frontiers in Robotics and AI

ORIGINAL RESEARCH
published: 15 March 2022
doi: 10.3389/frobt.2022.787291

DeepClaw 2.0: A Data Collection Platform for Learning Human Manipulation

Haokun Wang[1†], Xiaobo Liu[2†], Nuofan Qiu[2], Ning Guo[2], Fang Wan[3]* and Chaoyang Song[4,5]*

[1]Robotics and Autonomous Systems Thrust, System Hub, Hong Kong University of Science and Technology, Hong Kong, Hong Kong SAR, China, [2]Department of Mechanical and Energy Engineering, Southern University of Science and Technology, Shenzhen, China, [3]School of Design, Southern University of Science and Technology, Shenzhen, China, [4]Shenzhen Key Laboratory of Biomimetic Robotics and Intelligent Systems, Department of Mechanical and Energy Engineering, Southern University of Science and Technology, Shenzhen, China, [5]Guangdong Provincial Key Laboratory of Human-Augmentation and Rehabilitation Robotics in Universities, Southern University of Science and Technology, Shenzhen, China

Besides direct interaction, human hands are also skilled at using tools to manipulate objects for typical life and work tasks. This paper proposes DeepClaw 2.0 as a low-cost, open-sourced data collection platform for learning human manipulation. We use an RGB-D camera to visually track the motion and deformation of a pair of soft finger networks on a modified kitchen tong operated by human teachers. These fingers can be easily integrated with robotic grippers to bridge the structural mismatch between humans and robots during learning. The deformation of soft finger networks, which reveals tactile information in contact-rich manipulation, is captured passively. We collected a comprehensive sample dataset involving five human demonstrators in ten manipulation tasks with five trials per task. As a low-cost, open-sourced platform, we also developed an intuitive interface that converts the raw sensor data into state-action data for imitation learning problems. For learning-by-demonstration problems, we further demonstrated our dataset's potential by using real robotic hardware to collect joint actuation data or using a simulated environment when limited access to the hardware.

Keywords: robot learning, data collection, vis-tac sensing, soft robotics, imitation learning

Edited by:
Yasemin Bekiroglu,
Chalmers University of Technology,
Sweden

Reviewed by:
Gustavo Alfonso Garcia Ricardez,
Ritsumeikan University, Japan
Zhongtao Fu,
Wuhan Institute of Technology, China

****Correspondence:***
Fang Wan
wanf@sustech.edu.cn
Chaoyang Song
songcy@ieee.org

†These authors have contributed equally to this work and share first authorship

Specialty section:
This article was submitted to Computational Intelligence in Robotics, a section of the journal Frontiers in Robotics and AI

Received: *30 September 2021*
Accepted: *31 January 2022*
Published: *15 March 2022*

Citation:
Wang H, Liu X, Qiu N, Guo N, Wan F and Song C (2022) DeepClaw 2.0: A Data Collection Platform for Learning Human Manipulation. Front. Robot. AI 9:787291. doi: 10.3389/frobt.2022.787291

1 INTRODUCTION

Learning from human behaviors is of great interest in robotics (Argall et al., 2009; Kroemer et al., 2019; Osa et al., 2018). Dexterous operation of various tools plays a significant role in the evolution of human behaviors from ancient times (Kaplan et al., 2000) to modern civilization (Brown and Sammut, 2011). For imitation-based manipulation learning, it is common to collect behavior cloning data by directly observing the human hand (Christen et al., 2019) or through human-guided robot demonstration (Chu et al., 2016). However, it is also widely recognized that such dexterity in manipulation is also tightly related to the sense of touch through the fingers (Billard and Kragic, 2019), challenging to model and reproduce with current development in low-cost sensing solutions. For example, recent research in robotic Jenga player (Fazeli et al., 2019) and Gelsight sensors (Yuan et al., 2017) shows the potential in adopting active touch sensing for dexterous manipulation learning. However, a lack of low-cost, efficient, shareable, and reproducible access to the manipulation data remains a challenge ahead.

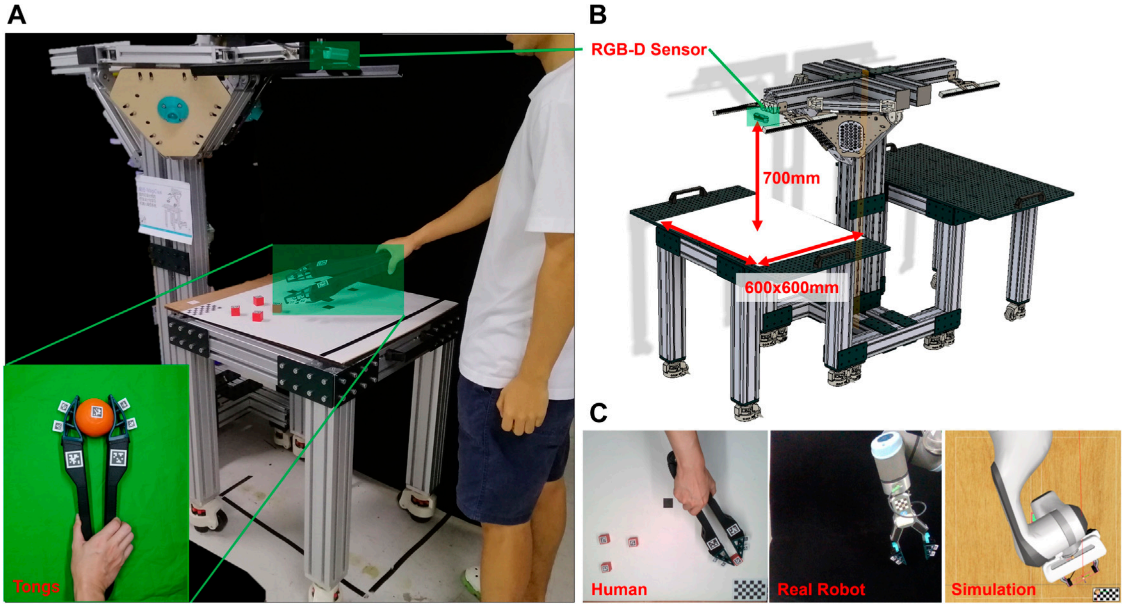


FIGURE 1 | Design overview of DeepClaw 2.0 for collection human manipulation data. As shown in **(C)**: manipulation data collected from a human teacher who operates a pair of modified kitchen tongs can be applied in imitation learning for robot in simulated or real environment. As shown in **(A)** an **(B)**: RGB-D sensor (Intel RealSense 435i is used in this paper) is fixed on the top, 700 mm away from the operating platform. Working space is a square with 600 × 600 mm. The origin of world coordinates is set in the left-top corner located by a checkerboard.

The correspondence issue with the arm and hand between humans and robots is another challenge yet to be resolved (Dautenhahn and Nehaniv, 2002). The human muscular-skeletal system provides a biological inspiration for modern robot manipulators and grippers, but most are rigidly designed with superhuman bandwidth for industrial purposes with a structural simplification for efficiency and accuracy (Gealy et al., 2019). When training with imitation learning algorithms, the undesirable motions from demonstrations and raw sensory inputs without proper embodiment are also tricky to resolve, given the structural mismatch in arm and hand between the robot learner and human teacher (Osa et al., 2018).

While it is common to learn from human demonstrations for robot learning, it remains difficult to track the dexterous multi-finger motion for exact behavior cloning using 2-finger parallel jaw grippers. Besides direct interaction with objects through the fingers, humans are also skilled at operating tools for dexterous manipulation. In this paper, we propose DeepClaw 2.0 (**Figure 1**) as a data collection platform for imitation learning, where a human teacher operates a pair of modified kitchen tongs to perform object manipulation. Using a single RGB-D camera (Intel RealSense D435i) as the only sensor, we can track the tongs' spatial motions with markers. We also attached a soft finger network on each tip of the tongs to infer unified physical interaction with objects using a vision-based force-sensing method. These soft fingers can also be fixed to parallel grippers, bridging the gap of structural mismatch between human fingers and robot grippers for learning. We collected comprehensive data of dexterous human manipulation with tools directly transferable to parallel grippers with the same set of fingers installed by tracking how humans operate the tongs and how the fingers deform. We also demonstrated how researchers could use this platform by feeding the collected data to commercial robot controllers to reproduce the manipulation, generating further data on the actuator angle, velocity, and torque/current incurred on each robot joint. For researchers with limited access to real robot hardware, we also demonstrated the possibility of motion reproduction in a simulated environment using CoppeliaSim (previously known as V-REP). We assumed the manipulation as a Markovian process and converted all data collected to fit a standard Markov Decision Process (MDP) model with an intuitive user interface. The proposed DeepClaw 2.0 platform is open-sourced with all configuration files and data hosted on Github[1] for shareable and reproducible research, aiming at a low-cost benchmarking solution among the robot learning community.

In the next section, we present the design details of DeepClaw 2.0, focusing on the data collection process. **Section 3** presents the experimental design, the collected dataset, and the reproduction in real and simulated robots. **Section 4** discusses some of the findings and potentials of the proposed platform. Conclusion, limitations, and future work are enclosed in the final section, which ends this paper.

[1] https://github.com/bionicdl-sustech/DeepClaw.

2 THE DATA COLLECTION DESIGN OF DEEPCLAW 2.0

2.1 Design Overview

The DeepClaw 2.0 aims at providing an open-sourced platform at a low cost to collect human data for learning robotic manipulation. As shown in **Figure 2**, the platform design leverages: 1) Continuous 6D pose estimation using low-cost cameras and simple markers for all state and action data of tongs, 2) the use of soft finger networks with omni-directional, passive adaptation for unified interaction by the humans and robots, and 3) object manipulation with human operators manually handling a pair of modified, 3D-printable kitchen tongs, which is transferable to parallel 2-finger robot grippers. The collected data involves the following stages of processing before becoming useful for imitation learning or learning-by-demonstration.

- **Raw Sensor Data**: There is only one sensor in DeepClaw 2.0, which is the low-cost Intel Realsense D435i with RGB-D capability. Therefore, the raw sensor data for each example are sequential images $I_N = \{i_0, i_1, \ldots, i_N\}$ of the task environment during the human demonstration, where i_t contains both RGB and depth image, recorded with timestamps $t \in \{0, 1, \ldots, N\}$.
- **Pose Estimation Data**: We attach markers on various places of the tongs, fingers, and objects and estimate their 6D poses $P_N = \{P_i^0, P_i^1, \ldots, P_i^N\}$ and $\Omega_N = \{\Omega_j^0, \Omega_j^1, \ldots, \Omega_j^N\}$ simultaneously by analyzing each recorded image, where $P_i^t, i = 1, \ldots, 6$ and $\Omega_j^t, j = 1, \ldots, M$ are poses of markers detected in corresponding image at timestamp t. We use P_i^t to represent pose of marker with ID i in the tongs and in the soft finger networks, and Ω_j^t to represent pose of marker with ID j in objects. The pose vector contains translation and rotation of the marker w. r.t the origin of the world coordinate system.
- **State-Action Data**: Based on the estimated pose data, we can extract information such as the 6D pose of tool center point of the tongs defined at the center of two tags on tongs with an offset to the horizontal plane of the tongs, the gripper's opening width, and the deformation of soft finger networks. Then, we convert the pose information to state data based on its physical representations. A selected few state data, especially those related to the soft finger networks' motion and deformation, will be used to extract action data by taking their time derivatives. A suggested way to utilize the pair of state-action data that formulating manipulation learning tasks as a task family described by a distribution of Markov Decision Processes (MDPs), $P(M)$, which follows the conventions used in (Kroemer et al., 2019). To be more specific, each manipulation M_i is fully defined by a tuple of $(S_r \times S_{e_i}, A, R_i, T_i, \gamma_i)$ where S_r is the state of the tongs, including the pose of the P_{tcp} and three distance variables $\|P_1^i - P_2^i\|, \|P_3^i - P_4^i\|, \|P_5^i - P_6^i\|$ describing the internal state of the tongs; S_{e_i} is the environment state composed of the tabletop setup S_{w_i} and the object states $\Omega_1^i \times \cdots, \times \Omega_j^i$; A is the action space of the tongs, which is inferred from the state of the tongs; R_i is the reward function depending on the goal of each task; T_i is time horizon i.e., the time steps in each episode; γ_i is the discount factor. The left tuple elements are not explicitly defined or computed.
- **Post-processed Data**: While one can already make use of the above state-action data for learning, we further integrated the capability for data reproduction and collection with real and simulated robots for joint and actuator data, which can be used for demonstration-related learning algorithm development (Argall et al., 2009).

2.2 Hardware Mechanic Design

As a shareable and reproducible standardized robot cell, the mechanical frame of DeepClaw2.0 uses aluminum extrusions easily obtained from local shops or global suppliers such as MISUMI Group Inc. In order to achieve ease of assembly, the DeepClaw2.0 platform uses aluminum plates that are drilled with a 2-by-4 hole array, and these aluminum plates are easily machined from local shops. Using bolts and connectors to assembly aluminum extrusions and drilled plates is convenient. Other components such as flanges and wheels are also needed and easy to assembly. All the components are listed in the **Figure 3** and **Table 1**, and more details are shown in **Supplementary Figure S1**. In order to provide a better light environment for data collection using cameras, we added two lighters fixed by plates. Also, we use some machined plates as the base of robot arms, which show the potential of DeepClaw2.0 as a data collection platform and a system that can realize the process from data collection to use. The top-left corner of **Figure 3** shows the detail of the robot arms, which using DeepClaw as a base to perform tasks such as implementing a trained learning model. The center and top-right of **Figure 3** show the ability of DeepClaw to collect data and use data simultaneously. The bottom-right of **Figure 3** shows the details of the gripper for data collection.

2.3 From Raw Sensor Data to Pose Estimation Data

The raw sensor data consist of color, and depth images with timestamps streamed at 30 fps from Intel Realsense D435i. Most researchers utilized some optical motion tracking system to extract human manipulation data (Kolahi et al., 2007). Commercialized high-precision products usually involve a multi-camera setup and complex system configuration, becoming expensive for shareable and reproducible research.

Pose estimation and tracking are the key factors affecting the performance of a motion capture system. We compared three commonly used marker detection algorithms using the same Realsense D435i camera. In the experiments, a 3D-printed L-shape board was mounted on the tool flange of UR10e. A $3\,cm \times 3\,cm$ ArUco tag, a 4 × 5 checkerboard with grid size 0.6 cm, and a $3\,cm \times 3\,cm$ AprilTag were attached to the L-shape board's center, respectively. UR10e repeated the same trajectory while we tracked the markers. We sampled ten fixed waypoints along the trajectory and calculated the detection success rate, the average computation time, and the average rotational error (Suzui

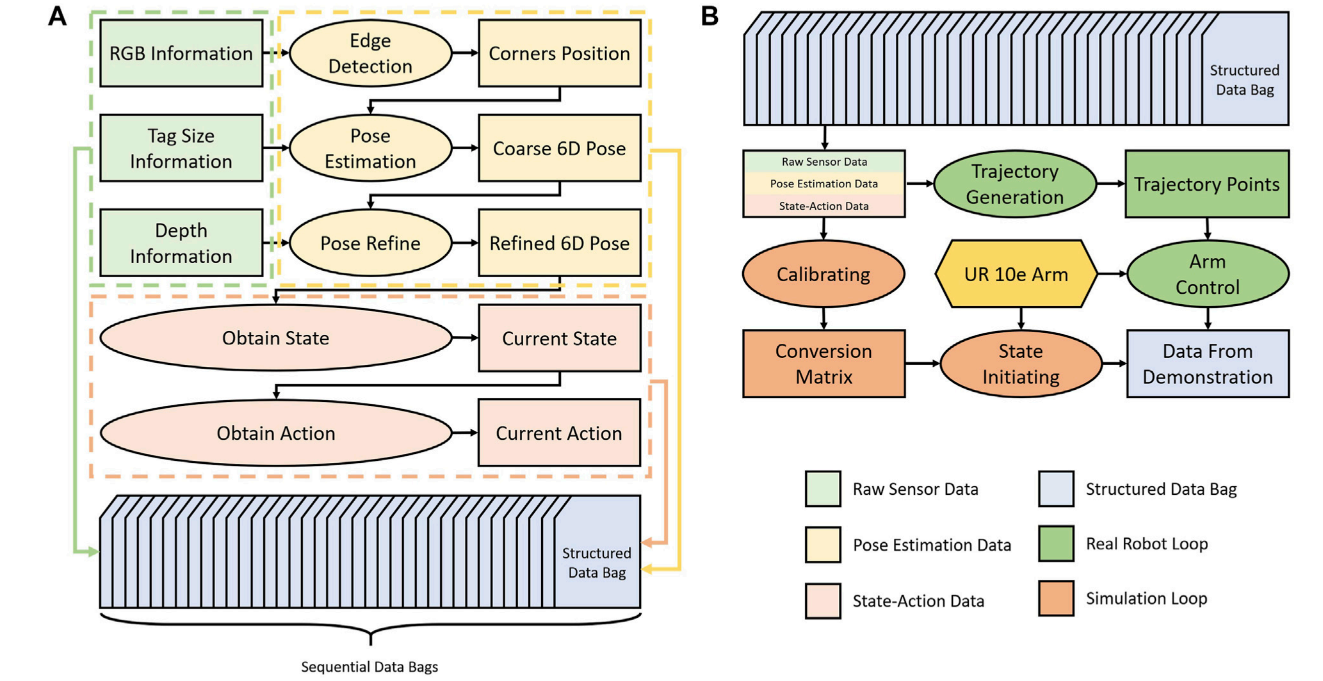


FIGURE 2 | The data collection pipeline of DeepClaw 2.0 is shown in **(A)**, which includes raw sensor data (green), pose estimation data (light yellow), state-action data (orange), and post-processed data (light gray). Raw data consists of three types of information: tag data (ID number and size) as the prior information, RGB image, and depth image captured by a single fixed camera (Intel RealSense D435i in this paper). Low-level features are collected as pose estimation data, include detected corner point and 6D pose of the marker. State-action information, constructed from the above low-level features, reveals the motion of both objects and tongs during the manipulation task. Two typical ways to use structured data bags are shown in **(B)**. The green branch indicates how the real robot reproduces the trajectory recovered from data bags, and the orange branch demonstrates steps to regain manipulation tasks.

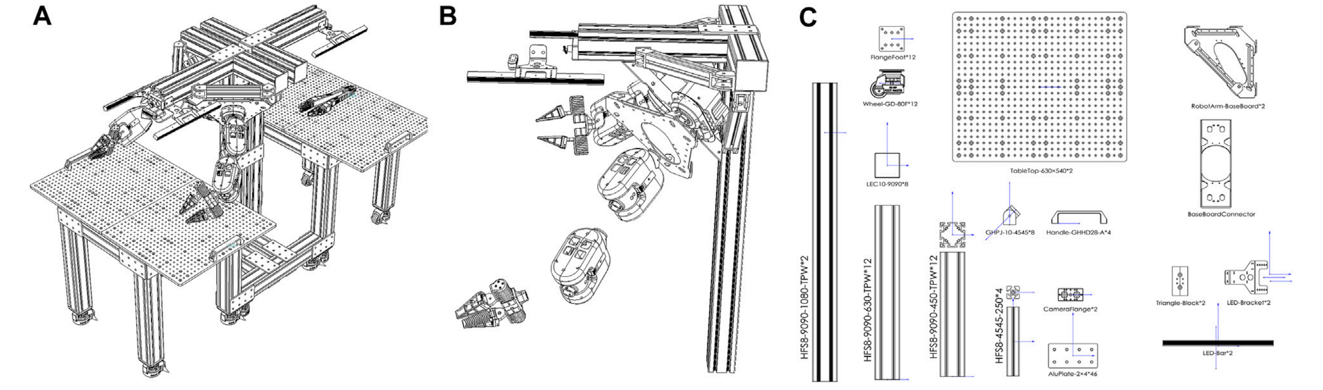


FIGURE 3 | DeepClaw2.0 Station: **(A)** 3D view, **(B)** robot arm explosion view, **(C)** details about components for data collection.

et al., 2019), as shown in **Table 2**. The trajectory was repeated ten times for each marker. Our results show that the AprilTag detector in ViSP is the most reliable method among the three, with a 100% detection success rate with the least rotational error at 0.36 radian. It also performed consistently in terms of efficiency with the least variations. The AprilTag detector in ViSP is the only method utilizing the extra depth information provided by the Realsense D435i, which might contribute to its superior performance among the three methods tested.

As a result, we adopted the AprilTag as the marker for our modified tongs design, as shown in **Figures 4B,C**. A set of the soft finger networks with omni-directional adaptation is also adopted, with further technical details explained in (Yang et al., 2020a,b; Wan et al., 2020). Both the tongs and soft finger networks are 3D printable. Three tags are fixed on each arm of the tongs, with one is near the end of the arm and two attached to the back of each soft finger network. All six tags can be uniquely identified and localized, producing a set of marker poses P_i^t where $i = 1, \ldots 6$ and t is the time sequence. The tags

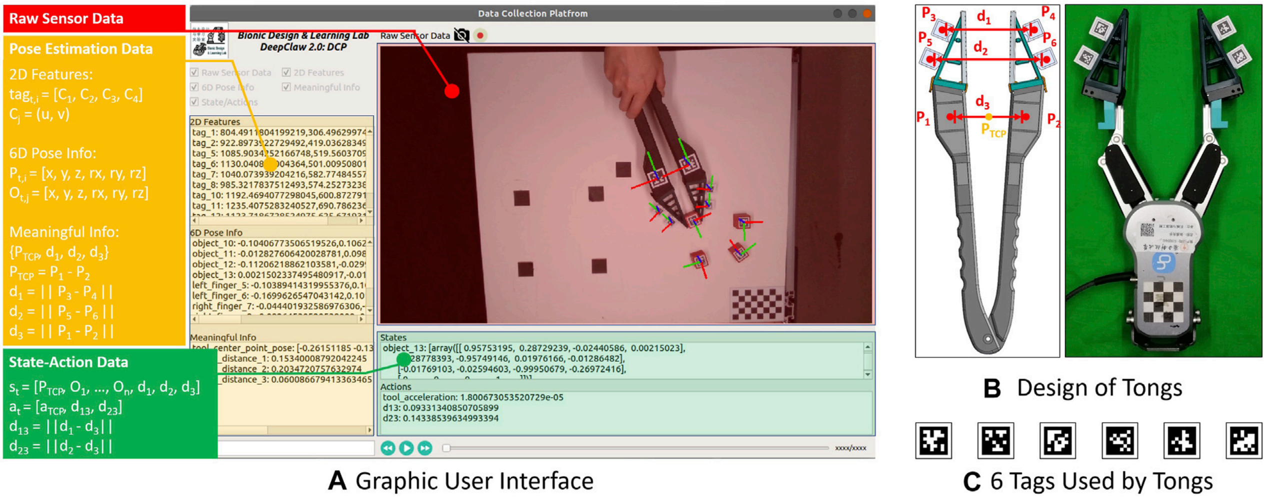
TABLE 1 | Bill of materials for DeepClaw 2.0.

Number	Type	Quantity	Number	Type	Quantity
1	AluExtru:HFS8-9090-1080-TPW	2	10	CameraFlange	2
2	AluExtru:HFS8-9090-630-TPW	12	11	Handle-GHHD28-A	4
3	AluExtru:HFS8-9090-450-TPW	12	12	TableTop-630 × 540	2
4	AluExtru:HFS8-4545-250-TPW	4	13	RobotArm-BaseBoard	2
5	AluExtruCover:LEC10-9090	8	14	BaseBoardConnector	1
6	Wheel-GD-80F	12	15	Triangle-Block	2
7	FlangeFoot	12	16	LED-Bracket	2
8	GHPJ10-4545	8	17	LED-Bar	2
9	AluPlate-2×4	46	—	—	—

TABLE 2 | Marker comparison for 6D pose estimation.

Method	Detection success rate (%)	Average computation time (s)	Average rotational error (rad)	Average position error (cm)
Aruco Minichino (2015)	32	**0.16 (0.17)**	1.60 (0.03)	0.292 (0.162)
Checkerboard Minichino (2015)	90	0.42 (0.12)	0.96 (0.19)	1.005 (0.628)
AprilTag in ViSP (Marchand et al. (2005)	**100**	0.20 (0.01)	**0.36 (0.12)**	**0.085 (0.057)**

Bold values represent the better performance.



```
2D Features
tag_1: 804.4911804199219,306.49629974
tag_2: 922.8973922729492,419.03628349
tag_5: 1085.9034252166748,519.5603709
tag_6: 1130.0408   004364,501.00950801
tag_7: 1040.0739392 04216,582.77484557
tag_8: 985.3217837512493,574.25273238
tag_10: 1192.4694077298045,600.872791
tag_11: 1235.4075283240527,690.786236

6D Pose Info
object_10: -0.1040677350651952 6,0.1062
object_11: -0.012827606420028781,0.098
object_12: -0.11206218862103581,-0.029
object_13: 0.0021502337495480917,-0.01
left_finger_5: -0.10389414319955376,0.1
left_finger_6: -0.1699626547043142,0.10
right_finger_7: -0.04440193258697630 6,-

Meaningful Info
tool_center_point_pose: [-0.26151185 -0.13
_distance_1: 0.15340008792042245
_distance_2: 0.2034720757632974
_distance_3: 0.06008667941336346 5

States
object_13: [array([[ 0.95753195,  0.28729239, -0.02440586,  0.00215023],
        28778393, -0.95749146,  0.01976166, -0.01286482],
       [-0.01769103, -0.02594603, -0.99950679, -0.26972416],

Actions
tool_acceleration: 1.800673053520729e-05
d13: 0.09331340850705899
d23: 0.14338539634993394
```

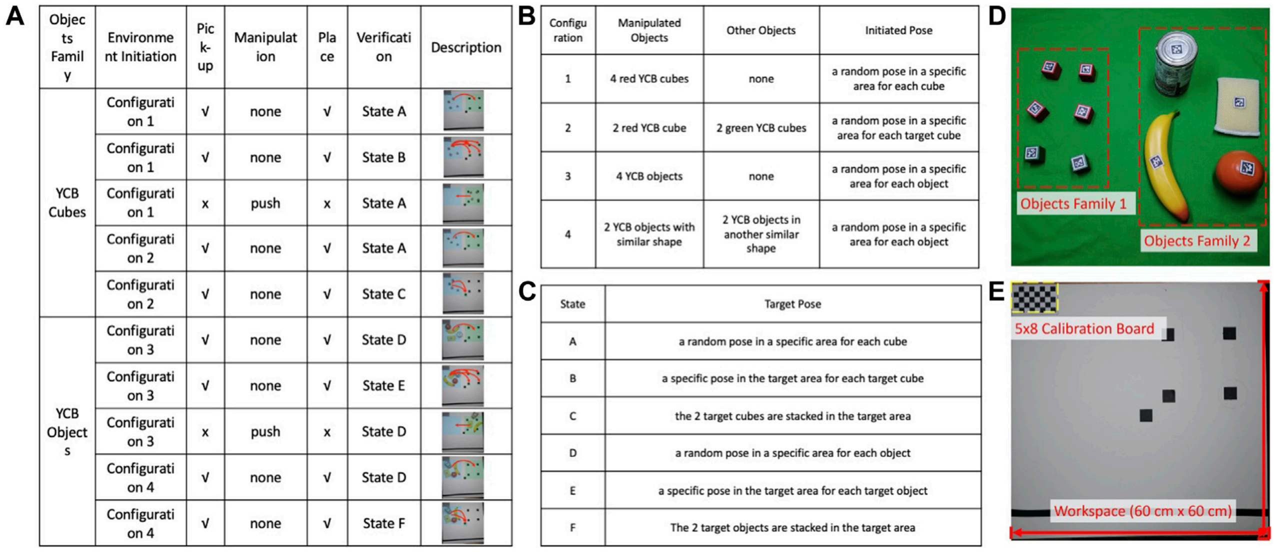
FIGURE 4 | Key components in human manipulation: **(A)** Shows the rendered graphic user interface in DeepClaw 2.0, consists of real-time RGB data flow (highlighted by the red rectangle), low-level features (highlighted by the yellow rectangle), and high-level state-action information (highlighted by the green rectangle). **(B)** Shows the similarities between the assembled tongs and an OnRobot RG6 gripper, some key parameters are pointed, and in **(C)**, six tags from the AprilTag 36h11 family used in this paper are shown.

were also attached to the objects so that we can keep track of the object poses O_j^t where j iterates over the object ID.

We also developed an intuitive graphical interface, as shown in **Figure 4A**, which can be used to visualize the real-time data streaming during the data collection as well as to examine the collected data.

2.4 Deformation Estimation *via* Passive Perception

It is essential and meaningful to estimate the contact behavior during manipulation. For example, in the task of grasping soft objects or objects of unknown weight, data generated by the interaction between humans and the environment during the teaching process can provide the robot with more abundant information, such as the closing distance of gripper or force adjustment strategy (Yuan et al., 2015). The tactile sensor can provide interactive information, which is ignored by the camera, for robot operations. However, the formation of data provided by the different tactile sensors is various and highly related to its mechanical design. It is not easy to generalize tactile sensing across different types of sensors. To fill this gap, DeepClaw 2.0 uses a low-cost RGB-D camera and provides a visual detection algorithm to estimate the contact deformation of the soft finger



Objects Family	Environment Initiation	Pick-up	Manipulation	Place	Verification	Description
YCB Cubes	Configuration 1	√	none	√	State A	
	Configuration 1	√	none	√	State B	
	Configuration 1	x	push	x	State A	
	Configuration 2	√	none	√	State A	
	Configuration 2	√	none	√	State C	
YCB Objects	Configuration 3	√	none	√	State D	
	Configuration 3	√	none	√	State E	
	Configuration 3	x	push	x	State D	
	Configuration 4	√	none	√	State D	
	Configuration 4	√	none	√	State F	

Configuration	Manipulated Objects	Other Objects	Initiated Pose
1	4 red YCB cubes	none	a random pose in a specific area for each cube
2	2 red YCB cube	2 green YCB cubes	a random pose in a specific area for each target cube
3	4 YCB objects	none	a random pose in a specific area for each object
4	2 YCB objects with similar shape	2 YCB objects in another similar shape	a random pose in a specific area for each object

State	Target Pose
A	a random pose in a specific area for each cube
B	a specific pose in the target area for each target cube
C	the 2 target cubes are stacked in the target area
D	a random pose in a specific area for each object
E	a specific pose in the target area for each target object
F	The 2 target objects are stacked in the target area

FIGURE 5 | We defined 10 manipulation tasks shown in **(A)**. This paper uses two objects families: four red and two green wooden cubes as one set and four common objects from YCB as another set. We stress two states and three operations in one manipulation task: initial state, target state, picking, pushing, and placing. The details of different initial states and target states are listed in **(B,C)**. The two families of objects from YCB are shown in **(D)**: A set of six cubes; a set of the orange, banana, sponge, and can. The setup of tabletop is shown in **(E)**.

network passively. By capture the motion of markers that are fixed at the finger, the contact behavior is recognized. We conducted experiments on ten typical manipulation tasks and compared the deformation of soft finger networks when interacting with objects of different shapes, such as cube and orange, and objects made of soft or solid materials, such as sponge and can.

2.5 Optional Collection of Post-processed Data

To verify the reproducibility of the structured data collected by DeepClaw 2.0, we transfer the collected data into the controller of a physical robot arm and a simulation environment. This post-processing step enables us to obtain robot state data about the robot joint/actuator. Based on the action information in collected data, we feed the smoothed trajectories in Cartesian space generated from human manipulators to the physical robot arm. The robot arm obtains the inverse kinematic solution and executes the motion planning by shadowing the human manipulations. The exact reproduction process can be done with simulated robot arms. Thus, the data can be used for robot learning-by-demonstration of specific tasks in the real world and in simulation.

3 EXPERIMENTS AND RESULTS

3.1 Experiment Setup and Procedure

We divide a manipulation task into five phases: The initial state, picking, manipulation, placing, and target state, where manipulation refers to robot actions other than picking and placing. By exploring the setup of objects and whether to execute picking, placing, or other manipulations, we designed a family of ten tasks, as summarized in **Figure 5** and **Supplementary Table S1**. From the Yale-CMU-Berkeley Object and Model Set (Calli et al., 2015), we selected the small wooden cubes of different colors and a few sample objects to represent two levels of object complexity. Each object is attached with a unique AprilTag for pose estimation. For each object set, the five tasks in **Figure 5A** from left to right differ in manipulation complexity. The origin of the world coordinate is at the bottom-right corner of the checkerboard with the *z*-axis facing upward. All the collected pose data are transformed relative to the world coordinate.

We invited five operators, asking each to repeat each of the ten tasks in five trials. As a result, we collected a dataset of 250 task executions in total. Each task execution data consists of a sequence of color and depth images, one pack of a structured data bag, and one video of the manipulation process recorded by another camera. The researcher starts the experiment by launching the data visualization interface. The operator is asked to stand in front of the tabletop holding the tongs in the right hand. After receiving the starting signal from the researcher, the operator starts executing the manipulation task. Once completed, the operator notifies the researcher to stop the recording in the user interface.

3.2 Experiment Results

We performed a mean filter to reduce the disturbance during the data collecting process. A sample statistical overview of the data is visualized in **Figure 6**. The statistical analysis provides a glimpse of the characteristics in each manipulation task, where different phases of a single task can be clearly identified. The trajectory

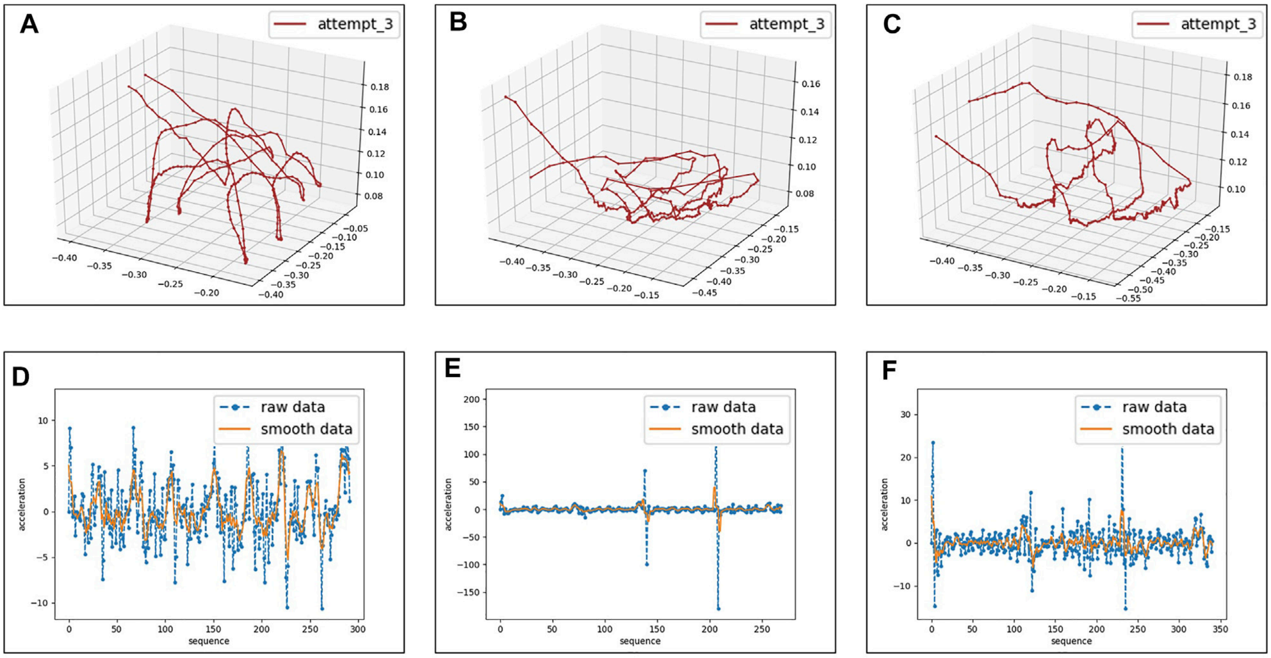


FIGURE 6 | Experiment results and analysis of the collected data. We plot the trajectory of the third attempt of task 2, task 3, and task 8 from a human operator in **(A–C)**. Operations (pushing, picking, or placing) and (initial or target) state can be easily distinguished by observing the motion trajectory. Sub-figures from **(D)** to **(F)** at the bottom line represent the corresponding acceleration sequence with and without smoothing. Motion-related data, such as position, velocity, and acceleration, provides a quantifiable indicator of operations.

visualization provides support for the reproduction experiment in real and simulated robots.

3.3 Experiment With Real and Simulated Robots

We reproduced the trajectory of a human operator executing task 1. Then, do the reproduction with UR10e. The robot states, including joint angle, joint current, and joint velocity, were recorded during the process. These experiments show that the structured data collected from different manipulation tasks can be reproduced by the robot arm in the same data process.

We also reproduced the same task in simulation using CoppeliaSim (Freese et al., 2010). The object's pose information was utilized to render an initial state of the objects cloning the real scene. The simulated Franka Emika arm with its 2-finger gripper then tried to follow the trajectory of a human operator. As shown in **Figure 1**, the simulated robot can reproduce task 1 recorded with a human operator. Please refer to the Github page for further details on the motion reproduction.

4 DISCUSSION

4.1 Dexterous Object Manipulation Through Operating Tools

Tracking human hand motion can be challenging and expensive (Billard and Kragic, 2019). Recent research already demonstrated success in tracking refined and real-time hand motion for training advanced manipulation skills using physical (OpenAI et al., 2019) or simulated robots (Han et al., 2018). Besides a multi-camera motion capture system, data gloves are another alternative (Sundaram et al., 2019; Dillmann, 2004), but may suffer from interfering with the natural motion and touch feeling of the human hand. Even when such data becomes cost-effectively available, which is yet to be the case, there remains another challenge in the availability of robotic hands matching the human's capability at a lower cost. While many multi-fingered robot hands (OpenAI et al., 2019; Zhou et al., 2019) are available, there remains a gap between technological maturity and cost-effectiveness.

This paper proposes a different perspective by collecting large-scale object manipulation data when human teachers are operating tools. Specifically, we adopt tongs as the tool of interest, commonly found in life and workplaces for food preparation or material handling. A convenient feature of the tongs is its two-fingered design, which is structurally similar to robotic grippers with parallel fingers. Although most tongs adopt a pivot or scissor mechanism, they usually come with "long arms terminating in small flat circular ends of tongs[2]," resulting in a convenient approximation to a parallel motion at a low cost. Our results demonstrate the potential to collect dexterous manipulation data from such configuration, which can be translated for imitation learning at a low cost in data collection.

[2] https://en.wikipedia.org/wiki/Tongs.

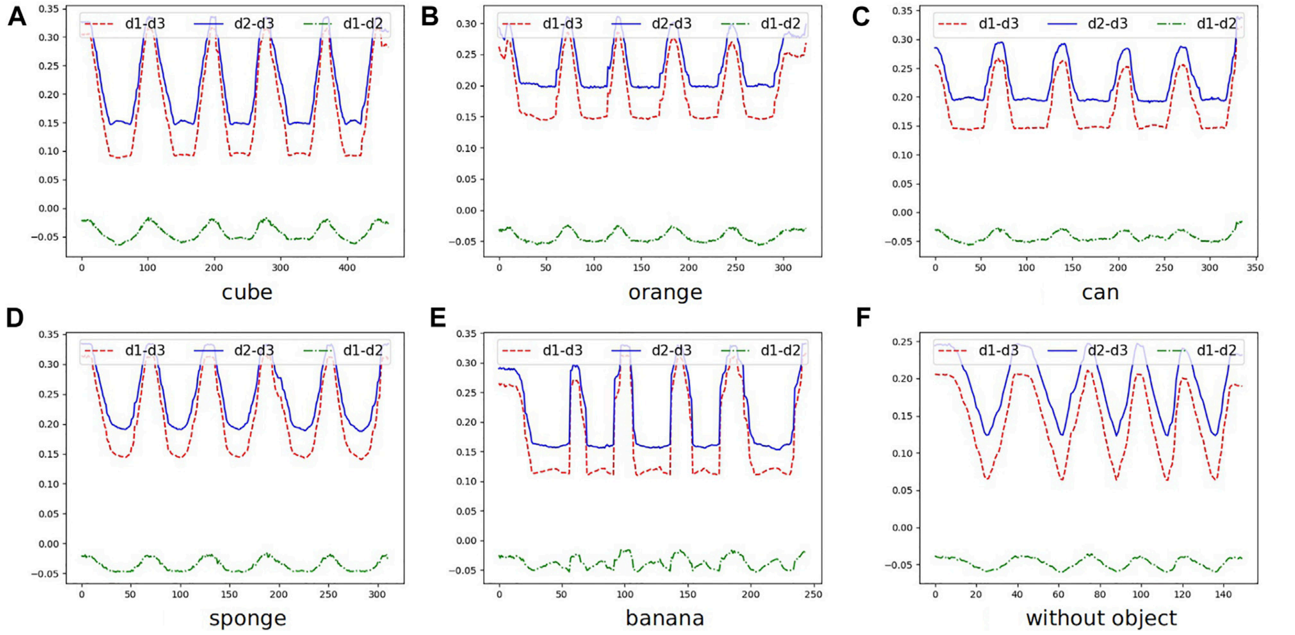


FIGURE 7 | The contact information of grasping five objects(**(A)**: cube, **(B)**: orange, **(C)**: can, **(D)**: sponge, and **(E)**: banana) and no object **(F)**) five times. To emphasis the differences in interaction, we record subtraction of every two values of distance. d1 and d2 reveal the deformation of soft finger networks. d3 is linear and proportional to the closing distance of tongs. Therefore, the subtraction of d1 and d3, or d2 and d3, may provide information of geometry (lower value when grasped means smaller size of the object) and material (curve or straight line at valley reveals soft or solid material). The precise prediction will be exploited in the future.

4.2 Vision-Based Force Sensing Using Soft Finger Network

We introduce a pair of soft finger networks to generate pseudo-force data about the physical interaction between the tongs and objects to keep the system simple, cost-effective, and versatile in data collection. Such a soft finger network provides an overconstrained grasping solution through compliant interaction at a low-cost (Yang L. et al., 2020; Wan et al., 2020), which can be leveraged through optical methods for non-contact force-sensing after calibration. Recent research demonstrated the possibility of embedding optical fibers inside such a simple finger network for tactile sensing in grasping (Yang Z. et al., 2020).

In this paper, we use markers instead of unifying the raw sensory data in image format and tracking the markers to collect the motion and deformation data of the soft finger when interacting with the objects, as shown in **Figure 7**. When both the robot learner and the human teachers use the same soft finger networks, we can directly use the deformation of the fingers as the pseudo-force data of physical interaction. If necessary, we can calibrate the soft finger network's stiffness to calculate the specific interaction force based on the measured deformation (Liu, 2020; Yang, 2020). Since these soft finger networks can be conveniently installed on the arms of the tongs and the robotic grippers, we can maintain a consistent interface of physical interaction. Such a vision-based sensory mechanism holds the potential to reduce the hardware cost and system complexity significantly, yet maintaining a consistent level of data robustness and conformity for training and deployment.

4.3 Feasibility of the Collected State-Action Data

The primary purpose of the proposed DeepClaw 2.0 platform is to facilitate the large-scale data collection for training models for manipulation learning algorithms, which is usually fitted to an MDP model. The DeepClaw platform facilitates a streamlined pipeline that encodes the sequential Raw Sensory Data in time-series image formats into State-Action Data with physical meanings for manipulation learning. Researchers can directly use State-Action Data by introducing specific transition functions, reward functions, and discount factors to develop algorithms based on MDP models for imitation learning.

For researchers interested in learning-by-demonstrations, if a physical robot is available, we also demonstrated the feasibility of collecting actuator data in motor current and joint data in angular position and velocity by feeding the collected State-Action Data to the controllers using a UR10e robot. Even if robot hardware is unavailable, we also demonstrated the feasibility of using a simulated Franka Emika robot to achieve a similar purpose, which may suffer from a reduced richness in data variety, but an enhanced benefit in reproducing the task with variant objects.

5 CONCLUSION

In this paper, we proposed the DeepClaw 2.0 platform with a low-cost RGB-D camera to collect the training data of object manipulation by tracking the spatial motion and deformation of a pair of specially designed soft finger networks with omni-directional adaptation. These fingers are used as a universal agent of physical interaction for dexterous objects manipulation, either by humans when operating a pair of tongs fixed with these fingers or by a robot with a common parallel jaw-gripper with such fingers installed. We can collect the pseudo-force interaction data between the fingers and objects using vision-based force-sensing through these soft finger networks. We presented an intuitive interface to manage the State-Action Data collection process for training imitation learning algorithms. We demonstrated a viable solution to collect further robot state and action data, which can be alternatively used for research on learning-by-demonstration.

This study is limited to a performance benchmark of the collected data for training manipulation learning, which will be studied in future work. As the RGB-D camera is the only sensor used, the quality of the collected data relies heavily on the performance of the camera used (Intel RealSense D435i). If higher resolution is necessary, systems such as Photoneo MotionX (Wan and Song, 2020) could be a potential alternative for high-quality point clouds with gray-scaled images, but at a much higher price. The vision-based force-sensing capability can be further improved for sensitivity and accuracy. We are currently working on an alternative design by embedding cameras inside the soft finger for more sensitive sensing, which will be addressed in another paper.

The future work of this study mainly aims at a more comprehensive system design towards a low-cost, open-sourced platform for robot manipulation learning research. As shown in **Figure 1** right, the full design of DeepClaw 2.0 also involves a second robot station with a lobster-inspired robot with soft fingers on a parallel gripper. Recent research by the Berkley Open Arm project (Gealy et al., 2019) proposed a promising design paradigm in this direction for robot hardware that “enables useful automation in unconstrained real-world human environments at low cost”.

DATA AVAILABILITY STATEMENT

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found below: https://github.com/bionicdl-sustech/DeepClaw.

AUTHOR CONTRIBUTIONS

HW initiated the research, prepared the original draft, developed the code, and analyzed the data. XL proofread the manuscript and prepared the code. NQ conducted the experiment, collected the data, assembled the workstation. NG worked on the robot driver development. FW initiated the original DeepClaw code and experiment, proofread the manuscript. CS supervised the project, initiated the original DeepClaw code, designed the experiment, proofread the manuscript.

FUNDING

This work was supported by the Science, Technology, and Innovation Commission of Shenzhen Municipality (Grant: ZDSYS20200811143601004, 20200925155748006), Centers for Mechanical Engineering Research and Education at MIT and SUSTech (MechERE Centers at MIT and SUSTech), and National Science Foundation of China (Grant: 51905252).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/frobt.2022.787291/full#supplementary-material

REFERENCES

Argall, B. D., Chernova, S., Veloso, M., and Browning, B. (2009). A Survey of Robot Learning from Demonstration. *Robotics Autonomous Syst.* 57, 469–483. doi:10.1016/j.robot.2008.10.024

Billard, A., and Kragic, D. (2019). Trends and Challenges in Robot Manipulation. *Science* 364. doi:10.1126/science.aat8414

Brown, S., and Sammut, C. (2011). *Tool Use Learning in Robots*.

Calli, B., Walsman, A., Singh, A., Srinivasa, S., Abbeel, P., and Dollar, A. M. (2015). Benchmarking in Manipulation Research: Using the yale-cmu-berkeley Object and Model Set. *IEEE Robot. Automat. Mag.* 22, 36–52. doi:10.1109/mra.2015.2448951

Christen, S., Stevsic, S., and Hilliges, O. (2019). Demonstration-guided Deep Reinforcement Learning of Control Policies for Dexterous Human-Robot Interaction. arXiv: Robotics. doi:10.1109/icra.2019.8794065

Chu, V., Akgun, B., and Thomaz, A. L. (2016). *Learning Haptic Affordances from Demonstration and Human-Guided Exploration*, 119–125.

Dautenhahn, K., and Nehaniv, C. L. (2002). *Imitation in Animals and Artifacts*.

Dillmann, R. (2004). Teaching and Learning of Robot Tasks via Observation of Human Performance. *Robotics Autonomous Syst.* 47, 109–116. doi:10.1016/j.robot.2004.03.005

Fazeli, N., Oller, M., Wu, J., Wu, Z., Tenenbaum, J. B., and Rodriguez, A. (2019). See, Feel, Act: Hierarchical Learning for Complex Manipulation Skills with Multisensory Fusion 4.

Freese, M., Singh, S., Ozaki, F., and Matsuhira, N. (2010). “Virtual Robot Experimentation Platform V-Rep: A Versatile 3d Robot Simulator,” in *Simulation, Modeling, and Programming for Autonomous Robots*. Editors N. Ando, S. Balakirsky, T. Hemker, M. Reggiani, and O. von Stryk (Berlin, Heidelberg: Springer Berlin Heidelberg), 51–62. doi:10.1007/978-3-642-17319-6_8

Gealy, D. V., McKinley, S., Yi, B., Wu, P., Downey, P. R., Balke, G., et al. (2019). Quasi-direct Drive for Low-Cost Compliant Robotic Manipulation. *Int. Conf. Robotics Automation (Icra)* 2019. doi:10.1109/icra.2019.8794236

Han, S., Liu, B., Wang, R., Ye, Y., Twigg, C. D., and Kin, K. (2018). Online Optical Marker-Based Hand Tracking with Deep Labels. *ACM Trans. Graphics* 37. doi:10.1145/3197517.3201399

Kaplan, H., Hill, K., Lancaster, J., and Hurtado, A. M. (2000). A Theory of Human Life History Evolution: Diet, Intelligence, and Longevity. *Evol. Anthropol.* 9, 156–185. doi:10.1002/1520-6505(2000)9:4<156:aid-evan5>3.0.co;2-7

Kolahi, A., Hoviattalab, M., Rezaeian, T., Alizadeh, M., Bostan, M., and Mokhtarzadeh, H. (2007). Design of a Marker-Based Human Motion Tracking System. *Biomed. Signal Process. Control.* 2, 59–67. doi:10.1016/j.bspc.2007.02.001

Kroemer, O., Niekum, S., and Konidaris, G. (2019). *A Review of Robot Learning for Manipulation: Challenges, Representations, and Algorithms*.

Liu, S. (2020). *A Research on the Mechanism of Vision Based Tactile Sensing for Adaptive Flexible GraspingMaster’s Thesis*. Harbin Institute of Technology.

Marchand, E., Spindler, F., and Chaumette, F. (2005). Visp for Visual Servoing: a Generic Software Platform with a Wide Class of Robot Control Skills. *IEEE Robot. Automat. Mag.* 12, 40–52. doi:10.1109/mra.2005.1577023

Minichino, J. (2015). *Learning OpenCV 3 Computer Vision With Python: Unleash The Power Of Computer Vision With Python Using OpenCV. Community Experience Distilled*. Packt Publishing.

OpenAIAkkaya, I., Andrychowicz, M., Chociej, M., Litwin, M., McGrew, B., et al. (2019). *Solving Rubik's Cube with a Robot Hand*, 1–51.

Osa, T., Pajarinen, J., Neumann, G., Bagnell, J. A., Abbeel, P., and Peters, J. (2018). *An Algorithmic Perspective on Imitation Learning*.

Sundaram, S., Kellnhofer, P., Li, Y., Zhu, J.-Y., Torralba, A., and Matusik, W. (2019). Learning the Signatures of the Human Grasp Using a Scalable Tactile Glove. *Nature* 569, 698–702. doi:10.1038/s41586-019-1234-z

Suzui, K., Yoshiyasu, Y., Gabas, A., Kanehiro, F., and Yoshida, E. (2019). "Toward 6 Dof Object Pose Estimation with Minimum Dataset," in IEEE/SICE International Symposium on System Integration (SII). IEEE, 462–467. doi:10.1109/sii.2019.8700331

Wan, F., and Song, C. (2020). Flange-based Hand-Eye Calibration Using a 3d Camera with High Resolution, Accuracy, and Frame Rate. *Front. Robot. AI* 7, 65. doi:10.3389/frobt.2020.00065

Wan, F., Wang, H., Wu, J., Liu, Y., Ge, S., and Song, C. (2020). A Reconfigurable Design for Omni-Adaptive Grasp Learning. *IEEE Robotics Automation Lett.* 5, 4210–4217. doi:10.1109/lra.2020.2982059

Yang, H. (2020). *Reconfigurable Design and Integration of Linear Collaborative Robot ModuleMaster's Thesis*. Harbin Institute of Technology.

Yang, L., Wan, F., Wang, H., Liu, X., Liu, Y., and Pan, J. (2020a). Rigid-soft Interactive Learning for Robust Grasping. *IEEE Robotics Automation Lett.* 5, 1720–1727. doi:10.1109/lra.2020.2969932

Yang, Z., Ge, S., Wan, F., Liu, Y., and Song, C. (2020b). "Scalable Tactile Sensing for an Omni-Adaptive Soft Robot finger," in 2020 3rd IEEE International Conference on Soft Robotics (RoboSoft), 572–577. doi:10.1109/robosoft48309.2020.9116026

Yuan, W., Dong, S., and Adelson, E. H. (2017). Gelsight: High-Resolution Robot Tactile Sensors for Estimating Geometry and Force. *Sensors* 17, 2762. doi:10.3390/s17122762

Yuan, W., Li, R., Srinivasan, M. A., and Adelson, E. H. (2015). "Measurement of Shear and Slip with a Gelsight Tactile Sensor," in 2015 IEEE International Conference on Robotics and Automation (ICRA) (IEEE), 304–311. doi:10.1109/icra.2015.7139016

Zhou, J., Chen, X., Chang, U., Lu, J., Leung, C. C. Y., Chen, Y., et al. (2019). A Soft-Robotic Approach to Anthropomorphic Robotic Hand Dexterity. *IEEE Access* 7, 101483–101495. doi:10.1109/access.2019.2929690

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.