Improving Real-World Data Collection with Data Glove for Imitation Learning on Low-Cost Dexterous Robotic Hand

Rajiv Bhardawaj†, Sai Bommisetty †, Jaú Gretler†, Fabio Hübel†,
Richard Kuhlmann†, Florian Lange†, Anton Pollak†, and Robert K. Katzschmann[1]*

***Abstract*— Imitation learning is a powerful and promising technique that enables robots to directly learn policies by observing human experts. The amount and quality of the training data is crucial for a successful implementation. However, collecting a sufficient amount of high quality training and testing data can be a time-consuming and expensive task. This paper proposes and implements several measures to simplify the process of data collection and ensure the data's suitability for training a working policy. The focus lies on the task of object manipulation. Precisely, an anthropomorphic, dexterous hand is used as the gripper to enable exact mimicking of the human hand's movement. Pressure sensors in the finger tips are used to accurately detect grasps. Simultaneously, a retargeting algorithm and a Gaussian Process enable precise control and teleoperation of the robotic hand. Online data visualization and ROS2 node management is simplified by a Graphical User Interface (GUI). Finally, visualization methods are presented to analyze the training progress and results of imitation learning.**

I. Introduction

A. Motivation

Building an anthropomorphic hand capable of human-like grasping and manipulation of objects is considered one of the most difficult robotic problems [1]. Thus, most robotic grippers today rely on simplified designs which perform well on specific tasks. Many of them fail to match a human hand's versatility and dexterity, which is required to capture the full potential of manipulation in industrial and private settings [2]. The trade-off between design complexity and control complexity is a key challenge in creating an anthropomorphic highly dexterous hand. In the past decade, simple designs with reduced control complexity have achieved best results [1]. However, recent developments in powerful learning-based controls, such as Imitation Learning (IL), open the door for increased design complexity [3].

Although control policies trained with imitation learning exhibit promising results, it is up to the current state of research to explore and extend the policies' capabilities. While doing so, researchers face various problems. Most prominent on the hardware end are the high costs of robotic hands. Regarding software, as for many machine learning tasks, sufficient data quantity and quality is inevitable. Creating the needed quantities of expert data for imitation learning inherently requires a high amount of human labor [3]. Moreover, the quality of data, specifically its informativeness regarding the underlying process, plays a crucial role in controlling real-world systems autonomously, as the input data serves as the sole medium of perception for deciding what controls to provide [4]. In light of these challenges, there is a clear motivation to develop a cost-effective, anthropomorphic robotic hand system tailored to enable state-of-the-art research in imitation learning for highly dexterous robotic hands.

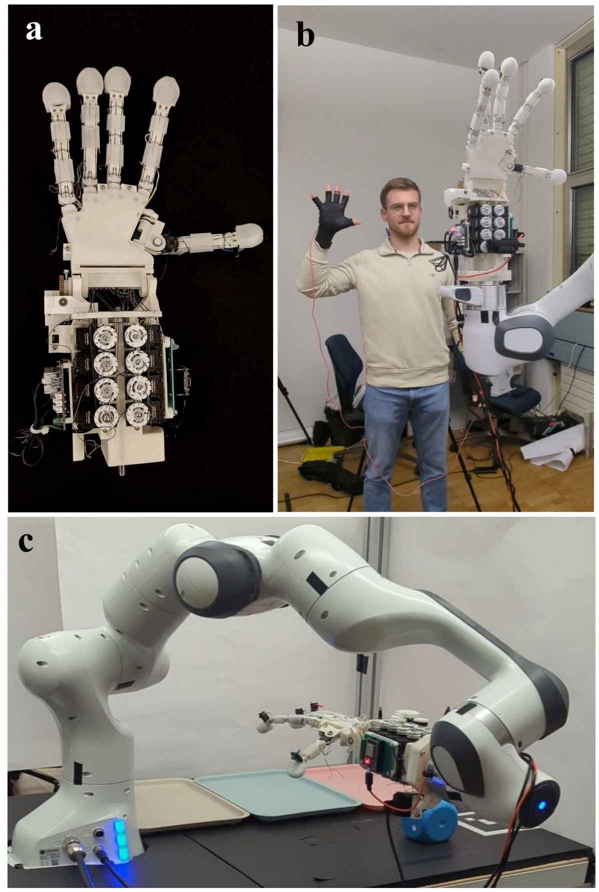


Fig. 1: Robotic System used in the paper. a: 16 Degrees of Freedom (DOF) Robotic Hand, b: Robotic Hand in teleoperation, c: Robotic Hand mounted on a FRANKA manipulator for autonomous color based cube-sorting with IL policy

B. Related Work

Imitation learning (IL) is a prominent research area for controlling dexterous robotic hands in autonomous manipula-

[1]Soft Robotics Lab, IRIS, D-MAVT, ETH Zurich, Switzerland
† Equal contribution.
∗ Corresponding author: rkk@ethz.ch

tion tasks [5]. A central requirement across IL approaches is the availability of high-quality expert data, enabling systems to imitate human behavior effectively. Behavioral cloning (BC), a widely used variant of IL, necessitates datasets containing state-action pairs to learn mappings through supervised learning. Data acquisition methods for IL often involve trade-offs between the quality and informativeness of the data and the speed of collection.

Strategies leveraging simulated environments for data collection, as proposed by *Wong et al.* [6] and *Mandlekar et al.* [7], offer advantages such as reduced reliance on expensive hardware, risk-free policy testing, and rapid data acquisition. However, transferring policies from simulation to real-world systems remains challenging due to the sim-to-real data quality gap. Data collection methodologies differ based on the role of the robotic hand during the process. In the most extreme case, the robotic hand is entirely absent from data collection, relying solely on videos of humans performing tasks. Although this approach enables rapid acquisition and the use of existing online video datasets, the resulting data quality is often suboptimal [5].

To address these challenges, *Wei et al.* [5] introduced a hand-over-hand method using a low-cost anthropomorphic robotic hand (HIRO). This system allows users to interact directly with the robotic hand, eliminating the need for data gloves or complex teleoperation hardware. It avoids the challenges of retargeting human motion to less dexterous robotic hands and provides direct tactile feedback, resulting in high-quality data. However, this setup does not support attachment to a robotic arm, limiting its use in fully autonomous manipulation tasks. *Kukliński et al.* [8] implemented teleoperation using a data glove with a three-finger gripper mounted on a UR5 robotic arm. While data gloves may seem intuitive, users reported that data collection was more effective and user-friendly with a control peg (hand-held device for 6D motion tracking) [8].

C. Contributions

Our work aims to solve several problems in collecting high quality real-world data efficiently using a highly dexterous anthropomorphic robotic hand for IL on manipulation tasks. We provide a low-cost robotic hand system specifically engineered for data collection using data gloves. Our system includes:

1) 16 Degrees of Freedom (DOF) robotic hand design including **sensitive force sensors**
2) Precise mapping of human hand pose on to robotic hand using **Retargeting** and **Gaussian Process Models** to cope with kinematic non-linearity, improving grasping precision and data quality
3) **Graphical User Interface** for efficient work processes during data collection
4) **Data visualization tool** to better understand collected data

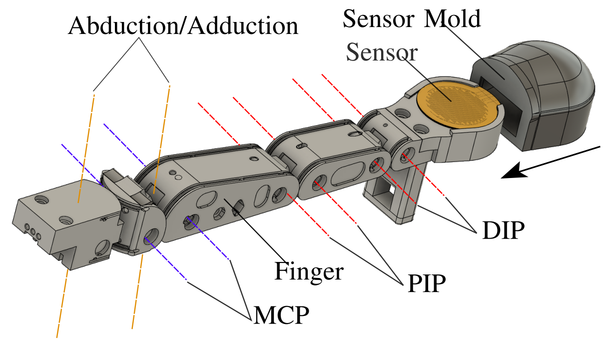


Fig. 2: CAD model of a finger with silicon mold for the sensor, showing the assembly direction of the mold and rotation axes of the rolling contact joints

II. Design

A. Design Overview

For the research presented in this paper, a 16 DOF tendon-driven robotic hand is used, which is shown in Figure 1. The hand incorporates five fingers with three DOF each, one sensor in every finger tip and an actuated, worm-gear driven wrist. In contrast to the other fingers, the lowest link of the thumb is comparably large. This resembles the ball of the thumb in the human hand. The key ideas behind this design are combining low-cost manufacturability with an anthropomorphic shape. To mimic the shape of a human hand, pinkie and index finger are placed at an angle. Low production cost is achieved by fabricating most parts from 3D-printed and off-the shelf materials. The most expensive components are the 16 servo motors (Dynamixel XC330-T288-T).

The joints in the fingers are tendon-actuated. This actuation method enables placing the motors outside of the hand, thus saving space and allowing more freedom in design. For the tendons, fishing line is used, which is a strong, light, thin, yet cheap and accessible material. The only joint that is not actuated by a tendon is the wrist. Here, a worm gear is used due to its self-locking characteristic. This enables large payloads and keeping the current position without permanently requiring torque from the motor.

B. Fingers

Each finger has 3 DOF, which are inspired by a human finger and their naming is accordingly: There is one joint for the abduction/adduction (lateral movement), as well as two for flexing and extending the finger, the metacarpophalangeal (MCP) joint and the coupled proximal interphalangeal (PIP) and distal interphalangeal (DIP) joints. The realization of these joints is shown in Figure 2. The yellow lines represent the rotation axes for the abduction/adduction movement, the blue lines the axes of the MCP and the red lines the axes of the PIP and DIP joints. The PIP and DIP joints are coupled, meaning they can only be actuated together.

The joints are implemented as rolling contact joints. This type of joint combines low friction with safety, as the joints do not break under high loads, but rather dislocate.

The movement in rolling contact joints incorporates two simultaneous rotations. Therefore, there are two axes for every joint in Figure 2. Additionally, they can be fabricated from 3D-printed parts and fishing line without requiring extra parts, i.e. bearings. However, they introduce non-linearity to the system kinematics, resulting in tendon slack and play in the finger position. This happens when one motor actuates both rotational directions of a joint, as is the case in this hand for cost reasons (motors are the most expensive parts). In this case, the actuating tendons are called a protagonist and its antagonist and are both mounted on the same motor. When the motor rotates, one tendon is pulled and the other one is released, resulting in movement into the direction of the pulled tendon. The geometry of the rolling contact joints leads to more length of the antagonist being released than length of the protagonist pulled in, leading to slack and play in the joint. One method to address this challenge is to separate the spool to which the tendons are attached in two introducing a rotational DOF between the spools. The spools are connected by a preloaded spring, and each tendon has an individual connection. If one of the tendons has play, torque from the spring induces a relative rotation between the spools, re-tightening the loose tendon. This mechanical measure cannot compensate for the non-linearity entirely. Therefore, sophisticated control algorithms are required to enable precise finger movements, which are necessary for high-quality data for imitation learning.

C. Sensing

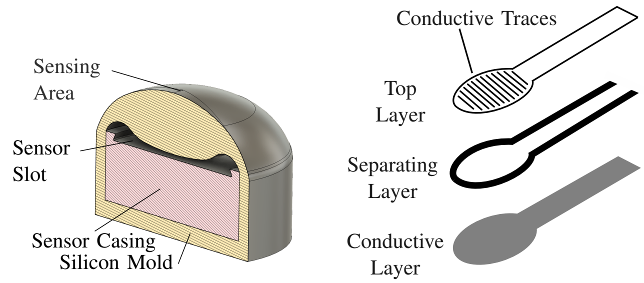
1) Sensors: Tactile data is highly relevant for manipulation tasks, as it provides operator feedback during data collection, as well as additional highly informative data input to the IL algorithm (i.e. grasp success detection) [5]. To enable pressure sensing in each finger tip, shunt-mode force sensing resistors (FSRs) [1] are integrated into each finger tip. The flat sensing area has a diameter of $12.7\,\mathrm{mm}$ and can theoretically sense applied forces in the range of $100\,\mathrm{g}$ to $10\,\mathrm{kg}$. This sensor type was chosen due to its low cost ($<$ 10CHF/unit), high robustness [9] and maturity.

2) Measurement circuit: The circuit consists of five voltage dividers, each containing a $10\,\mathrm{M\Omega}$ in series with the variable resistance of an FSR sensor. Applying a force onto the sensor decreases its resistance, leading to a voltage drop, which is measured through the analog-pins of an ESP32-Wroom microcontroller.

3) Silicone Fingertips: It was observed that the sensors' minimal force baseline was too high for it to register light touches of the fingertips. This is due to their operating principle based on two polymer sheets with one of them bearing a conductive pattern and the other one a semi-conducting polymer (see Figure 3(b)): Applying a force presses the sheets against each other. After overcoming a certain force threshold, the sensor's resistance drops with increasing force [9]. To mitigate this, a silicon mold[2] was

[1]SparkFun SEN-09375 FSR sensors

[2]DragonSkin Fast, Shore Hardness 10A



((a)) CAD Design of the silicon mold fingertip

((b)) Structure of a shunt-mode FSR sensor (modeled after [9] and [10])

Fig. 3: Fingertip and sensor design

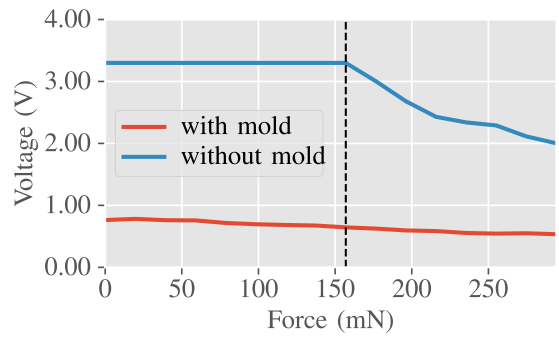


Fig. 4: Fingertip sensitivities for 15 datapoints with and without silicon mold

cast to cover the fingertips. It applies a constant (though not exactly known) *pre-pressure* to the FSR sensors. The pre-pressure varies between 30-40$\%$ of the analog range for each mold.

It was observed that the sensors' sensitivity was highest in the center, dropping towards the edges. Therefore, the sensor-facing side of the silicon mold is modeled as an ellipsoid (see Figure 3(a)), concentrating the force in its center. This simultaneously increases the resulting pressure acting on the sensor, since the outside force is acting on a more confined area of the sensor. Finally, the mold's outside is also ellipsoid-shaped, enabling sensing forces from a wide range of attack angles (forces up to $60°$ were measurable).

Thus, the purpose of the designed fingertips was fourfold:

1) Increase friction to improve grasping
2) Apply pre-pressure to the sensors
3) Concentrate the outside force in the sensor center
4) Provide an ellipsoid outside interface

These measures resulted in a drop in the minimal sensing force to as low as $F_{min} = 4\,\mathrm{g} \cdot 9.81\frac{\mathrm{m}}{\mathrm{s}^2} \approx 49\,\mathrm{mN}$, as shown in Fig. 4. While trading off some of the sensitivity at higher forces, the sensor becomes responsive at the aforementioned very low forces (versus approx. $150\,\mathrm{mN}$ for the sensor without mold, see black dashed line Figure 4). Furthermore, the response is approximately linear for all measured datapoints.

4) Calibration: The pre-pressure applied to each FSR sensor manifests itself as different offsets in the analog signal

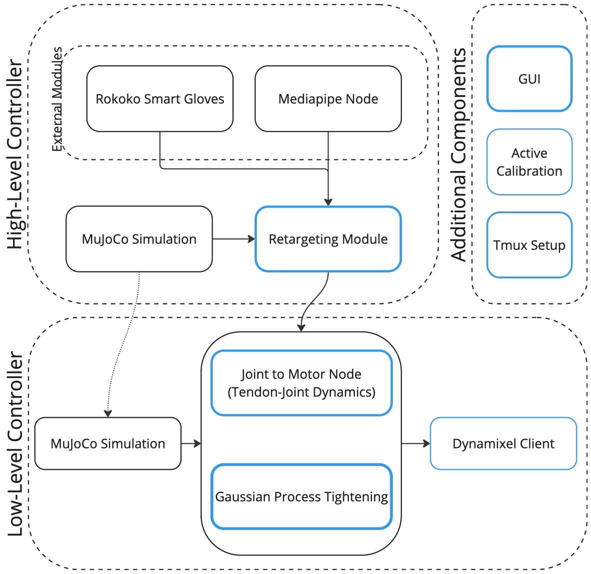


Fig. 5: Diagram of Control and Testing Architecture, with outlined contributions by weight

reading. Thus, a sensor calibration routine was introduced, which, after startup of the ESP32, measures the pre-pressure offset for 10^4 measurement cycles, computes the mean and subtracts this mean offset for all subsequent sensor measurements.

5) Filtering: To reduce the noise observed in the calibrated sensor readings, a Moving Average Filter is used to filter out spikes.

6) ROS2 Integration: The `MicroROS` framework [11] provided integration of the sensor data into ROS2, publishing the raw sensor data to the host computer.

III. Control

Effective imitation learning requires efficient subsystem integration and repeatable performance across multiple mediums: model, simulation, and the physical system, as well as any external modules. To ensure accurate performance of the robot across different modules and configurations, a hierarchical control structure is assumed.

The higher-level controller structure operates at the $x \in \mathbb{R}^{16}$ joint-level, which is the granularity at which other higher-level modules, such as the retargeter, teleoperation, simulation, and imitation learning policy operate. Once a joint state command is issued, a low-level controller utilizes the tendon-driven actuation model to translate the command into motor positions for each DOF. This hierarchical approach allows subsystems to be adaptable for different control mediums. As such, teleoperation can be efficiently performed using the Rokoko smart gloves[3] or the webcam using the mediapipe pipeline [12], just as well as internal modules integrated with the Multi-Joint dynamics with Contact (MuJoCo) Simulation environment [13].

[3]Motion Capture system for hand-tracking and motion, by the Rokoko animation technology company

To ensure the reliability of the low-level controller, an automatic calibration method was developed which utilizes a guiding device along with mechanical hard stops to pull each finger into the desired position reliably and automatically.

A. Retargeting

The retargeting module serves as a critical element in the imitation learning pipeline as its performance affects three major components of the resulting policy: the quality of the collected data, the minimization of the sim-to-real gap, and the accuracy of the applied kinematic model to the true dynamics.

Rokoko smart gloves are used to provide input for the robot hand and Franka arm movements in the form of MANO Keypoints [14], which are subsequently mapped to the low-level controllers. The retargeting module, inspired by [15] and [16], aims to optimize the retargeting error of 17 key-vector pairs between the input MANO keypoints, and a simplified kinematics model of the hand. Each key-vector pair is assigned a priority to allow the retargeter to maximize a designed control objective.

We introduce an additional pre-processing step for the MANO keypoints to help maintain the consistency of the input data across different users, improve the robustness against positioning differences in the gloves across each session, and support more precise motion tracking across the morphological differences, by applying a series of transformations. This results in better abduction and adduction, a motion behavior not performed in the original paper, as well as other performance requirements. This application can be further extended to non-anthropomorphic morphologies as well, permitting the use of control and learning techniques using this pipeline.

B. Gaussian Processes for sim-to-real gap

Similar to the accuracy requirement for the retargeting step in the teleoperation pipeline, it is imperative that the mapping of the motor positions to the fingers' joint angles is precise when collecting data for imitation learning.

An inherent flaw of a rolling contact joint based design of the fingers is the introduction of non-linear tendon slack, which worsens as the tendons wear out over time. The general mathematical modeling, omitted here for brevity, does not capture these two highly influential factors, leading to a significant sim-to-real gap (from commanded to observed joint angles) and inaccurate real-world hand poses. Although the introduction of spring-loaded spools (see Sec. II-B) alleviates the issue, they do not fully compensate the slack.

The solution proposed here is a hybrid approach of first-principles and data-driven modeling. In this instance, due to its flexibility and ease of use, a Gaussian Process is used to estimate the residuals of the commanded and observed finger joint angle for each motor using specifically collected data. The immediate benefit associated with this method is an improved range of motion (former modeling did not reach full flexion on the real plant) in addition to the generally more precise finger poses that align with the commanded joint angles. This allowed the pilots of the hand to achieve more articulate and accurate finger movements, speeding up and improving the quality of data collection.

A disadvantage inherent to using a data-driven model is the manual (re-)collection of the data, especially after extensive usage or tendon re-tightening. The hyperparameters (RBF kernel) were chosen to reflect the uncertainty in the manual measurements and the strong correlation of the slack across similar joint angles.

IV. Visualization

Unlike data collection within simulation, real-world data collection often presents challenges such as the time taken to collect each sample, hardware uncertainties, etc. A set of visualization tools allowed to improve the efficiency of data collection while maintaining the quality of data sent for training.

A. Graphical User Interface

By leveraging each subsystem's ability to run passively and independently, a centralized user interface for all stakeholders allows for the following:

- Operation Status and Kill Switches: By displaying the status of each component, along with kill switches, it is possible to sequentially enable each component, such as ensuring accurate retargeting within simulation before transmitting the data to the hand, and reduce the blast radius in case of an emergency.
- Manual Joint Actuation: This tool was helpful during the Gaussian Process Training (see Sec. III-B), and also to debug hardware and motor issues.
- Dynamic Calibration: The operator would be able to dynamically recalibrate the hand in between recordings should they choose to do so. This helped reduce noisy data collected from the hand.
- Sensor Data: A live visualization allows the operator to ensure the data quality of sensors (see Sec. II-C) is consistent during data collection.

An effective graphical user interface helped to reduce the time taken to record data samples, starting from 5 samples per hour to over 75 samples per hour. This also helped to avoid accidental damage to the hardware.

B. Data Quality Studies

The quality of expert demonstrations and the capability of the dataset to capture the states and observation spaces effectively is highly correlated to the success of an imitation learning model. To ensure a high probability of success, qualitative studies were conducted on the collected data to uncover potential blind spots within the demonstrations. A few examples of such studies include:

- Sensor Data Verification: Figure 7 compares the trend of sensor measurements vs when the grasp is performed. The correlation between the two is important for the model to understand how to use the sensor data effectively.

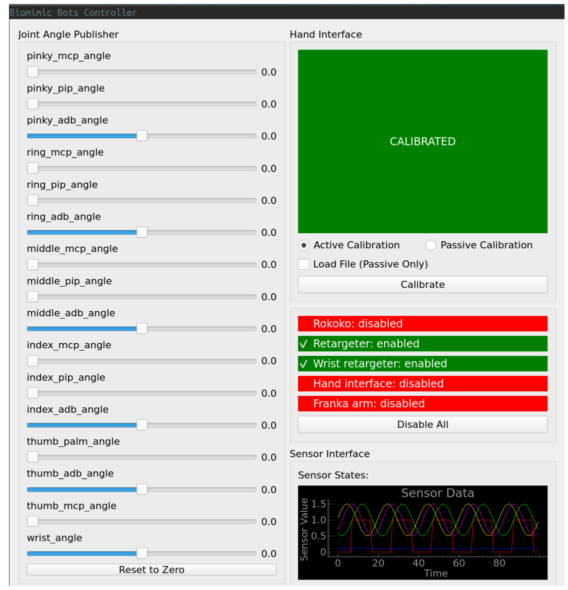


Fig. 6: Graphical User Interface

- Effective Coverage of the State and Action Space: Figure 8 illustrates the various pick-up and drop-off locations within the data. This helped guide future recordings to ensure an even distribution of pick-up and drop-off locations across demonstrations to achieve even coverage of the state and action space, thereby allowing the model to generalize better.

V. Experimental Testing of Data Collection

The methods outlined in this paper were used to collect data for training a color-based cube sorting policy on a robotic manipulator using imitation learning.

Throughout testing, the retargeting functionality performed well, operating in real-time at about $8{-}20\,\mathrm{ms}$[4], and with accurate performance. Deviations in performance were most commonly attributed to approximations inaccuracies of the non-linear model and calibration limitations of the mechanical model.

The control adjustments guided by the Gaussian Process Estimation (Sec. III-B) further increased the teleoperation accuracy of the finger and wrist components. This greatly reduced the frequency of missed grasps, which would have required re-recording, and improved our recording throughput.

The GUI and node management (Sec. IV-A) allowed for faster set-up times, shorter intervals between re-recordings, and expedited the onboarding process for training more users to utilize the platform for imitation learning. The live sensor data provided qualitiative feedback on the quality of grasp detected by the hand during the data-collection process, thereby limiting the possibility of collecting poor-quality data.

The data quality studies (Sec. IV-B) helped correlate poor model performance to possible pit-falls in the demonstra-

[4]running on a 4Ghz Intel Core Ultra 7 CPU

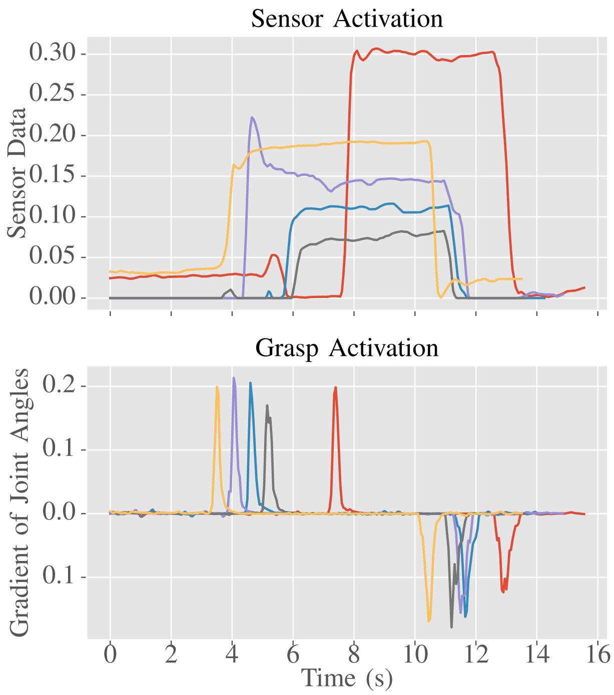


Fig. 7: Comparison of Sensor Data vs Grasp Data for 5 trajectories. Positive Spikes in the Grasp show the pick-up time. Negative spikes show the drop-off time.

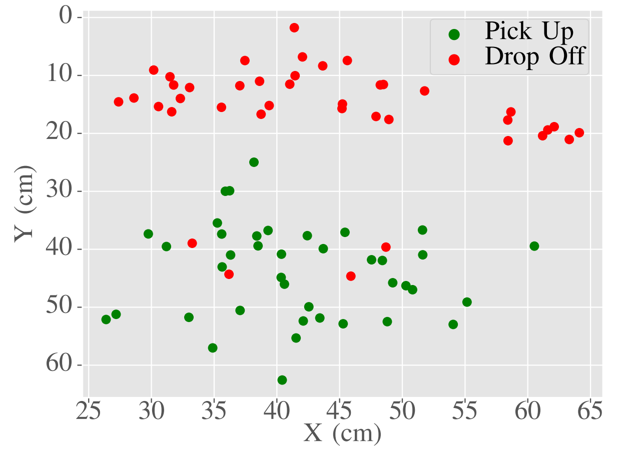


Fig. 8: Plot of various pick up and drop off locations, from 40 randomly sampled demonstrations. The axes limits represent the size of the workspace.

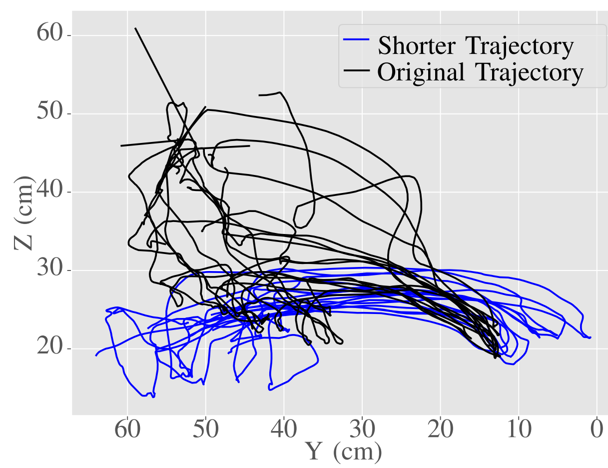


Fig. 9: Franka Trajectory Comparison between original and improved, shorter approach. The workspace is parallel to the X-Y plane

tions. For example, our initial approach for each demonstration was to start each trajectory with the arm far away from the subject of interest, as seen in Figure 9. This lead the model to prioritize the initial movement to the table and ignore the grasping and sorting aspects of the task. An improvement in performance was noticed when we shortened the trajectory and prioritized the grasping and sorting parts, highlighting the uniform weighting of the collected trajectory during policy training.

Overall, the presented measures significantly reduced the time needed to collect a dataset, and resulted in an effective policy to be trained and deployed on the robot.

VI. Conclusion

Our work presents a dexterous anthropomorphic robotic hand system which is engineered to enhance the data collection process for training an imitation learning based policy. Our focus lies on real-world data collection for manipulation tasks utilizing a data glove for human control of the robot. We propose methods to improve data quality and collection efficiency. Therefore, a newly designed low cost 16 DOF dexterous gripper is presented, which offers easy manufacturability and enables capturing tactile data as an additional feature for training. Furthermore, control approaches such as fast online retargeting of hand poses and residual estimation using Gaussian Processes for compensating kinematic non-linearity are implemented to allow precise mapping of the human hand to the robot hand. To streamline the recording process and improve efficiency, a compelling GUI and data analysis approaches are introduced. During testing, the system proves its capabilities by enabling the user to reduce collection time and increase data quality, allowing researchers to focus on creating the most capable control policies using imitation learning.

Future work should set out to test and collect quantitative feedback on the implemented features against a variety of manipulation tasks, such that they can be compared against existing methods of data collection, including those relying on simulation. In addition, the sensor integration could be further refined. The current concept could be downsized to smaller sensors, which would improve the handling of smaller objects. Regarding retargeting, incorporating a learned adaptive weighting scheme over the key-vectors could potentially help dynamically optimize specific retargeting angles based on the action performed.

ACKNOWLEDGMENT

The authors thank the team behind the Real-World Robotics course at ETH Zurich, which greatly enabled this project.

REFERENCES

[1] C. Piazza, G. Grioli, M. Catalano, and A. Bicchi, "A century of robotic hands," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, no. 1, pp. 1–32, May 3, 2019.

[2] Z. Kappassov, J.-A. Corrales, and V. Perdereau, "Tactile sensing in dexterous robot hands — review," *Robotics and Autonomous Systems*, vol. 74, pp. 195–220, 2015.

[3] M. Zare, P. M. Kebria, A. Khosravi, and S. Nahavandi, "A survey of imitation learning: Algorithms, recent developments, and challenges," *IEEE Transactions on Cybernetics*, vol. 54, no. 12, pp. 7173–7186, 2024.

[4] B. Fang, S. Jia, D. Guo, M. Xu, S. Wen, and F. Sun, "Survey of imitation learning for robotic manipulation," *International Journal of Intelligent Robotics and Applications*, vol. 3, pp. 362–369, 2019.

[5] D. Wei and H. Xu, "A wearable robotic hand for hand-over-hand imitation learning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 18 113–18 119.

[6] J. Wong, A. Tung, A. Kurenkov, *et al.*, *Error-aware imitation learning from teleoperation data for mobile manipulation*, 2021. arXiv: `2112.05251 [cs.RO]`.

[7] A. Mandlekar, D. Xu, R. Martín-Martín, Y. Zhu, L. Fei-Fei, and S. Savarese, *Human-in-the-loop imitation learning using remote teleoperation*, 2020. arXiv: `2012.06733 [cs.RO]`.

[8] K. Kukliński, K. Fischer, I. Marhenke, *et al.*, "Teleoperation for learning by demonstration: Data glove versus object manipulation for intuitive robot control," in *2014 6th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, 2014, pp. 346–351.

[9] S. Yaniger, "Force sensing resistors: A review of the technology," in *Electro International, 1991*, 1991, pp. 666–668.

[10] Tekscan, Inc., *How Does a Force Sensing Resistor (FSR) Work? — Tekscan*, https://www.tekscan.com/blog/flexiforce/how-does-force-sensing-resistor-fsr-work.

[11] K. Belsare, A. C. Rodriguez, P. G. Sánchez, *et al.*, "Micro-ros," in *Robot Operating System (ROS): The Complete Reference (Volume 7)*, A. Koubaa, Ed. Cham: Springer International Publishing, 2023, pp. 3–55.

[12] C. Lugaresi, J. Tang, H. Nash, *et al.*, "Mediapipe: A framework for perceiving and processing reality," in *Third workshop on computer vision for AR/VR at IEEE computer vision and pattern recognition (CVPR)*, vol. 2019, 2019.

[13] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2012, pp. 5026–5033.

[14] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together," *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 245:1–245:17, vol. 36, no. 6, Nov. 2017.

[15] A. Sivakumar, K. Shaw, and D. Pathak, *Robotic telekinesis: Learning a robotic hand imitator by watching humans on youtube*, 2022. arXiv: `2202.10448 [cs.RO]`.

[16] O. M. Andrychowicz, B. Baker, M. Chociej, *et al.*, "Learning dexterous in-hand manipulation," *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 3–20, 2020.