (g_l^M, g_r^M, S^M)$, the two grippers first go to g_l^M and g_r^M at pre-defined orientations using a whole-body controller [77]. The end of this initial motion is the beginning of the bimanual manipulation described in Sec. III by the screw axis S. While the left hand is possibly executing a trajectory τ_l, the right hand will move relative to it following the constraints indicated by S. ScrewMimic creates $k \in 1 \dots K$ waypoints along the screw axis with steps of θ_T/K, where θ_T is a predetermined total amount of translation along the axis-line for $m =$ `prismatic` type, or the total amount of rotation around the axis-line for $m =$ `revolute` and $m =$ `revolute3D` types. In the latter case, the orientation of the right hand is kept constant during the motion. Assuming an initial 6D pose for the right hand of T_0^{right} with respect to the left hand, the right-hand poses will be given by $T_i = \exp([S]\theta_k)\ T_0^{right}$, where $\exp$ is the matrix exponential and $\theta_k = k\theta_T/K$.

Given the method explained above to generate bimanual manipulation behavior based on a screw action, we now explain the iterative procedure to fine-tune an initially failing action. Inspired by prior exploratory approaches [66, 70, 78], ScrewMimic implements a sampling-based optimization framework based on the cross-entropy method (CEM). The process starts with obtaining the initial screw action from the prediction model that was trained on the human demonstration. Next, an initial sampling distribution, D, is used to sample screw axis parameters around the initial screw axis. E samples, $\xi_{1,e}$, are drawn from this distribution. CEM then requires a reward to score each sample and guide a reweighting of the sampling distribution (D) for the next epoch. In ScrewMimic, the CEM optimization process is self-supervised through an autonomously generated reward based on the length of the episode and the amount of force employed, measured by a force-torque (FT) sensor in the right hand's wrist. Concretely, after each epoch, all the trajectories up to that epoch are ranked by their length: the longer an episode runs without failure, the better it is. We implement three self-detected failures: 1) when the robot is not applying enough force (norm of the wrench signal is below a threshold), indicating that it may be moving in free space instead of manipulating, 2) when the robot is applying too much force (norm of the wrench signal is above a threshold), indicating that it is trying to manipulate an object in the wrong way, and 3) when the robot loses grasp (measured by the finger proprioception). After all the episodes are ranked by their lengths, ScrewMimic takes the top T trajectories and ranks them by the mean wrenches employed over the episode; using lower force for the manipulation is considered more efficient. These episodes form the elite set. ScrewMimic updates the sampling distribution based on the elite set and uses the new distribution in another epoch. The process repeats for N epochs or until the bimanual manipulation succeeds. The CEM fine-tuning procedure is summarized in Algorithm 1.

The successfully executed screw action $\sigma^r = (g_l^r, g_r^r, S^r)$ is added to the training dataset (see Fig. 2) to enhance the action prediction model. This iterative process, if performed repeatedly, can facilitate continuous improvement of both the robot's policy and the prediction model, creating a self-supervised feedback loop where each component bolsters the other as demonstrated in Sec. V.

Algorithm 1 Cross-Entropy Method Optimization

```
Require: parameter distribution $D$, total epochs $N$, episodes
  in each epoch $E$, elite trajectories threshold $T$, $S_{init} =
  (\hat{s}_{init}, q_{init})$ initial screw axis

  $\xi_{init} \leftarrow (\hat{s}_{init}, q_{init})$
  $D \leftarrow N(0, \sigma^2)$
  for $n = 1...N$ do
      for $e = 1...E$ do
          Sample $\epsilon_{n,e} \sim D$
          Execute $\xi_{n,e} = \xi_{init} + \epsilon_{n,e}$
          Collect reward $R_{n,e}$; reset environment
      end for
      $\xi_1, \xi_2 ... \xi_T \leftarrow$ Order trajectories $\xi_{0,0}, \xi_{0,1}$ ... $\xi_{n,E}$ based
  on rewards
      $\Omega \leftarrow \{\epsilon_{\xi_1}, \epsilon_{\xi_2}$ ... $\epsilon_{\xi_T}\}$
      Fit $D$ to $\Omega$
  end for
  $\xi_{final} \leftarrow \xi_{init} + \epsilon_{final}$
```

V. Experimental Evaluation

We evaluate ScrewMimic on six real-world bimanual tasks: open bottle, close zipper, insert roll, close laptop, stir and cut. These tasks collectively encompass three types of screw joint models: prismatic, revolute, and revolute3D. They also involve screw actions in two types of objects: articulated objects with actual physical joints constraining their motion (as in bottles, rolls, and laptops) and objects without constraints, where the screw action creates a virtual joint that facilitates the correct bimanual manipulation (as in stirring, cutting, and zipping a jacket). While the manipulation of articulated objects has been studied more extensively in the past [79, 80, 81, 82, 83], this is the first time, to the best of our knowledge, that a framework unifies the bimanual manipulation of rigid and articulated objects through virtual joints. In the following, we explain each task in brief:

- open bottle: A bottle with its cap closed is placed upright on the table. The robot performs the opening action as defined by the screw action, followed by a lift arm command. We consider success if the cap is separated from the base of the bottle at the end.
- close zipper: A jacket is kept in a configuration as shown in the first row of Fig. 4. We consider success if the robot zips 90% of the jacket at the end.
- insert roll. A roll is placed beside the box aligned as shown in the fourth row of Fig. 4. We consider success if the robot inserts 90% of the roll inside the box at the end.
- close laptop: A laptop is placed on the table, opened to around 100°. We consider success if the robot closes the laptop (final opening $< 10°$).
- stir: A container with a ladle propped against its side is placed in front of the robot. The container has two different colored beans, initially separated. We consider success if the two types of beans are significantly mixed after the stirring as measured by a human evaluator.
- cut: The robot is holding a scraper knife in one gripper and tasked with cutting a block of clay ($\sim$7 cm in height). We consider success if the block of clay is cut into two pieces at the end.

In all our experiments, we use a PAL-Robotics Tiago++ bimanual manipulator and control its two arms using a whole-body controller that maps desired end-effector poses for both arms to joint torques using an inverse-kinematics-based solution with task-priority control to avoid self-collisions [77]. For perception, we use an Orbbec Astra S RGB-D camera mounted on Tiago++'s head both to observe humans and to predict screw actions on objects, and an ATI mini45 force-torque sensor mounted on the right hand's wrist.

For each task, ScrewMimic begins with the screw action predicted by the trained model after observing a single human interaction using a perceived point cloud as input, and fine-tunes it using its self-supervised iterative procedure. Each trial of the procedure is limited to a maximum of 5 epochs, each containing 5 episodes, after which, if the procedure did not find a successful screw action, we consider the trial a failure. The fine-tuning takes around 40 minutes, demonstrating a reasonable real-world exploration time. Success is verified manually after each episode, and a human resets the environment if necessary.

Experiments and Results:

In our experiments, we aim to answer four questions:

Q1) Is a single human demonstration enough for* ScrewMimic *to achieve success in bimanual manipulation tasks? To evaluate this question, we perform three trials per human demonstration for each of the tasks and observe if, in the trials, ScrewMimic successfully achieves the bimanual tasks with its self-supervised fine-tuning in screw action space. We also annotate the amount of interaction (episodes) necessary on average to succeed in the task. We use a single demonstration per task, but each trial starts with a different (novel) location of the object(s). Therefore, ScrewMimic needs to predict the screw action in a new location and start the iterative process there. A trial for each task is depicted in each row of Fig. 4, columns 1 to 6. The results are summarized in Table I.

TABLE I
Generalization to new object poses

	# Successes	Avg Epochs and Episodes
Open Bottle	2/3	(3, 18)
Close Zipper	3/3	(2, 11)
Insert Roll	3/3	(1, 8)
Close Laptop	3/3	(2, 12)
Stir	2/3	(3, 16)
Cut	3/3	(1, 7)

Overall, ScrewMimic achieves an aggregated success of 90% in all trials. ScrewMimic failed only in one trial of the open bottle and the stir tasks as the fine-tuning process finished without any successful screw action. Due to the small number of episodes (25 maximum), we observe a marked dependency on the first screw action samples for the fine-tuning procedure, which could be alleviated with a larger number of episodes per epoch. Despite that, we consider that our experiments indicate that, in most cases, **ScrewMimic succeeds in all studied bimanual manipulation tasks using only a single video of a human demonstration**, thanks to the structure provided by the screw action for perception and self-supervised exploration. Additionally, we also conduct experiments to analyze the robustness of ScrewMimic to noisy demonstrations. We observe that despite noisy demonstrations, ScrewMimic is able to extract a screw axis sufficiently accurate for fine-tuning. The details and results of the experiment are shown in Appendix D.

Q2) Can a policy fine-tuning and model retraining loop enable* ScrewMimic *to continually improve and generalize to new objects? We assess if ScrewMimic can use the corrected screw action obtained after fine-tuning to improve the prediction model and generalize to unseen objects. In this experiment, the screw action prediction model is first trained with the noisy screw action parsed from the human

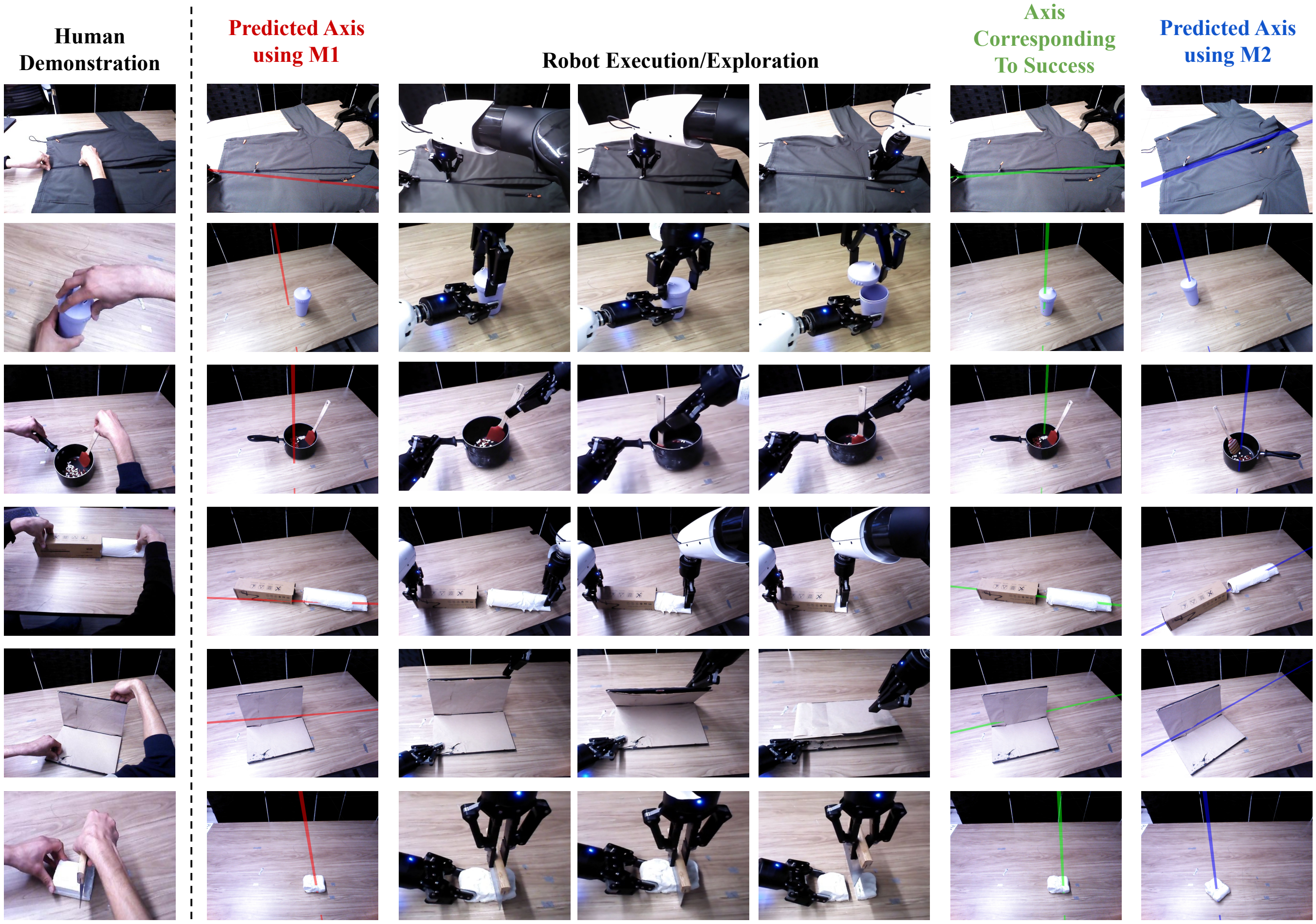


Fig. 4. **Screw action fine-tuning and prediction model re-training result.** The first column shows the human demonstration for each task. The second column shows the axis predicted by M1, the model trained on the axis extracted from the human demonstration, with the object at a novel pose. Columns 3-5 show snapshots from an episode in the fine-tuning stage. Column 6 shows the axis corresponding to the successful trajectory obtained during the aforementioned process. Column 7 shows the predicted axis for a novel object pose from the prediction model re-trained on the corrected axis. This result shows how the robot starts from a noisy screw axis and using the screw action fine-tuning, corrects the axis. Furthermore, it also shows that this corrected axis can be used to re-train the prediction model to output a more accurate axis.

demonstration (denoted M1 in Table II). The robot then executes and fine-tunes this screw action to obtain a corrected one and uses it to re-train the prediction model, obtaining the model M2. SCREWMIMIC then uses model M2 to predict and execute the bimanual tasks and performance is measured. Table II reports the exploration iterations required until success is achieved as (epochs, episodes), where each epoch consists of 5 episodes and the policy is updated after each epoch. Our experiments indicate that, after retraining, SCREWMIMIC succeeds at the task with the same object (second column) almost zero-shot, showing that the prediction model can be iteratively improved using the corrected action obtained from the fine-tuning stage.

We then assess how SCREWMIMIC handles new objects. We place a novel object of the same category at a new pose and run the same experiment (see Fig. 5). For certain tasks such as `open bottle` and `stir`, initially, using M2, the screw action prediction is sub-par (Table II, third column). This is due to the large geometric difference between the bottle that M2 was trained on and the new bottle. However, after fine-tuning and obtaining model M3, SCREWMIMIC can complete the task with the new object almost zero-shot (Table II, fourth column). For certain other tasks such as `insert roll` and `cut`, the initial screw action prediction is good as the structural difference is smaller between the old and the new object. This indicates that **SCREWMIMIC helps create a self-learning loop where the robot can continually expand its manipulation capabilities to new objects**. We also compare training from scratch with a pre-trained PointNet model and observe an improvement in the screw axis prediction on novel objects. The details and results are shown in Appendix C

Q3) What benefits does the screw axis representation have compared to a more direct, $N\times$6-DoF representation? To assess the importance of the screw axis representation we compare it with two baselines visualized in Fig. 6. First, the *FM + $N\times$6-DoF* baseline (Table III, first row) extracts the initial trajectory from hand tracker using the wrist poses as

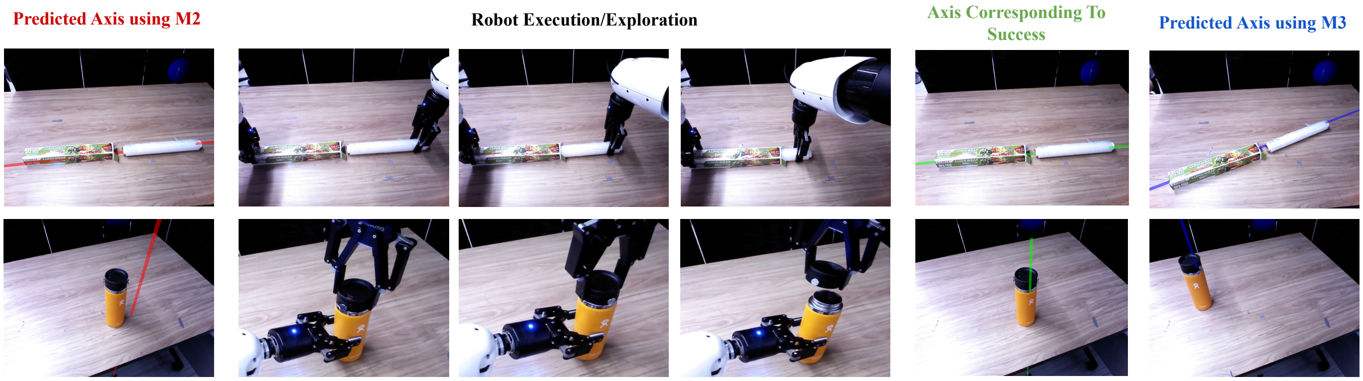


Fig. 5. **Generalization to new objects.** The first column shows the axis predicted by M2, the model trained on the corrected screw action for the first object. Columns 2-4 show snapshots from an episode in the fine-tuning stage. Column 5 shows the axis corresponding to the successful trajectory obtained during the aforementioned process. Column 6 shows the predicted axis from the prediction model re-trained on the corrected axis (M3). Thus, ScrewMimic can obtain reasonable screw action predictions and fine-tune them to generalize to new objects. We show results for other tasks in Appendix, Sec. E.

TABLE II
Generalization to new objects [1]

	Same object		New object	
	M1	M2	M2	M3
Open Bottle	(3, 16)	(0, 1)	(2, 14)	(0, 1)
Close Zipper	(1, 9)	(0, 1)	(0, 3)	(0, 1)
Insert Roll	(1, 7)	(0, 2)	(0, 3)	(0, 1)
Close Laptop	(2, 12)	(0, 1)	(0, 4)	(0, 2)
Stir	(3, 16)	(0, 1)	(1, 6)	(0, 2)
Cut	(1, 7)	(0, 1)	(0, 2)	(0, 1)

[1] Results reported as (Epochs, Episodes) until a success is reached.

TABLE III
Action Representation Comparison

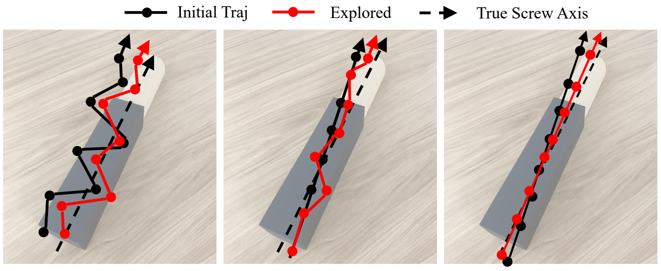
	Task	Success?	#Episodes	Dense Metric
FM + $N\times$6-DoF (DEFT*)	Bottle	No	25/25	0%
	Roll	No	25/25	0%
	Laptop	No	25/25	10%
Screw + $N\times$6-DoF	Bottle	No	25/25	10%
	Roll	No	25/25	50%
	Laptop	No	25/25	50%
Screw + Screw (ScrewMimic)	Bottle	Yes	16/25	100%
	Roll	Yes	7/25	100%
	Laptop	Yes	11/25	100%

N waypoints directly. We call this space as $N\times$6-DoF as it has N waypoints with each waypoint described by a 6-DoF pose. During fine-tuning, it explores in the $N\times$6-DoF space by adding noise to the initial waypoints (Fig. 6a). With this baseline, we ablate the screw representation both as the human demonstration parser and as the action space during fine-tuning. Note that this baseline emulates DEFT's [66] fine-tuning stage. Due to the unavailability of DEFT's code/model, we attempt to approximate DEFT's methodology as closely as possible. Key differences include: 1) the use of both hands in our method, 2) the extraction (rather than prediction) of grasping locations directly from demonstrations, and 3) a reliance on ScrewMimic's self-generated CEM reward, rather than human-assigned scores. The second baseline, *Screw + $N\times$6-DoF* (Table III, second row) extracts the initial trajectory from the output of the hand tracker using ScrewMimic's parser module as a screw axis, but explores by adding Gaussian noise in SE(3) during fine-tuning (Fig. 6b). This baseline parses the human demonstration in the same way as ScrewMimic but explores differently. We compare against our proposed ScrewMimic (Table III, third row), indicated as *screw + screw* (Fig. 6c).

Each baseline obtains an initial trajectory from a human demonstration, then performs the fine-tuning procedure in the corresponding action space. For each method, we run one trial of the exploration process for each of the three tasks —`open bottle`, `insert roll` and `close laptop` as shown in Table III. We indicate whether the robot can achieve success in the allotted 5 epochs (5 episodes each), as well as the percentage of the task that the robot completes (Table III, last column), measured by a human.

Our results in Table III indicate that neither *FM + $N\times$6-DoF* nor *Screw + $N\times$6-DoF* representations enable task success, as exploring in the $N\times$6-DoF space is much more challenging. We hypothesize that, for *FM + $N\times$6-DoF*, the failing behavior is not only caused by the large uncorrelated exploration space but also by a more noisy initial trajectory that keeps the inherent noise present in the hand-tracking module. In contrast, the use of screw actions enables ScrewMimic to clean the perceived human demonstration and also makes the fine-tuning process more efficient by exploring in the reduced screw axis space.

Q4) Are both autonomous reward signals correctly guiding the policy fine-tuning stage? The screw action policy fine-tuning stage requires a way to rank episodes in our CEM procedure, guiding the robot to explore around good episodes while disregarding bad ones to converge to success. ScrewMimic uses two signals to rank any episode as described in Sec. IV-C: the length of an episode (based on a loss of grasp/fixation or exceeding a Force-Torque (FT) threshold), and the mean wrench measured over the episode. To assess the importance of these two signals for the fine-tuning stage, we ablate each component individually. Since removing the FT sensor threshold can be dangerous for the robot and the



(a) FM + N × 6-DoF Noise (b) Screw + N × 6-DoF Noise (c) Screw + Screw

Fig. 6. **Exploration intuition.** The different action representations from Table III are illustrated in 2D for the `insert tissue roll` task. The true (prismatic) screw axis is visualized as a dashed line. The resulting initial and sampled exploration trajectories are visualized as black and red, respectively. (a) The baseline *FM + $N \times$6-DoF*: obtaining an initial trajectory from FrankMocap as N 6-DoF waypoints, then performing exploration around that trajectory with noise in 6-DoF space. (b) The baseline *Screw + $N \times$6-DoF*: perceiving the initial trajectory as a screw axis but exploring with noise in 6-DoF space. (c) SCREWMIMIC (*Screw + Screw*), perceiving the demonstration as a screw action and exploring in the space of screw axes.

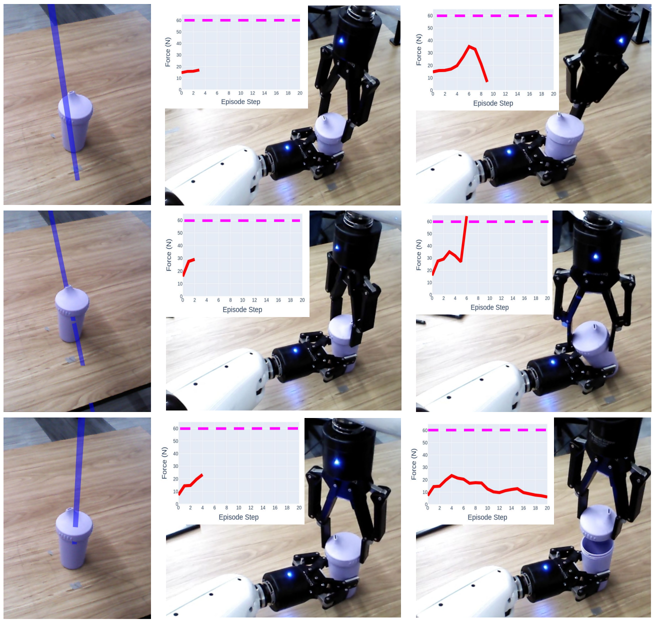


Fig. 7. **Reward intuition.** Each row shows the screw axis and the snapshots of the corresponding episode to showcase the use of the reward components – Grasp lost (first row), FT sensor reaching the threshold (second row). Each snapshot has the corresponding FT sensor reading until that timestep. The pink line shows the FT threshold. The last row shows an example of a success with the corresponding FT sensor readings.

objects, we always retain it. Fig. 7 provides examples of the roles of these reward terms. The results in Table IV indicate that for the `open bottle` and `stir` tasks, if either of the two components is absent, the policy fine-tuning process fails and the robot fails to complete the task within the allotted rollouts. For the `insert roll` task, there was never a grasp lost in any episode, so SCREWMIMIC succeeds even without the grasp loss detection. However, it fails without the mean episode sensed wrench. This shows that both reward signals are critical for SCREWMIMIC's policy fine-tuning stage.

TABLE IV
ABLATION OF REWARD COMPONENTS

	Task	Success?	#Episodes	Dense Metric
w/o Grasp Lost Detection	Bottle	No	25/25	20%
	Roll	Yes	7/25	100%
	Stir	No	25/25	None
w/o Mean Episode FT	Bottle	No	25/25	30%
	Roll	No	25/25	10%
	Stir	No	25/25	None
SCREWMIMIC	Bottle	Yes	16/25	100%
	Roll	Yes	7/25	100%
	Stir	Yes	18/25	None

VI. LESSONS AND CONCLUSION

In this work, we present and validate SCREWMIMIC, a robust representational framework for bimanual manipulation that significantly boosts performance by simplifying complex tasks into screw actions derived from a single human demonstration. While our results demonstrate the capabilities of SCREWMIMIC, it is not without limitations and scope for future work. First, the screw action formulation, although versatile, does not fit all bimanual manipulation tasks, e.g., tasks where the hands are not constrained to move along a single axis, such as cutting in a zig-zag motion. Future work can extend SCREWMIMIC to include sequences of screw axes, requiring more complex inference and exploration algorithms. Second, we train a separate prediction model for each object class, which limits the generalization capabilities of SCREWMIMIC. This can be addressed by training a single multi-task model on a diverse array of objects. Large-scale human-activity datasets [53, 84] offer an exciting avenue to scale up the range of tasks and objects that SCREWMIMIC can learn from. For that, our method should also relax the dependency on depth sensors, which could be obtained instead from RGB using recent algorithms [85, 86]. Third, episode success is recorded manually in our experiments. This could be automated in the future using vision-language foundation models [87, 88]. Fourth, while SCREWMIMIC focuses on improving the screw axis prediction, it could be beneficial to also fine-tune the grasp contact points. Finally, due to 3D sensor limitations, some reflective surfaces such as the laptop cannot be correctly perceived from all angles and we need to cover them. We do not deem this a problem of SCREWMIMIC but rather of the (relatively outdated) depth sensor—using more modern 3D sensors would alleviate it. Despite these limitations, SCREWMIMIC demonstrates that using a screw axis space representation for bimanual actions facilitates efficient exploration leading to strong improvements in task success. Additionally, the incorporation of a self-supervised fine-tuning process allows the robot to iteratively refine its own actions. Our work is a promising step towards enabling robots to efficiently learn complex bimanual manipulation tasks by watching humans.

References

[1] Sandra A Heldstab, Zaida K Kosonen, Sonja E Koski, Judith M Burkart, Carel P van Schaik, and Karin Isler. Manipulation complexity in primates coevolved with brain size and terrestriality. *Scientific reports*, 6(1):24528, 2016.

[2] O. V. Kazennikov, Brian I. Hyland, Michel R. Corboz, Alexandre Babalian, Eric M. Rouiller, and Mario Wiesendanger. Neural activity of supplementary and primary motor areas in monkeys and its relation to bimanual and unimanual movement sequences. *Neuroscience*, 89: 661–674, 1999. URL https://api.semanticscholar.org/CorpusID:2804384.

[3] Opher Donchin, A. D. Gribova, Orna Steinberg, Hagai Bergman, and Eilon Vaadia. Primary motor cortex is involved in bimanual coordination. *Nature*, 395:274–278, 1998. URL https://api.semanticscholar.org/CorpusID:4370872.

[4] J. A. Scott Kelso. Phase transitions and critical behavior in human bimanual coordination. *The American journal of physiology*, 246 6 Pt 2:R1000–4, 1984. URL https://api.semanticscholar.org/CorpusID:45949058.

[5] Karen E Adolph, Bennett I Bertenthal, Steven M Boker, Eugene C Goldfield, and Eleanor J Gibson. Learning in the development of infant locomotion. *Monographs of the society for research in child development*, pages i–162, 1997.

[6] Jérôme Barral, Bettina Debû, and Christina Rival. Developmental changes in unimanual and bimanual aiming movements. *Developmental neuropsychology*, 29(3): 415–429, 2006.

[7] Yoshihito Koga and J-C Latombe. Experiments in dual-arm manipulation planning. In *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pages 2238–2239. IEEE Computer Society, 1992.

[8] Christian Smith, Yiannis Karayiannidis, Lazaros Nalpantidis, Xavi Gratal, Peng Qi, Dimos V Dimarogonas, and Danica Kragic. Dual arm manipulation—a survey. *Robotics and Autonomous systems*, 60(10):1340–1353, 2012.

[9] Adrià Colomé and Carme Torras. Dimensionality reduction for dynamic movement primitives and application to bimanual manipulation of clothes. *IEEE Transactions on Robotics*, 34:602–615, 2018. URL https://api.semanticscholar.org/CorpusID:20726581.

[10] Oliver Kroemer, Christian Daniel, Gerhard Neumann, Herke van Hoof, and Jan Peters. Towards learning hierarchical skills for multi-phase manipulation tasks. *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1503–1510, 2015. URL https://api.semanticscholar.org/CorpusID:12178097.

[11] Kevin Sebastian Luck and Heni Ben Amor. Extracting bimanual synergies with reinforcement learning. *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4805–4812, 2017. URL https://api.semanticscholar.org/CorpusID:20155594.

[12] Yuanpei Chen, Tianhao Wu, Shengjie Wang, Xidong Feng, Jiechuan Jiang, Zongqing Lu, Stephen McAleer, Hao Dong, Song-Chun Zhu, and Yaodong Yang. Towards human-level bimanual dexterous manipulation with reinforcement learning. *Advances in Neural Information Processing Systems*, 35:5150–5163, 2022.

[13] Satoshi Kataoka, Seyed Kamyar Seyed Ghasemipour, Daniel Freeman, and Igor Mordatch. Bi-manual manipulation and attachment via sim-to-real reinforcement learning. *ArXiv*, abs/2203.08277, 2022. URL https://api.semanticscholar.org/CorpusID:247476081.

[14] Raymond C Goertz. Fundamentals of general-purpose remote manipulators. *Nucleonics*, pages 36–42, 1952.

[15] Robonaut: Nasa's space humanoid. *IEEE Intelligent Systems and Their Applications*, 15(4):57–63, 2000.

[16] Jonathan Bohren, Radu Bogdan Rusu, E. Gil Jones, Eitan Marder-Eppstein, Caroline Pantofaru, Melonee Wise, Lorenz Mösenlechner, Wim Meeussen, and Stefan Holzer. Towards autonomous robotic butlers: Lessons learned with the pr2. In *2011 IEEE International Conference on Robotics and Automation*, pages 5568–5575, 2011. doi: 10.1109/ICRA.2011.5980058.

[17] Aude Billard and Danica Kragic. Trends and challenges in robot manipulation. *Science*, 364(6446):eaat8414, 2019.

[18] Nikolaus Vahrenkamp, Dmitry Berenson, Tamim Asfour, James Kuffner, and Rüdiger Dillmann. Humanoid motion planning for dual-arm manipulation and re-grasping tasks. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2464–2470. IEEE, 2009.

[19] Benjamin Cohen, Sachin Chitta, and Maxim Likhachev. Single-and dual-arm motion planning with heuristic search. *The International Journal of Robotics Research*, 33(2):305–320, 2014.

[20] Ping Hsu. Coordinated control of multiple manipulator systems. *IEEE Transactions on Robotics and Automation*, 9(4):400–410, 1993.

[21] Adrià Colomé and Carme Torras. *Reinforcement learning of bimanual robot skills*. Springer, 2020.

[22] R Zollner, Tamim Asfour, and Rüdiger Dillmann. Programming by demonstration: dual-arm manipulation tasks for humanoid robots. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 1, pages 479–484. IEEE, 2004.

[23] Heecheol Kim, Yoshiyuki Ohmura, and Yasuo Kuniyoshi. Transformer-based deep imitation learning for dual-arm robot manipulation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8965–8972. IEEE, 2021.

[24] Elena Gribovskaya and Aude Billard. Combining dynamical systems control and programming by demonstration for teaching discrete bimanual coordination tasks to a humanoid robot. In *Proceedings of the 3rd ACM/IEEE*

international conference on Human robot interaction, pages 33–40, 2008.

[25] Tamim Asfour, Pedram Azad, Florian Gyarfas, and Rüdiger Dillmann. Imitation learning of dual-arm manipulation tasks in humanoid robots. *International journal of humanoid robotics*, 5(02):183–202, 2008.

[26] Joao Silvério, Leonel Rozo, Sylvain Calinon, and Darwin G Caldwell. Learning bimanual end-effector poses from demonstrations using task-parameterized dynamical systems. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 464–470. IEEE, 2015.

[27] Sylvain Calinon, Zhibin Li, Tohid Alizadeh, Nikos G Tsagarakis, and Darwin G Caldwell. Statistical dynamical systems for skills acquisition in humanoids. In *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pages 323–329. IEEE, 2012.

[28] Andrej Gams, Bojan Nemec, Auke Jan Ijspeert, and Aleš Ude. Coupling movement primitives: Interaction with the environment and bimanual tasks. *IEEE Transactions on Robotics*, 30(4):816–830, 2014.

[29] Lucia Pais Ureche and Aude Billard. Constraints extraction from asymmetrical bimanual tasks and their use in coordinated behavior. *Robotics and autonomous systems*, 103:222–235, 2018.

[30] Nadia Figueroa and Aude Billard. Learning complex manipulation tasks from heterogeneous and unstructured demonstrations. In *Proceedings of Workshop on Synergies between Learning and Interaction*, 2017.

[31] Ana-Lucia Pais Ureche and Aude Billard. Learning bimanual coordinated tasks from human demonstrations. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts*, pages 141–142, 2015.

[32] 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.

[33] Albert Tung, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Yuke Zhu, Li Fei-Fei, and Silvio Savarese. Learning multi-arm manipulation through collaborative teleoperation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9212–9219. IEEE, 2021.

[34] Franziska Krebs and Tamim Asfour. A bimanual manipulation taxonomy. *IEEE Robotics and Automation Letters*, 7(4):11031–11038, 2022.

[35] Christian RG Dreher and Tamim Asfour. Learning temporal task models from human bimanual demonstrations. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7664–7671. IEEE, 2022.

[36] Satoshi Kataoka, Seyed Kamyar Seyed Ghasemipour, Daniel Freeman, and Igor Mordatch. Bi-manual manipulation and attachment via sim-to-real reinforcement learning. *arXiv preprint arXiv:2203.08277*, 2022.

[37] Rohan Chitnis, Shubham Tulsiani, Saurabh Gupta, and Abhinav Gupta. Efficient bimanual manipulation using learned task schemas. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1149–1155. IEEE, 2020.

[38] Wenshuai Zhao, Jorge Peña Queralta, and Tomi Westerlund. Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In *2020 IEEE symposium series on computational intelligence (SSCI)*, pages 737–744. IEEE, 2020.

[39] Rudolf Lioutikov, Oliver Kroemer, Guilherme Maeda, and Jan Peters. Learning manipulation by sequencing motor primitives with a two-armed robot. In *Intelligent Autonomous Systems 13: Proceedings of the 13th International Conference IAS-13*, pages 1601–1611. Springer, 2016.

[40] Yahav Avigal, Lars Berscheid, Tamim Asfour, Torsten Kröger, and Ken Goldberg. Speedfolding: Learning efficient bimanual folding of garments. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–8, 2022. doi: 10.1109/IROS47612.2022.9981402.

[41] Fan Xie, Alexander Chowdhury, M De Paolis Kaluza, Linfeng Zhao, Lawson Wong, and Rose Yu. Deep imitation learning for bimanual robotic manipulation. *Advances in neural information processing systems*, 33: 2327–2337, 2020.

[42] Thomas Weng, Sujay Man Bajracharya, Yufei Wang, Khush Agrawal, and David Held. Fabricflownet: Bimanual cloth manipulation with a flow-based policy. In *Conference on Robot Learning*, pages 192–202. PMLR, 2022.

[43] Huy Ha and Shuran Song. Flingbot: The unreasonable effectiveness of dynamic manipulation for cloth unfolding. In *Conference on Robot Learning*, pages 24–33. PMLR, 2022.

[44] Rohan Chitnis, Shubham Tulsiani, Saurabh Gupta, and Abhinav Gupta. Intrinsic motivation for encouraging synergistic behavior. *arXiv preprint arXiv:2002.05189*, 2020.

[45] Jennifer Grannen, Priya Sundaresan, Brijen Thananjeyan, Jeffrey Ichnowski, Ashwin Balakrishna, Minho Hwang, Vainavi Viswanath, Michael Laskey, Joseph E Gonzalez, and Ken Goldberg. Untangling dense knots by learning task-relevant keypoints. *arXiv preprint arXiv:2011.04999*, 2020.

[46] Jennifer Grannen, Yilin Wu, Suneel Belkhale, and Dorsa Sadigh. Learning bimanual scooping policies for food acquisition. In *6th Annual Conference on Robot Learning*, 2022.

[47] Aleksandar Batinica, Bojan Nemec, Aleš Ude, Mirko Raković, and Andrej Gams. Compliant movement primitives in a bimanual setting. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 365–371. IEEE, 2017.

[48] Giovanni Franzese, Leandro de Souza Rosa, Tim Ver-

burg, Luka Peternel, and Jens Kober. Interactive imitation learning of bimanual movement primitives. *IEEE/ASME Transactions on Mechatronics*, 2023.

[49] Aditya Ganapathi, Priya Sundaresan, Brijen Thananjeyan, Ashwin Balakrishna, Daniel Seita, Jennifer Grannen, Minho Hwang, Ryan Hoque, Joseph E Gonzalez, Nawid Jamali, et al. Learning dense visual correspondences in simulation to smooth and fold real fabrics. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11515–11522. IEEE, 2021.

[50] Fabio Amadio, Adrià Colomé, and Carme Torras. Exploiting symmetries in reinforcement learning of bimanual robotic tasks. *IEEE Robotics and Automation Letters*, 4(2):1838–1845, 2019.

[51] Arpit Bahety, Shreeya Jain, Huy Ha, Nathalie Hager, Benjamin Burchfiel, Eric Cousineau, Siyuan Feng, and Shuran Song. Bag all you need: Learning a generalizable bagging strategy for heterogeneous objects, 2023.

[52] Jennifer Grannen, Yilin Wu, Brandon Vu, and Dorsa Sadigh. Stabilize to act: Learning to coordinate for bimanual manipulation. In *Conference on Robot Learning*, pages 563–576. PMLR, 2023.

[53] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022.

[54] Aravind Sivakumar, Kenneth Shaw, and Deepak Pathak. Robotic telekinesis: Learning a robotic hand imitator by watching humans on youtube. In *Robotics: Science and Systems*, 2022.

[55] Haoyu Xiong, Quanzhou Li, Yun-Chun Chen, Homanga Bharadhwaj, Samarth Sinha, and Animesh Garg. Learning by watching: Physical imitation of manipulation skills from human videos. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7827–7834. IEEE, 2021.

[56] Annie S Chen, Suraj Nair, and Chelsea Finn. Learning generalizable robotic reward functions from" in-the-wild" human videos. *arXiv preprint arXiv:2103.16817*, 2021.

[57] Lin Shao, Toki Migimatsu, Qiang Zhang, Karen Yang, and Jeannette Bohg. Concept2robot: Learning manipulation concepts from instructions and human demonstrations. *The International Journal of Robotics Research*, 40(12-14):1419–1434, 2021.

[58] Yecheng Jason Ma, William Liang, Vaidehi Som, Vikash Kumar, Amy Zhang, Osbert Bastani, and Dinesh Jayaraman. Liv: Language-image representations and rewards for robotic control. *arXiv preprint arXiv:2306.00958*, 2023.

[59] Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022.

[60] Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

[61] Ilija Radosavovic, Tete Xiao, Stephen James, Pieter Abbeel, Jitendra Malik, and Trevor Darrell. Real-world robot learning with masked visual pre-training. In *Conference on Robot Learning*, pages 416–426. PMLR, 2023.

[62] Yu Rong, Takaaki Shiratori, and Hanbyul Joo. Frankmocap: A monocular 3d whole-body pose estimation system via regression and integration. In *IEEE International Conference on Computer Vision Workshops*, 2021.

[63] Priyanka Mandikal and Kristen Grauman. Dexvip: Learning dexterous grasping with human hand pose priors from video. In *Conference on Robot Learning*, pages 651–661. PMLR, 2022.

[64] Kenneth Shaw, Shikhar Bahl, and Deepak Pathak. Videodex: Learning dexterity from internet videos. In *Conference on Robot Learning*, pages 654–665. PMLR, 2023.

[65] Kenneth Shaw, Shikhar Bahl, Aravind Sivakumar, Aditya Kannan, and Deepak Pathak. Learning dexterity from human hand motion in internet videos. *The International Journal of Robotics Research*, page 02783649241227559, 2024.

[66] Aditya Kannan, Kenneth Shaw, Shikhar Bahl, Pragna Mannam, and Deepak Pathak. Deft: Dexterous finetuning for real-world hand policies. *CoRL*, 2023.

[67] 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.

[68] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F. Fouhey. Understanding human hands in contact at internet scale. *CoRR*, abs/2006.06669, 2020. URL https://arxiv.org/abs/2006.06669.

[69] Homanga Bharadhwaj, Abhinav Gupta, Shubham Tulsiani, and Vikash Kumar. Zero-shot robot manipulation from passive human videos, 2023.

[70] Shikhar Bahl, Abhinav Gupta, and Deepak Pathak. Human-to-robot imitation in the wild. 2022.

[71] Russell Mendonca, Shikhar Bahl, and Deepak Pathak. Structured world models from human videos. 2023.

[72] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*, 2016.

[73] Bruno Siciliano, Oussama Khatib, and Torsten Kröger. *Springer handbook of robotics*, volume 200. Springer, 2008.

[74] Kevin M Lynch and Frank C Park. *Modern robotics*. Cambridge University Press, 2017.

[75] Yves Guiard. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of motor behavior*, 19(4):486–517, 1987.

[76] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr – modulated detection for end-to-end multi-modal understanding, 2021.

[77] Nicolas Mansard, Olivier Stasse, Paul Evrard, and Abderrahmane Kheddar. A versatile generalized inverted kinematics implementation for collaborative working humanoid robots: The stack of tasks. In *2009 International conference on advanced robotics*, pages 1–6. IEEE, 2009.

[78] Freek Stulp and Olivier Sigaud. Path integral policy improvement with covariance matrix adaptation. *arXiv preprint arXiv:1206.4621*, 2012.

[79] Jürgen Sturm, Advait Jain, Cyrill Stachniss, Charles C Kemp, and Wolfram Burgard. Operating articulated objects based on experience. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2739–2744. IEEE, 2010.

[80] Yiannis Karayiannidis, Christian Smith, Francisco Eli Vina Barrientos, Petter Ögren, and Danica Kragic. An adaptive control approach for opening doors and drawers under uncertainties. *IEEE Transactions on Robotics*, 32 (1):161–175, 2016.

[81] Roberto Martín-Martín and Oliver Brock. Cross-modal interpretation of multi-modal sensor streams in interactive perception based on coupled recursion. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3289–3295. IEEE, 2017.

[82] Ben Abbatematteo, Stefanie Tellex, and George Konidaris. Learning to generalize kinematic models to novel objects. In *Proceedings of the 3rd Conference on Robot Learning*, 2019.

[83] Yian Wang, Ruihai Wu, Kaichun Mo, Jiaqi Ke, Qingnan Fan, Leonidas J Guibas, and Hao Dong. Adaafford: Learning to adapt manipulation affordance for 3d articulated objects via few-shot interactions. In *European Conference on Computer Vision*, pages 90–107. Springer, 2022.

[84] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *European Conference on Computer Vision (ECCV)*, 2018.

[85] Reiner Birkl, Diana Wofk, and Matthias Müller. Midas v3. 1–a model zoo for robust monocular relative depth estimation. *arXiv preprint arXiv:2307.14460*, 2023.

[86] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. *arXiv preprint arXiv:2401.10891*, 2024.

[87] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PMLR, 2021.

[88] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 2022.

[89] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920, 2015. doi: 10.1109/CVPR.2015.7298801.

Appendix

A. Screw action with Left-Hand Trajectory

Fig. 8 depicts three steps of a robot execution with a non-empty left-hand trajectory. Since ScrewMimic defines the screw axis of manipulation as the relative motion between both hands, the absolute motion of one of them does not affect the motion generated from the same screw action. We consider the *actuation* part of the bimanual manipulation to be the effect of this relative motion between hands rather than the absolute motion of them.

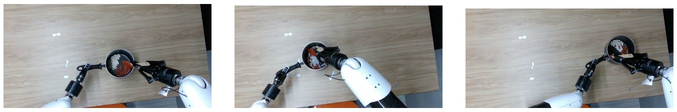


Fig. 8. **Screw Action execution with left-hand trajectory** Three steps of a robot execution of a `stir` task with non-zero left-hand motion. Since ScrewMimic focuses on the generation of relative motion between hands, the same screw action can be used even when there exists any absolute motion of one of them (left hand).

B. Hyperparameters

TABLE V
Screw Action Prediction Model Hyperparameters

Hyperparameters	Value
train epochs	2000
batch size	16
optimizer	Adam
learning rate	1e-3
pointcloud encoder	PointNet [72]
number of points	2048
layer-activations	ReLU
Screw Axis Regression	
Architecture	Conv1d (64, 128, 1024) + FC (1024, 512, 256, 6)
Loss	MSE
Grasp Contact Segmentation	
Architecture	Conv1d (64, 128, 128, 512, 2048, 256, 256, 128, 3)
Loss	negative log likelihood

C. Using Pretrained PointNet Model

We conduct experiments to analyze if using a pre-trained PointNet model helps to better generalize to new objects as compared to training from scratch. We pretrain a PointNet model on the ModelNet-40 dataset [89] for the classification task. We then use the pretrained feature encoder and fine-tune it on the screw axis prediction task. Finally, we compare this model (*M2*) to our original model that was trained from scratch (*M1*) on the screw action prediction task. The test set consists of 4 different bottles (1 bottle seen during training and 3 unseen bottles) as shown in Fig. 9 The test set consists of 10 poses for each of the four bottles. We use two metrics to evaluate their performance:

- Mean distance between predicted and ground truth screw axis (in meters)
- Mean angle between the predicted and ground truth screw axis (in degrees)

As shown in Table VI, with a pre-trained PointNet, we indeed observe an improvement in the screw axis prediction on novel objects, although the performance on the training object remains the same. While this would not affect the exploration in case of the training object, it would lead to more efficient optimization in the CEM phase for novel objects and better generalization capabilities of the policy

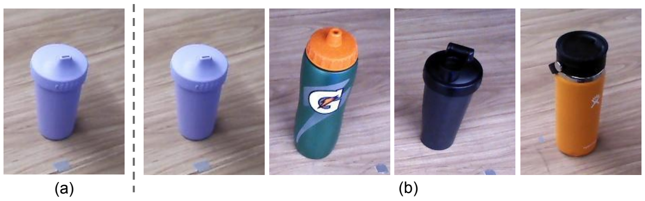


Fig. 9. (a) Training bottle. (b) Testing Bottles

TABLE VI
PointNet trained from scratch (M1) vs Pretrained+Finetuned (M2)

	M1	M2
Blue Bottle (Training bottle)	0.01 m, 2.1°	0.01 m, 2.0°
Green Bottle (with orange cap)	0.03 m, 3.6°	0.01 m, 2.3°
Black Bottle	0.05 m, 5.3°	0.02 m, 3.8°
Orange Bottle (with black cap)	0.09 m, 6°	0.03 m, 4.2°

D. Robustness to Noisy Demonstrations

To analyze the robustness of ScrewMimic to noisy human demonstrations we conduct the following two experiments:

a) ***Artificially adding increasing amounts of noise (controlled study):*** In the first experiment, we investigate how well ScrewMimic can adapt when increasing amounts of artificial noise are introduced to a trajectory. We focus on a specific task — `open bottle`. We manually annotate the ground truth screw axis and compute the corresponding noise-free ground truth hand trajectory (shown in green in Fig. 10). This trajectory corresponds to the trajectory of an acting hand relative to a reference hand. We introduce five different levels of noise to the ground truth hand trajectory, affecting both position and orientation, and observe the changes in the screw axis computed by ScrewMimic with increasing noise levels. These trajectories and their respective screw axes are illustrated in Fig. 10, where colors transitioning from light to dark depict the sequence of actions from start to finish. We only visualize the positions (and not the orientations) for clarity. We created 20 noisy trajectories for each noise level, resulting in 100 test trajectories. We use two metrics to evaluate the performance for each noise level: a) mean distance error between predicted and ground truth screw axes (in meters), and b) mean angle error between the predicted and ground truth screw axis (in degrees).

Table VII and Fig. 10 show the results of our experiment. We observe that the accuracy of the screw axis detected by

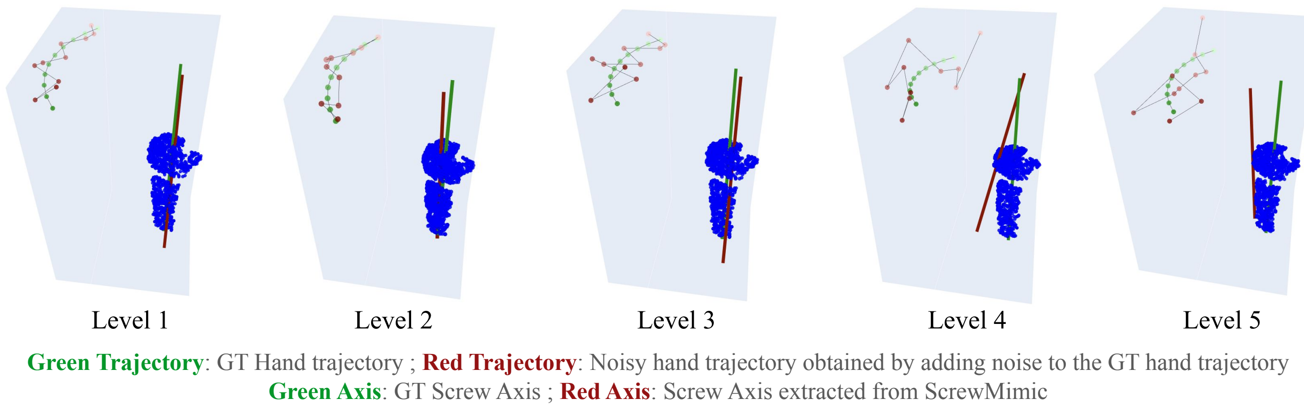

Level 1
Level 2
Level 3
Level 4
Level 5
Green Trajectory: GT Hand trajectory ; Red Trajectory: Noisy hand trajectory obtained by adding noise to the GT hand trajectory
Green Axis: GT Screw Axis ; Red Axis: Screw Axis extracted from ScrewMimic

Fig. 10. **ScrewMimic's axis extraction with increasingly noisy demonstrations.** The five levels represent increasing noise applied to the ground truth hand trajectory (position and orientation). Comparing each screw axis with the ground truth screw axis in this figure and the numbers in Table VII, and performing the fine-tuning experiment with the axis inferred from the highest noise level, we observe that although ScrewMimic does suffer from increasing noise in the hand trajectories, it is able to extract an axis sufficiently accurate for fine-tuning.

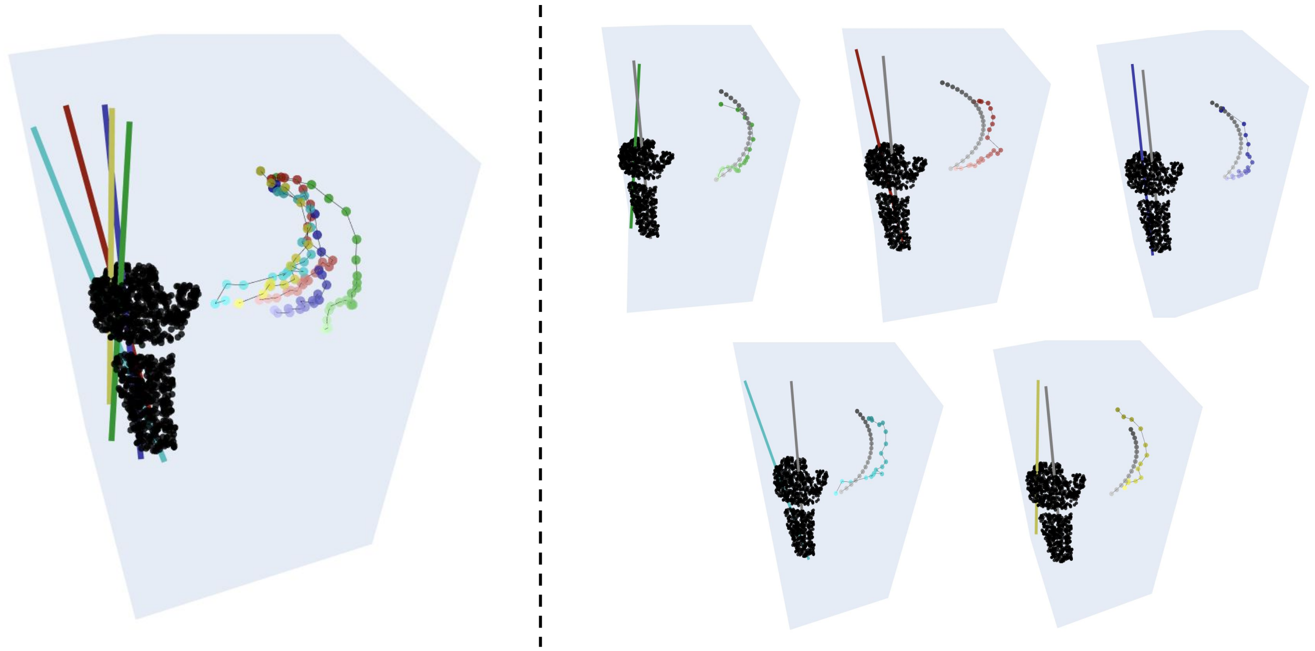
TABLE VII
ScrewMimic's Axis Extraction with Increasingly Noisy Demonstrations

	Level 1 pos=$N(0, 1.0cm)$ orn=$N(0, 2.5°)$	Level 2 pos=$N(0, 1.5cm)$ orn=$N(0, 5.0°)$	Level 3 pos=$N(0, 2.0cm)$ orn=$N(0, 7.5°)$	Level 4 pos=$N(0, 2.5cm)$ orn=$N(0, 10.0°)$	Level 5 pos=$N(0, 3.0cm)$ orn=$N(0, 12.5°)$
Distance between GT axis and Extracted axis (cm)	$0.5cm \pm 10^{-5}$	$0.8cm \pm 10^{-4}$	$1.1cm \pm 10^{-5}$	$1.7cm \pm 10^{-4}$	$2.1cm \pm 10^{-2}$
Angle between GT axis and Extracted axis (degrees)	$4.0° \pm 2.0°$	$6.5° \pm 3.0°$	$9.4° \pm 5.0°$	$11.1° \pm 8.5°$	$13.1° \pm 6.0°$

ScrewMimic declines as we increase the noise in the hand trajectories. To test if ScrewMimic can perform a successful fine-tuning even with the highest noise level, we conduct the following experiment: we use the axis inferred by ScrewMimic from the trajectory in level 5 to bootstrap the ScrewMimic fine-tuning step. We observe that even in this adversarial condition, success is achieved after 4 epochs and 21 episodes. This is comparable to the performance of ScrewMimic on the bottle opening task in our original experiments as shown in Table I. Thus, ScrewMimic is able to "clean up" the noise and extract an axis good enough to bootstrap the fine-tuning step. This shows that even though the quality of the screw axis inferred by ScrewMimic declines with increasing noise, the axis still proves adequate for initiating the fine-tuning process.

*b) **Naturally occurring noise (perceptual noise):*** In the second experiment, we evaluate the robustness of ScrewMimic to naturally occurring noise when perceiving human demonstrations. We collect five different human demonstrations for the bottle opening task for the same pose of the bottle. Variability in the trajectories arises from differences in individual demonstrations and noise from the hand-pose detector. We compare the screw axis computed by ScrewMimic for these five demonstrations to a manually annotated ground truth axis. Fig. 11 shows the qualitative results for this experiment: Fig. 11 (a) on the left helps visually compare the five human trajectories and the corresponding screw axes as computed by ScrewMimic. Note that the trajectory corresponds to the trajectory of the acting hand (right hand in this case) relative to the reference hand (left hand). Fig. 11 (b) on the right shows a comparison of the trajectory and computed screw axis with the ground truth trajectory and screw axis for each of the five human demonstrations.

Table VIII shows the quantitative results of the distance error and angle error between the axis computed by Screwmimic and the ground truth axis. These axis errors are comparable to the axis errors for the `open bottle` task in our original experiments (refer to Sec. V Q1) which are $1.42cm$ mean distance error and $11.2°$ mean angle error. As can be seen in the first row of Table I in the main paper, ScrewMimic's fine-tuning process is able to correct the initial noisy axis and succeed at the task for the most part. Since the errors in the axes as shown in Table VIII are comparable to the error in our original `open bottle` experiments, we can infer that ScrewMimic would be able to fine-tune these noisy axes. This shows that despite the diversity in the trajectories due to variations in demonstrations and detection noise, ScrewMimic consistently infers a screw axis with an accuracy that proves adequate for initiating the fine-tuning process.



(a) Comparing Screw Axis extracted from ScrewMimic for 5 human demonstrations

(b) Comparing each screw axis extracted by ScrewMimic with the ground-truth screw axis

Fig. 11. **Screw axis extracted for five human demonstrations.** (a) Five human trajectories and their corresponding screw axes extracted by ScrewMimic. (b) Individual trajectories and extracted screw axis along with the ground truth trajectory and screw axis. Despite the diversity in the trajectories due to variations in demonstrations and detection noises, ScrewMimic is able to extract a screw axis sufficiently accurate for fine-tuning.

TABLE VIII
ANALYZING SCREW AXIS EXTRACTED FROM FIVE HUMAN DEMONSTRATIONS

	Demo 1	Demo 2	Demo 3	Demo 4	Demo 5
Distance between GT and Extracted axes (cm)	0.91 cm	1.26 cm	1.45 cm	1.33 cm	1.10 cm
Angle between GT axis and Extracted axis (degrees)	6.0°	6.5°	6.4°	12.3°	8.7°

E. Additional Generalization Results

Fig. 12 shows results for the rest of the four tasks for the experiment described in Sec. V Q2). Columns 1 and 5 in the figure show that SCREWMIMIC can obtain reasonable screw actions for new objects and can fine-tune it to generalize to new objects. Furthermore, the last column shows that retraining the screw action prediction model with the successful screw action can improve the prediction model as well. This helps create a self-learning loop where the robot can continually expand its manipulation capabilities to new objects.

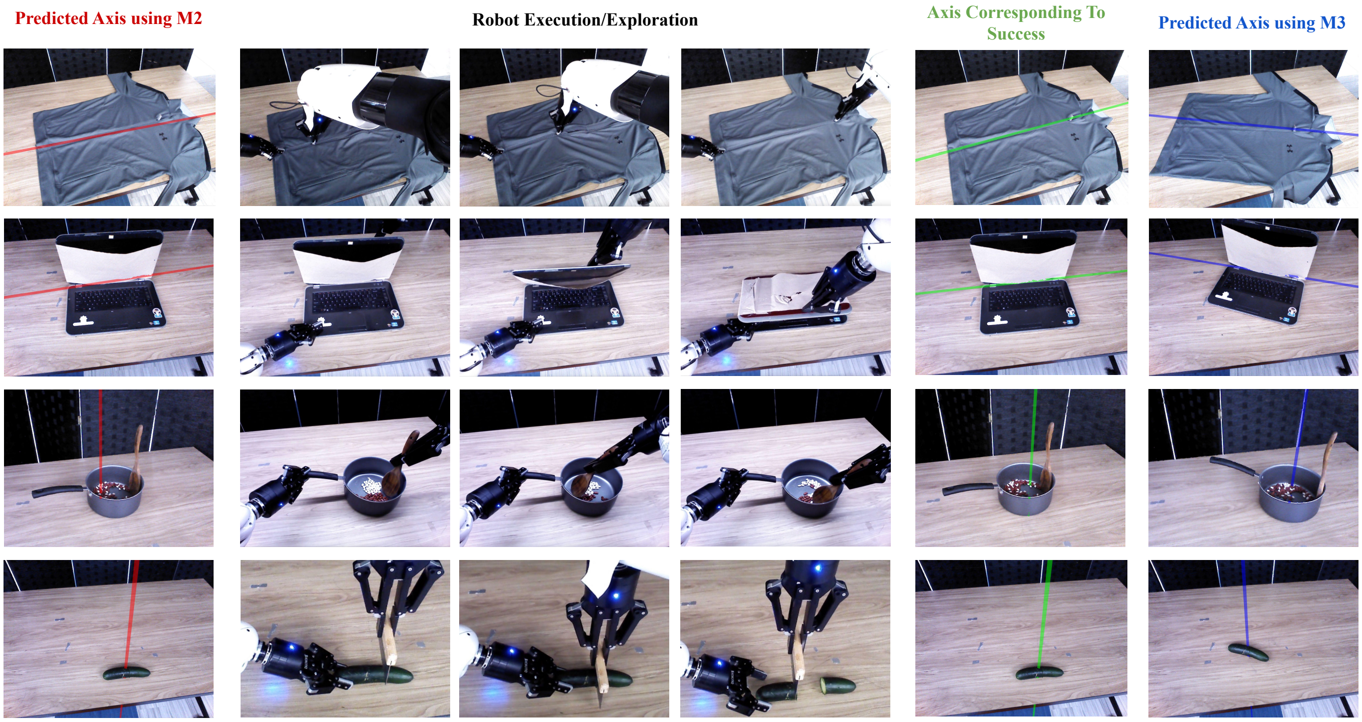


Fig. 12. **Generalization to new objects.** The first column shows the axis predicted by M2, the model trained on the corrected screw action for the first object. Columns 2-4 show snapshots from an episode in the fine-tuning stage. Column 5 shows the axis corresponding to the successful trajectory obtained during the aforementioned process. Column 6 shows the predicted axis from the prediction model re-trained on the corrected axis (M3). Thus, ScrewMimic can obtain reasonable screw action predictions and fine-tune them to generalize to new objects.