VinT-6D: A Large-Scale Object-in-hand Dataset from Vision, Touch and Proprioception

Zhaoliang Wan [1 2]   Yonggen Ling [2]   Senlin Yi [1]   Lu Qi [3]   Wangwei Lee [2]   Minglei Lu [2]   Sicheng Yang [2]
Xiao Teng [2]   Peng Lu [2]   Xu Yang [4]   Ming-Hsuan Yang [3]   Hui Cheng [1]

Abstract

This paper addresses the scarcity of large-scale datasets for accurate object-in-hand pose estimation, which is crucial for robotic in-hand manipulation within the "Perception-Planning-Control" paradigm. Specifically, we introduce VinT-6D, the first extensive multi-modal dataset integrating vision, touch, and proprioception, to enhance robotic manipulation. VinT-6D comprises 2 million VinT-Sim and 0.1 million VinT-Real splits, collected via simulations in MuJoCo and Blender and a custom-designed real-world platform. This dataset is tailored for robotic hands, offering models with whole-hand tactile perception and high-quality, well-aligned data. To the best of our knowledge, the VinT-Real is the largest considering the collection difficulties in the real-world environment so that it can bridge the gap of simulation to real compared to the previous works. Built upon VinT-6D, we present a benchmark method that shows significant improvements in performance by fusing multi-modal information. The project is available at https://VinT-6D.github.io/.

1. Introduction

Estimating 6D object-in-hand pose (Wang et al., 2019; Dikhale et al., 2022) is a crucial area in both computer vision and robotics, that benefits numerous applications such as dexterous manipulation (Kelestemur et al., 2022; Qi et al.,
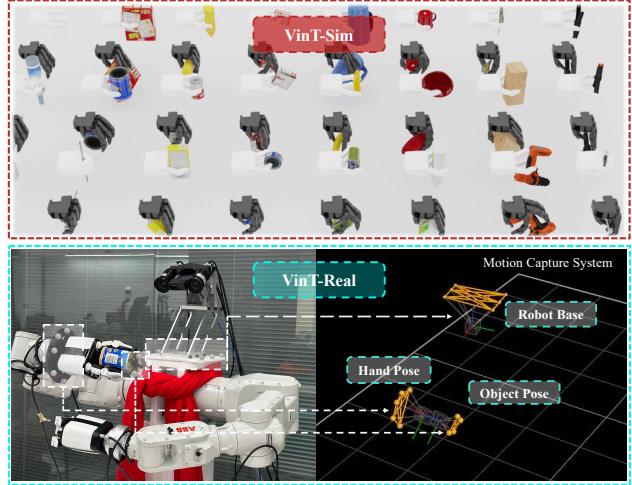

Figure 1. **Large-scale object-in-hand dataset VinT-6D comprising synthesized and real-world splits naming VinT-Sim and VinT-Real.** VinT-Sim aims to generate realistic data across vision, touch, and proprioception. VinT-Real is gathered through a precisely calibrated and aligned multi-modal robot platform, where a motion capture system obtains accurate object and hand poses.

2023b; Suresh et al., 2023) and grasping (Liang et al., 2020). Specifically, utilizing multi-modal signals, including vision, touch, and proprioception, is emerging as a significant research trend due to their unique characteristics (Liang et al., 2020; Cui et al., 2021; Röstel et al., 2022; Kelestemur et al., 2022; Lin et al., 2023b). These approaches align closely with human biological mechanisms, where the coordination of eyes, muscles, and joints is central to manipulating objects held in the hands.

Despite significant progress in representation learning across three modalities, this field faces two major issues due to the existing low-quality real-world dataset. First, due to the scarcity of large-scale public data, one common practice is to resort to synthesized data for training, resulting in a substantial domain gap with real-world environments at test time. Second, as existing methods focus on synthesized data for two-finger grippers, they could not be applied to complex scenarios such as occlusions in three or four-finger grasping. Both aspects hinder the further deployment of 6D

Table 1. **Datasets for Object-in-Hand Pose Estimation.** Such datasets are rare and often constrained by size, cost, hardware configuration (including hand types and tactile sensor varieties), and ground truth quality. The symbol ○ indicates datasets not to be publicly released or slated for future release. ✓ and - indicate whether has some modality or location of tactile sensors.

Dataset	Num of Fingers	Available Modalities			Tactile Distribution			Data Scale	
		Vision	Touch	Prop.	Tip	Pulp	Palm	Sim	Real
Hand-Object (Wen et al., 2020)	2	✓	-	✓	-	-	-	12 K	1 K
VITA (Dikhale et al., 2022)	4	○	○	○	✓	✓	-	0.2 M	-
Fast-Grasp'D (Turpin et al., 2023)	3, 4, 5	○	-	-	-	-	-	1 M	-
TEG-Track (Liu et al., 2023)	2	○	○	○	✓	-	-	○	○
PoseFusion (Tu et al., 2023)	5	✓	✓	✓	✓	-	-	-	35 K
FeelSight (Suresh et al., 2023)	4	○	○	○	✓	-	-	-	○
VinT-6D (Ours)	3, 4	✓	✓	✓	✓	✓	✓	2 M	0.1 M

object-in-hand pose estimation in robotic applications.

Collecting a high-quality dataset for 6D object-in-hand pose estimation is challenging. The usage of various tactile sensors and robotic hand configurations makes it hard to generalize findings across different robotic hands. Gathering real-world data is also tricky due to time constraints, sensor malfunctions, imprecise data, and the difficulty of aligning multiple modalities. To achieve high-quality data collection, it is essential to establish a robust integration of hardware and software that can address these challenges and ensure data integrity.

We address the above-mentioned problems and build the large-scale multi-finger object-in-hand perception dataset VinT-6D that comprises synthesized and real-world splits naming VinT-Sim and VinT-Real (Figure 1). To the best of our knowledge, our dataset performs better than existing datasets (Dikhale et al., 2022; Turpin et al., 2023) in the aspect of both numbers and quality (Table 1). Specifically, VinT-Sim focuses on bridging the sim2real gap in visual, tactile, and proprioceptive sensing by developing a consistent representation of tactile readings and accurately simulating taxel distributions, as detailed in the touch signal modeling section (Section 3.1.1). The process involves loading the robotic hand and object into MuJoCo (Todorov et al., 2012) for iterative stable grasping simulations (Section 3.1.2), followed by rendering in Blender (Community, 2018) for multi-view, realistic vision data (Section 3.1.3);

VinT-Real captures multi-modal data through our custom-developed robotic platform, featuring an integration of visual, tactile, proprioceptive sensors and a motion capture system. We meticulously customize marker assemblies on fixed parts of each object, ensuring sub-millimeter level accuracy in object pose acquisition without hindering the object's functionality (Section 3.2.1). The platform's calibrated sensors precisely align visual and tactile data, allowing the robot to emulate toddler-like exploratory movements in various grasp poses, including shifts and rotations within the workspace.

Based on the VinT-6D dataset, we propose a fundamental baseline VinT-Net for accurate object-in-hand pose estimation with synergizing vision, touch, and proprioception signals. Our key design is the integration of touch and proprioception with vision, synergistically merging diverse and complementary sensory inputs. This design could benefit for object-in-hand pose estimation, particularly when vision is obscured by the robot's hand occlusions. Embedded touch sensors in fingers and palms and proprioceptive data from hands and arms offer valuable supplementary information. Extensive experiments show the effectiveness of our method compared with the other works.

2. Related Work

2.1. Object Pose Estimation from Vision

In computer vision, generic object pose estimation datasets such as LineMod (Hinterstoisser et al., 2011) and YCB-Video (Calli et al., 2017) provide valuable data, including color and depth images, segmentation labels, and object poses, facilitating significant advances in object pose estimation (Wang et al., 2019; Peng et al., 2019; Li et al., 2019; He et al., 2020; Chen et al., 2020; He et al., 2021; Wang et al., 2021; Chen et al., 2022; Lipson et al., 2022). However, existing datasets are less suitable for robotics applications demanding higher precision. Recently, a new dataset for robotic manipulation with millimeter-level pose accuracy was introduced (Tyree et al., 2022), but it focuses on objects on tables, not addressing in-hand scenarios. The study by (Wen et al., 2020) is notable for using hand state estimation to improve pose accuracy and for providing a dataset for a two-fingered gripper. However, in-hand perception is challenging for multi-fingered hands due to vision occlusions. This highlights the need for a dataset to benchmark vision-based multi-fingered object-in-hand poses.

2.2. Object Pose Estimation from Touch

Tactile signals are crucial for object-in-hand pose estimation during manipulation, as shown in studies like (Liang et al.,

2020; Röstel et al., 2022; Lin et al., 2023a; Kelestemur et al., 2022; Lin et al., 2023b), with most methods using touch and proprioception data to iteratively refine object pose models through filtering theory. However, these approaches, relying on sensitive tactile feedback and lacking vision's global perspective, often struggle with consistent performance, underlining the challenges of relying solely on tactile sensing. For a broader and more comprehensive review of robotic tactile perception, readers are directed to (Li et al., 2020).

2.3. Object Pose Estimation from Vision and Touch

Recent advancements in visual-tactile 6D object-in-hand pose estimation, initiated by the simulation dataset from (Dikhale et al., 2022) and extended by (Li et al., 2023) and (Rezazadeh et al., 2023), have shown promise. However, these studies, including simplistic simulations of vision and touch, reveal a significant domain gap and highlight the need for more realistic data synthesis and real-world collection. Contributions like the small-scale dataset from (Tu et al., 2023) using a Shadow hand with BioTac sensors (sha), (Liu et al., 2023)'s sequential visual-tactile tracking framework, and (Suresh et al., 2023)'s exploration with DIGIT sensors (Lambeta et al., 2020) on the AllegroHand (SimLab, 2016), demonstrate the growing interest in this field. Yet, these works have certain limitations in real-world applications, such as limited tactile data from fingertips only, reliance on two-finger grippers or external cameras to prevent occlusions, and stationary palm (comparison in Table 1). These highlight current research gaps and the pressing need for a comprehensive and publicly available dataset that realistically represents multi-finger object-in-hand pose for robotic manipulation in real-world settings.

3. VinT-6D Dataset

We introduce the VinT-6D dataset, a comprehensive large-scale collection that combines two distinct data types: VinT-Sim and VinT-Real. VinT-Sim, designed to realistically synthesize and minimize the sim-to-real gap, contributes two million visual, tactile, and proprioceptive samples. Meanwhile, VinT-Real has 0.1 million high-quality, real-world data instances, all meticulously gathered from a precisely calibrated robotic system. The dataset features 25 household objects selected for their size, material, and utility diversity.

3.1. VinT-Sim

Collecting large-scale data for robotic in-hand perception presents considerable challenges due to the time-intensive process, sensor wear and tear, pose imprecision, and difficulties in aligning different modalities. Thus, the simulator plays a crucial role in synthesizing such extensive data, leading to our motivation to build the VinT-Sim split. A major challenge for such a collection is narrowing the gap between
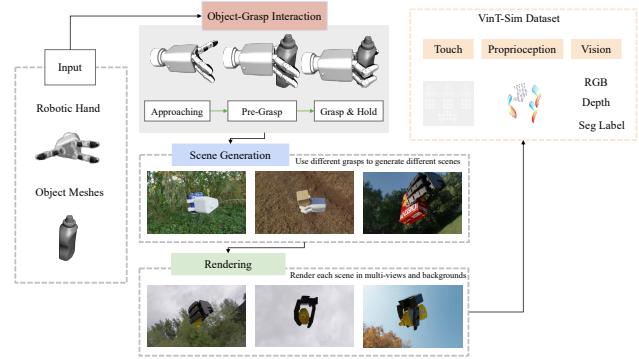

Figure 2. **VinT-Sim Dataset Generation Pipeline.** VinT-Sim requires a robotic hand as input, which can have either three or four fingers along with an object model. There are three components involved in this process: (1) Simulating whole-hand touch. (2) Generating tactile data and proprioception information through object-grasp interactions. (3) Rendering each object-grasp scene with various realistic backgrounds and capturing multiple views.

simulation and reality, approached from three perspectives: touch, proprioception, and vision. The process of generating datasets is illustrated in Figure 2.

3.1.1. SIMULATING TOUCH

Difficulties of simulating tactile modality. Exploring tactile sensors, such as pressure and vision-based sensors, presents challenges for accurate and consistent representation in simulation with their unique characteristics. Concretely, the non-linear positive correlation to applied pressure for each taxel complicates direct simulation efforts, and the sensor performance degradation over time affects reading fidelity, necessitating dynamic simulation adjustments. Additionally, inherent performance differences between taxels due to manufacturing inconsistencies require individual calibration, making the direct simulation of tactile readings non-trivial.

Consistent representation of tactile readings. We focus on the simulation of contact positions rather than mimicking each taxel's specific measurements. For both pressure and vision-based tactile sensors, well-calibrated contact positions can be robust to contact force and time variation. Our approach involves simulating each taxel as a force sensor with a non-linear response to the applied force. For each tactile sensor, we represent it as a cuboid cylinder topped with a blue hemisphere on the finger to generate touch signals and sensor reading upon contact and record the activated tactile sensor's position as the tactile data. For a detailed insight into the simulating touch sensor modeling and calibration process, please refer to Appendix A.1.2.

Alignment to taxel distribution in real world. For all tactile sensors on the whole hand, it is essential to carefully calibrate the robot's hand, tactile sensors, and their spatial
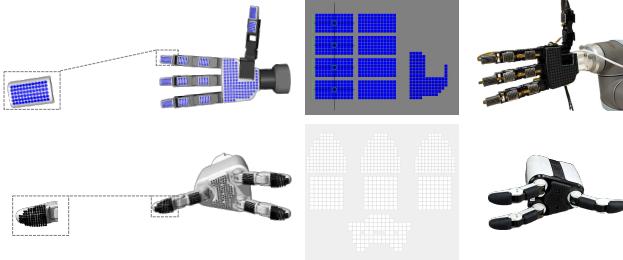
Figure 3. **Simulated and Real-world Robotic Hands with Whole-Hand Tactile Perception.** In VinT-6D, both the three-fingered Trx hand and the four-fingered Allegro hand are used to generate or collect datasets. These robotic hands are equipped with array-based tactile sensors covering the entire hand, with the simulated sensors distributed similarly to the real-world setup.

relationship to accurately replicate tactile contact in simulations. For our real-world setup (refer to Figure 3), we equip two robotic hands (three-fingered Trx and four-fingered Allegro) with array-based tactile sensors. These sensors, with 620 and 679 taxels respectively, are piezoresistive, curved, and unlike flat arrays, have various taxel distributions due to being equipped on the surface of the whole hand. We meticulously scan each sensor's location relative to the hand's parts to ensure accurate replication of taxel distribution in the simulation. When the robotic hand holds an object, the activated tactile positions are recorded as a local touch point cloud (D_{touch}) relative to the palm frame, providing a robust and precise simulated tactile signal.

3.1.2. SIMULATING OBJECT-GRASP INTERACTION

Various object-grasp types and hand configurations influence the diversity of the dataset. Therefore, our simulation should include appropriate grasp types and account for different hand configuration

Selected stable grasping. Recent advances, such as (Turpin et al., 2023) and (Wan et al., 2023), have significantly contributed to the generation of stable multi-finger grasps. However, these advancements have primarily focused on grasp stability and have not taken post-grasp task-specific manipulation into account. As a result, their practical application in real-world robotic manipulation might be limited. This leads us to ask: What form of interaction is required between a robotic hand and an object to enable grasping for manipulation? Our answer is selected stable grasping involves not only stable holding of the object but also readiness for manipulation. We develop a MuJoCo-based simulation for generating realistic object-grasp interactions. In this setup, we carefully fix the robotic hand and the object in predetermined poses within the simulation environment. To ensure a lifelike pre-grasp stance, we tangentially align the robotic hand's palm with a randomly selected point on the object's graspable surface. By employing inverse kinematics to move the finger joints, our system allows for tactile sensor contact with the object. Additionally, the proximity between

the hand and the object follows a set ratio, further enhancing the simulation's realism.

Replicating real-world scenarios. Our approach to grasping objects is based on replicating real-world scenarios to achieve effective tactile engagement. By conducting numerous grasping processes for two hands with different configurations (three-fingered and four-fingered), we have been able to create a dataset of 125000 unique successful graspings. Our focus has been ensuring the grasp's stability and usefulness for future manipulative tasks. To achieve this, the hand executes a power grasp upon contact, lifting the object to a predetermined height for a physically realistic integration. We meticulously document the proprioception data, including the poses of the object (P_{obj}), hand (P_{hand}), and finger knuckles ($P_{knuckle}$), for subsequent vision rendering. The object-grasp setups are detailed examples, which can be found in Appendix A.1.3.

3.1.3. SIMULATING VISION

Generating realistic visual data is essential for bridging the gap between simulation and real-world scenarios (Movshovitz-Attias et al., 2016). While MuJoCo provides accurate physical simulations, its basic rendering capabilities limit the visual realism of the simulated environments.

Photo-realistic object-in-hand rendering. To achieve this, we leverage Blender's advanced rendering features, including ray tracing, diverse shaders, and real-time viewport rendering through its Cycles engine. Based on recorded proprioception data, we can create photo-realistic object-in-hand images by importing object and hand models into Blender and setting their poses and finger knuckle positions. To enhance realism, we use various HDRI backgrounds for illumination and diverse scene backdrops, rendering each setup from multiple viewpoints within a hemispherical range. To ensure authenticity, we also carefully manage ray and glossy reflection bounces in reflective objects. As illustrated in Appendix A.1.4, these techniques enable us to generate highly realistic visual data, improving domain adaptation.

Real-world camera characteristics. We use a multi-step process to enhance the realism of our simulated depth images. We start by rendering depth images in Blender and then apply post-processing techniques to simulate real-world conditions, specifically for those captured by a Kinect Azure TOF camera. We consider various factors such as distance, lighting, and surface specularity. To enhance realism, we introduce noise, create holes, and smooth the rendered depth images, following methodologies like those in (Tolgyessy et al., 2021). By replicating the Kinect Azure sensor's unique noise characteristics, we aim to accurately recreate real-world conditions in our simulations. Finally, we convert the depth images to point clouds and align them
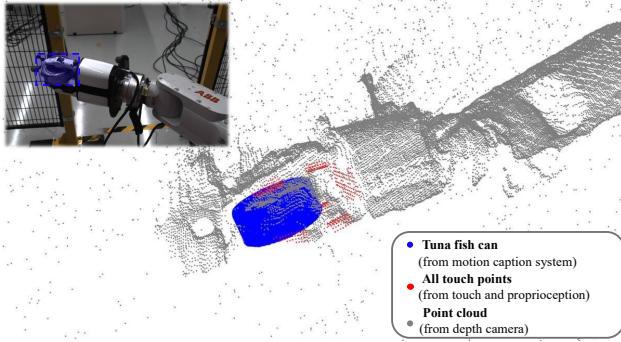
Figure 4. **Visualization of full tactile points aligned with vision in VinT-Real.** Gray points represent the point cloud from the depth camera, blue points depict the transformed model from the motion capture system, and red points indicate the full touch points of the hand.

with the local touch point clouds at the camera frame. This process is crucial for ensuring our depth images are as realistic and accurate as possible.

3.2. VinT-Real

To acquire accurate robotic in-hand perception data, it is crucial to establish a reliable hardware-software platform as well as sensor calibration and maintain precise data alignment. We detail our approach to ensure the quality and diversity of the collected data.

3.2.1. MODALITY ALIGNMENT

Intra-modal Alignment. VinT-Real needs to align depth images from an Azure Kinect TOF camera with RGB images from a stereo camera setup. We achieve this through precise camera calibration and distortion correction, resulting in well-aligned vision images. However, segmenting objects in hand can be a challenging task due to occlusions. To overcome this, our solution utilizes touch and proprioception as cues for the SAM segmentation model (Kirillov et al., 2023). The SAM model is informed of activated touch points identified via robotic kinematics, which improves the segmentation process, as illustrated in Figure 16. You can find more information on our real vision application in Appendix A.2.3.

Inter-modal Alignment. We use touch positions to create a touch point cloud that is then translated into real-world applications through a calibrated robotic system. We calculate the hand pose relative to the robot's base using forward kinematics, starting from the finger knuckles to the hand frame. Our approach accurately maps activated touch sites onto the knuckles' positions to establish the touch point cloud. We ensure sub-millimeter accuracy object-in-hand pose (Appendix A.2.4) by augmenting the Trx-hand with a flange plate featuring markers, which are captured by a motion cap-

Figure 5. **Custom-Developed Robot Platform.** The objects selected in VinT-Real are neatly placed on the desk for easy access and manipulation.

ture system. Our custom-designed fixtures equipped with markers for each object guarantee precise and consistent pose data when objects are held in hand. We make all fixture models available on our website. Our well-aligned multi-modal data is visualized in Figure 4. Overall, our approach surpasses common ArUco tag-based methods (Calli et al., 2017; Dikhale et al., 2022; Xu et al., 2023) and guarantees precise alignment with the vision system (Appendix A.2.3).

3.3. Dataset Analysis

3.3.1. OBJECT CATEGORIES

As we aimed to design VinT-6D, we established three primary selection criteria for the objects that would be used in the project. They should be easy to hold, commonly found in daily life, and made of various materials such as plastics, metal, and glass. This selection reflects the growing emphasis on domestic robotics applications, especially in healthcare and elderly assistance. At first, we selected 21 objects from the YCB-Video dataset (Calli et al., 2017), considering their hand-held size and influence in robotics. However, we also recognized the limited availability and variety of objects, particularly outside the US. We added five more everyday objects with transparent and reflective features to address this, which you can find in Figure 18. Then, we carefully selected 10 common items from them, such as a can of tomato soup or a bottle of mustard (Figure 19). Our main objective was to identify their functional requirements and the challenges that arise when handling them. These items include a metal shaker and a wine glass, which come in different sizes and materials, and some of them require a certain level of skill when it comes to rotating them or holding them with just the fingertips.
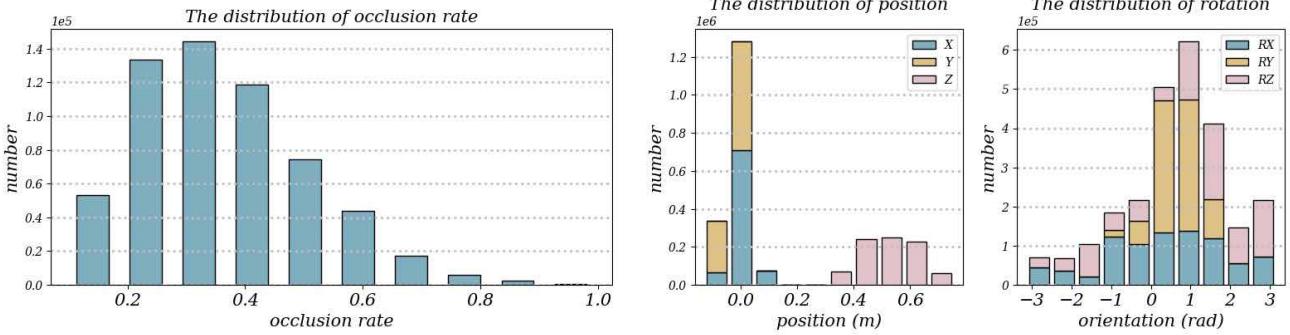
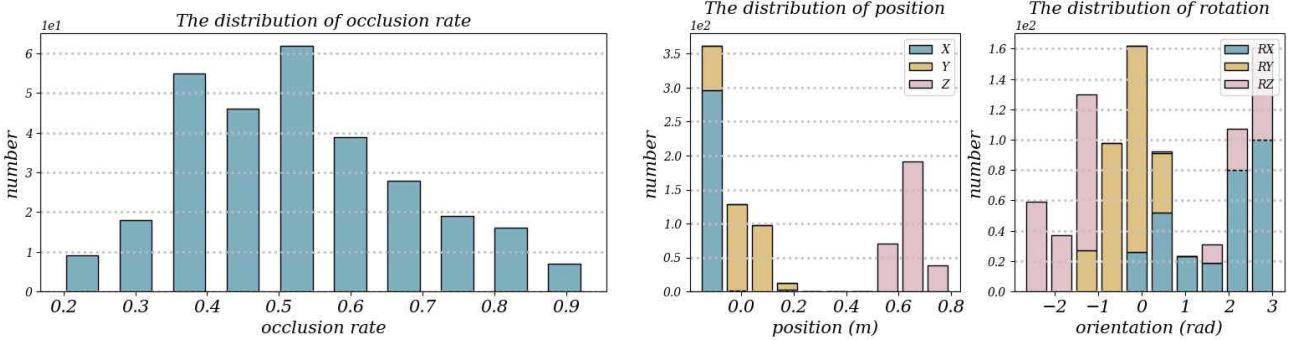*Figure 6.* **Distribution of occlusion rate and camera poses in VinT-Sim.**

Figure 7. **Distribution of and occlusion rate and object poses in VinT-Real.**

3.3.2. GENERALIZATION OF ROBOTIC HANDS

The use of different multi-fingered hands in robotics presents unique challenges that must be addressed. Robotic vision, touch, and proprioception can vary significantly depending on the configuration of the hands used, as each configuration provides different proprioception information and activated taxels. Moreover, the size and appearance of the hand can also impact the robotic vision. To overcome these challenges and enhance the generalization of robotic hands, VinT-Sim incorporates two widely recognized types of hands - three-fingered and four-fingered. The Trx hand, with its three fingers, eight degrees of freedom, is our own custom-made solution. The Allegrohand, on the other hand, is a commercially available four-fingered hand with 16 degrees of freedom. In VinT-Real, we use the Trx hand at the end of the robot arm.

3.3.3. WHOLE-HAND TACTILE PERCEPTION

VinT-6D is a unique dataset that uses whole-hand tactile perception, which sets it apart from other datasets that rely on fingertip tactile sensing alone. By strategically placing array-based tactile sensors across the fingertips, pulp, and palm, this dataset offers comprehensive local information during object-hand interaction, particularly in scenarios

where vision is obstructed. The Trx hand features 620 taxels, while the Allegro hand boasts 679 taxels, covering the entire hand. This promises to significantly advance robotic in-hand perception capabilities through extensive area contact.

3.3.4. DIVERSITY OF AVAILABLE INPUT MODALITIES

VinT-6D provides researchers with various input modalities to aid further exploration. In VinT-Sim, researchers can access a range of visual inputs such as color images, depth images, and segmentation labels. Moreover, VinT-Sim also offers tactile point cloud and proprioception data, such as finger poses relative to the palm. In Vint-Real, we collect the left and right eye images from a stereo camera, depth and IR images from a Kinect Azure camera, touch readings from tactile sensors, finger joint angles from Trx-hand, and object and hand poses from the Vicon motion capture system, as demonstrated in Figure 5. These diverse modalities offer promising directions for specific in-hand perception like transparent object pose estimation and reconstruction or perceiving liquid using IR images.

3.3.5. WELL-CALIBRATED ROBOTIC PLATFORM

Achieving high-quality data collection demands a well-calibrated robotic platform with all sensors aligned in

Table 2. **Object-in-hand segmentation accuracy with different occlusion levels in VinT-Real.**

OCCLUSION RATE	20%	30%	40%	50%
MIOU	96.41	94.44	96.63	95.65

timestamps and coordinate frames. Precise alignment of multi-sensors ensures accurate data collection, particularly when acquiring segmentation labels as explained in Appendix A.2.3. To guide the SAM modal (Kirillov et al., 2023) in object segmentation from hand, we assign 'add' and 'remove' prompts, which use touch and proprioception information to guide visual segmentation. Aligning the timestamps and coordinates of multi-sensors is crucial as it makes it possible to segment the object from the hand.

3.3.6. ROBUST SEGMENTATION TOWARDS OCCLUSION

Segmenting the in-hand object from vision remains a significant challenge. In our dataset, we concentrate on achieving robust object segmentation reliable object segmentation even when the robotic hand occludes a portion of the object. To address this, we have employed multi-modal prompts in SAM to differentiate between objects and hands. Specifically, we use negative ("-") for thumb and positive ("+") for index and middle finger tactile points, along with two object points. This mirrors typical object-in-hand poses collected in real-world scenarios. To evaluate SAM's reliability when dealing with occlusions ranging from 20% to 50% on a 'tomato soup can' from VinT-Real, we conducted a comparison against manually segmented ground truth across 100 sets of data. The results are presented in Table 2.

3.3.7. DATA DIVERSITY

We created 2,500 interactions for each object in VinT-Sim, resulting in a wide range of poses. These poses were rendered in Blender across four different environments, with three camera positions for each. This led to a total of 2 million visual-tactile data sets. The dataset itself includes finger-induced occlusions ranging from 10% to 90%, as well as various camera poses (Figure 6) and object poses (Figure 7).

To diversify VinT-Real, we densely sample within the robot's workspace, akin to a toddler's exploratory play. Our data collection mirrors this process, capturing object-in-hand scenarios using vision, touch, and proprioception. We establish 125 key positions relevant to the robot and camera workspaces in the Cartesian space and introduce 8 hand rotations at each, totaling 1000 unique poses. These poses, designed to avoid obstruction by fixtures with markers and to be solvable via inverse kinematics, are repeatedly adjusted across ten variations per object. This ensures a diverse touch point collection and poses variability. We control hand tra-

jectories and arm movements through robotic motion planning, amassing 100,000 well-aligned data instances across 10 objects, encompassing vision, touch, and proprioception.

4. VinT-Net

We introduce VinT-Net to enhance research in robotic object-in-hand pose estimation and validate the proposed dataset. This network acts as a simple yet effective baseline for multi-finger object-in-hand pose estimation challenges. VinT-Net can efficiently process various inputs, such as RGB and depth images and local touch points derived from robotic tactile and proprioceptive sensors. The architecture of VinT-Net comprises two crucial sub-modules: a sensing aggregation module (refer to Section 4.1) and a 3D keypoint-based pose estimation module (see Section 4.2). The framework of VinT-Net is illustrated in Figure 8. It accurately estimates the object's 6D pose from the robot's camera perspective.

4.1. Sensing Aggregation Module

In the Sensing Aggregation Module, our focus is on leveraging color images, depth images, and locally computed touch data derived from proprioceptive and touch sensors. Utilizing the robot's forward kinematics, detailed in Figure 8, we obtain the precise local touch point data D_{touch}. The module employs three separate branches for processing: the color vision data D_{color} undergoes processing via a U-Net-based model (Ronneberger et al., 2015) for appearance feature extraction. Simultaneously, the depth vision data D_{depth} and the local touch data D_{touch} are processed using individual PointNet++ (Qi et al., 2017) networks, isolating local and global geometric features. Following the principles of advanced vision fusion techniques (Wang et al., 2019), we integrate the color and depth information on a pixel-wise level. The local touch data, treated as a complementary element to the depth image, is fused with the depth features to produce a comprehensive visual feature F_v and a fused visual-tactile feature F_{vt}, as detailed in Figure 8.

4.2. 3D Keypoint-Based Pose Estimation Module

It has been observed that in pose estimation of rigid objects, the 3D spatial relationship between any two points on the object remains constant, regardless of the object's movements. This observation has been noted in (He et al., 2020; 2021). As a result, it has been found that predicting selected 3D key points on the object based on visible surface appearances in a camera view is effective. However, occlusions can significantly impair vision when a robotic hand grasps an object. To address this issue, our method uses fused multi-modal embeddings to predict both the object's central point and its keypoint offsets. A multi-modal fusion module is used to extract per-point features from vision and touch inputs, which are then fed into three MLP heads. These
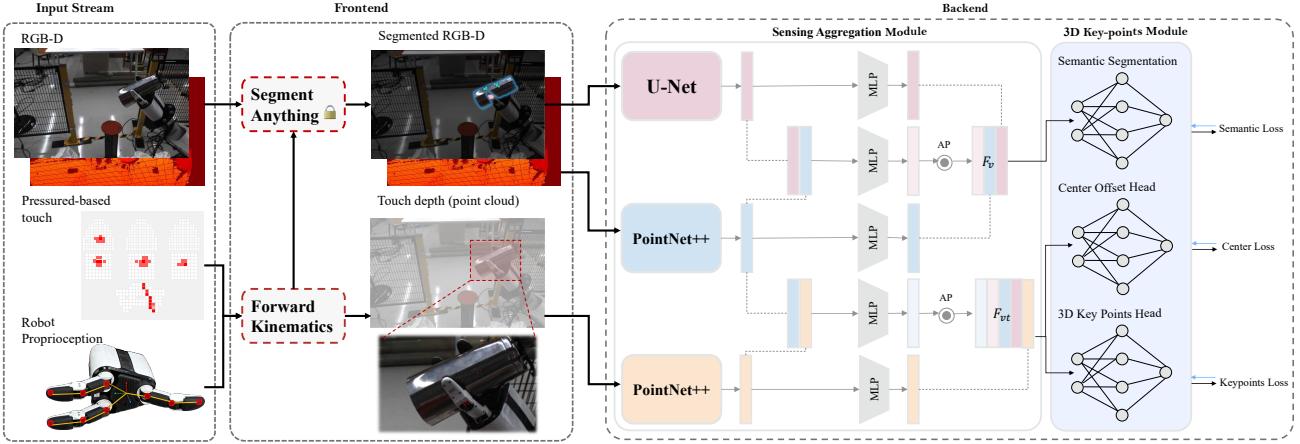
Figure 8. **Overview of VinT-Net.** The Frontend receives well-aligned vision and raw touch and proprioception data. Using SAM (Kirillov et al., 2023) and hand's forward kinematics, it acquires the segmentation label and touch point cloud separately. Then, the Backend predicts the object's pose.

heads are responsible for predicting semantic labels, center, and keypoint offsets. Interestingly, our experiments showed that integrating touch data into the semantic segmentation head did not enhance performance. This may indicate that additional touch information may interfere with the head's functionality. We apply Focal Loss (Lin et al., 2017) for semantic segmentation supervision and $L1$ Loss for center and 3D keypoint offset prediction. A combined multi-task loss augments the overall model accuracy :

$$L_{\text{multi-task}} = \lambda_1 L_{\text{semantic}} + \lambda_2 L_{\text{keypoints}} + \lambda_3 L_{\text{center}} \quad (1)$$

where λ_1, λ_2 and λ_3 are the weights for each task.

In the final step, following (He et al., 2020; 2021), a clustering algorithm is employed to distinguish between instances with identical semantic labels. A least-squares fitting algorithm accurately determines the 6D pose parameters.

$$L_{\text{least-squares}} = \sum_{j=1}^{M} ||kp_j - (R \cdot kp_j' + t)||^2 \quad (2)$$

where M represents the pre-selected 3D keypoints of an object, R symbolizes rotation, and t represents translation.

5. Experimental Results

5.1. Experimental Setup

Implementation Details. We used the Adam optimizer with an initial learning rate of 0.01 for training and set the batch size at 24. The training process was conducted over 25 epochs, and we set the hyper-parameters λ_1, λ_2, and λ_3 to 1, 2, and 1, respectively. The training and testing processes were executed on a computing server equipped with 6 Quadro RTX 8000 GPUs. The VinT-Sim synthesis procedures were conducted by a cloud computing platform that utilized 16 NVIDIA P40 GPUs.

Table 3. **Quantitative Evaluation on VinT-6D Dataset.** ADD(S) AUC metric is reported.

	SIM	REAL	VINT-6D
BLUE BOTTLE	87.52	93.45	94.15
LARGE SHAKER	88.04	94.66	96.23
STICK	78.04	83.72	83.76
POTTED MEAT CAN	88.56	94.31	95.53
TOMATO SOUP CAN	89.77	94.65	95.44
POWER DRILL	88.04	91.72	97.09
TUNA FISH CAN	90.54	92.34	93.89

Evaluation Metrics. In evaluation, we follow (He et al., 2020) and evaluate our method with the ADD and ADD-S metrics, as defined in (Xiang et al., 2017). The ADD metric measures the average distance between the object's vertices transformed by the predicted 6D pose $[R, t]$ and the corresponding vertices transformed by the ground truth pose $[R^*, t^*]$.

$$\text{ADD} = \frac{1}{m} \sum_{x \in \mathcal{O}} ||(Rx + t) - (R^* x + t^*)|| \quad (3)$$

where x is one of the m vertexes on the object mesh $\mathcal{O}$. The ADD-S metric is used for symmetrical objects; it computes the average distance based on the closest point distance.

$$\text{ADD-S} = \frac{1}{m} \sum_{x_1 \in \mathcal{O}} \min_{x_2 \in \mathcal{O}} ||(Rx_1 + t) - (R^* x_2 + t^*)|| \quad (4)$$

We compute the ADD(S) AUC, the area under the accuracy-threshold curve, which is obtained by varying the ADD distance threshold for non-symmetric objects and ADD-S distance for symmetric objects.

5.2. Evaluation on VinT-6D

We conduct experiments to compare the performance of models trained on purely simulated data, real-world data,

Table 4. **Ablation Studies of Vision and Touch in VinT-Real.** 'Touch' comprises touch and proprioception data.

OBJECTS	VISION	VISION + TOUCH
BLUE BOTTLE	90.34	93.45
LARGE SHAKER	87.04	94.66
STICK	78.04	83.72
POTTED MEAT CAN	91.23	94.31
TOMATO SOUP CAN	86.78	94.65
POWER DRILL	90.04	91.72
TUNA FISH CAN	85.81	92.34

Table 5. **Comparison to other object-in-hand methods on 'tomato soup can'.** ADD-0.05d metric is reported.

METHODS	ADD
PVN3D (HE ET AL., 2020)	74.60
OBJECT-HAND-POSE (WEN ET AL., 2020)	76.74
OURS	82.43

and a combination of both. The results of these experiments are presented in Table 3. The results show that our comprehensive calibration and simulation strategy effectively bridges the sim2real gap by augmenting the simulated data to the real data, as evidenced by the performance improvement when combining simulated and real data for training.

5.3. A Comprehensive Analysis

We first ablate the benefits of touch and proprioception on the dataset, then compare the results with other object-in-hand pose estimation methods, and finally evaluate the robustness under various occlusion levels.

Benifit of touch and proprioception. This ablation study evaluates the contributions of tactile and proprioception modalities in VinT-Real. In Table 4, we compare our visual-tactile Vint-Net with the visual baseline (He et al., 2020). Our visual-tactile VinT-Net surpasses the baseline relying solely on vision by a margin, demonstrating the incorporation of additional tactile information notably enhances performance.

Comparison to other object-in-hand pose estimation methods. Table 5 shows that our VinT-Net significantly outperforms recent object-in-hand pose estimation methods like (Wen et al., 2020) on the ADD-0.05d metric, which notably employs hand state estimation to enhance pose accuracy. To ensure a fair comparison given hardware differences, we reproduced the Object-Hand-Pose method by replacing its two-finger gripper with our three-finger hand, which has reduced degrees of freedom. This and other adjustments were made to estimate the 'tomato soup can' pose within our VinT-Real dataset, as shown in Table 5.

Robustness to occlusion by the multi-fingered hand. To prove the robustness against occlusion caused by the multi-fingered hand, we further investigate the 'Tomato Soup Can' under different occlusion levels. Table 6 reveals a notable

Table 6. **Robustness to occlusion by the multi-fingered hand on 'tomato soup can.'** ADD-0.05d metric is reported.

OCCLUSION RATE	20%	30%	40%	50%
VISION ONLY	93.31	89.45	85.77	80.43
VISION + TOUCH	94.76	93.89	90.47	88.81

decrease in accuracy for vision-based approaches as occlusion intensifies, from 93.31% at 20% occlusion to 80.43% at 50% occlusion. Conversely, the multi-modal fusion strategy, which leverages both vision and tactile inputs, exhibited remarkable resilience against increasing occlusion levels. Its accuracy slightly dipped from 94.76% at 20% occlusion to 88.81% at 50% occlusion.

6. Conclusion

We contribute VinT-6D, a pioneering multi-modal dataset for 6D object-in-hand pose estimation, by integrating vision, touch, and proprioception. With over 2 million and 0.1 million synthesized (VinT-Sim) and real data (VinT-Real), VinT-6D is tailored for robotic hands, providing high-quality, well-aligned data for accurate object-in-hand pose estimation. Our benchmark method leveraging VinT-6D showcases notable performance improvements, highlighting the dataset's potential to bridge the gap between simulated and real-world applications.

7. Limitations and Future Work

While we have endeavored to narrow the sim2real gap, we acknowledge that our existing VinT-6D dataset lacks diversity in terms of objects and scenes. As part of our continuous research efforts, we intend to introduce a broader range of elements and more varied object-in-hand grasping scenarios in the future. Furthermore, our proposed VinT-Net represents an instance-level pose estimation approach that employs a simple fusion strategy for dataset validation. This leaves a large space to explore category-level or zero-shot object-in-hand pose estimation, which could potentially benefit from a more advanced fusion of multiple modalities.

Impact Statement

The imminent release of VinT-6D marks a substantial contribution to the field, promising to enhance research and development in robotic manipulation.

Acknowledgement

References

Polyhaven. https://polyhaven.com. Accessed: 2024-02-01.

Shadowhand. https://ninjatek.com/shop/edge/. Accessed: 2024-02-01.

Calli, B., Singh, A., Bruce, J., Walsman, A., Konolige, K., Srinivasa, S., Abbeel, P., and Dollar, A. M. Yale-cmu-berkeley dataset for robotic manipulation research. *International Journal of Robotics Research*, 36(3):261–268, 2017.

Chen, H., Wang, P., Wang, F., Tian, W., Xiong, L., and Li, H. Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2781–2790, 2022.

Chen, W., Jia, X., Chang, H. J., Duan, J., and Leonardis, A. G2l-net: Global to local network for real-time 6d pose estimation with embedding vector features. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4233–4242, 2020.

Community, B. O. *Blender - a 3D modelling and rendering package*. Stichting Blender Foundation, Amsterdam, 2018. URL http://www.blender.org.

Cui, S., Wang, R., Hu, J., Wei, J., Wang, S., and Lou, Z. In-hand object localization using a novel high-resolution visuotactile sensor. *IEEE Transactions on Industrial Electronics*, 69(6):6015–6025, 2021.

Dikhale, S., Patel, K., Dhingra, D., Naramura, I., Hayashi, A., Iba, S., and Jamali, N. Visuotactile 6d pose estimation of an in-hand object using vision and tactile sensor data. *IEEE Robotics and Automation Letters*, 7(2):2148–2155, 2022.

Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F. J., and Marín-Jiménez, M. J. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.

He, Y., Sun, W., Huang, H., Liu, J., Fan, H., and Sun, J. Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation. In *IEEE/CVF Conference on Computer Vision and pattern Recognition*, pp. 11632–11641, 2020.

He, Y., Huang, H., Fan, H., Chen, Q., and Sun, J. Ffb6d: A full flow bidirectional fusion network for 6d pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3003–3013, 2021.

Hinterstoisser, S., Holzer, S., Cagniart, C., Ilic, S., Konolige, K., Navab, N., and Lepetit, V. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In *International Conference on Computer Vision*, pp. 858–865, 2011.

Kelestemur, T., Platt, R., and Padir, T. Tactile pose estimation and policy learning for unknown object manipulation. *arXiv preprint arXiv:2203.10685*, 2022.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

Lambeta, M., Chou, P.-W., Tian, S., Yang, B., Maloon, B., Most, V. R., Stroud, D., Santos, R., Byagowi, A., Kammerer, G., et al. Digit: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation. *IEEE Robotics and Automation Letters*, 5(3):3838–3845, 2020.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *ICML*, pp. 1207–1216, 2000.

Li, H., Dikhale, S., Iba, S., and Jamali, N. Vihope: Visuo-tactile in-hand object 6d pose estimation with shape completion. *IEEE Robotics and Automation Letters*, 2023.

Li, Q., Kroemer, O., Su, Z., Veiga, F. F., Kaboli, M., and Ritter, H. J. A review of tactile information: Perception and action through touch. *IEEE Transactions on Robotics*, 36(6):1619–1634, 2020.

Li, Z., Wang, G., and Ji, X. Cdpn: Coordinates-based disentangled pose network for real-time rgb-based 6-dof object pose estimation. In *IEEE/CVF International Conference on Computer Vision*, pp. 7678–7687, 2019.

Liang, J., Handa, A., Van Wyk, K., Makoviychuk, V., Kroemer, O., and Fox, D. In-hand object pose tracking via contact feedback and gpu-accelerated robotic simulation. In *IEEE International Conference on Robotics and Automation*, pp. 6203–6209, 2020.

Lin, Q., Yan, C., Li, Q., Ling, Y., Lee, W., Zheng, Y., Wan, Z., Huang, B., and Liu, X. Tracking object's pose via dynamic tactile interaction. *International Journal of Humanoid Robotics*, pp. 2350021, 2023a.

Lin, Q., Yan, C., Li, Q., Ling, Y., Zheng, Y., Lee, W., Wan, Z., Huang, B., and Liu, X. Tactile-based object pose estimation employing extended kalman filter. In *2023 International Conference on Advanced Robotics and Mechatronics*, pp. 118–123, 2023b.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision*, pp. 2980–2988, 2017.

Lipson, L., Teed, Z., Goyal, A., and Deng, J. Coupled iterative refinement for 6d multi-object pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6728–6737, 2022.

Liu, Y., Xu, X., Chen, W., Yuan, H., Wang, H., Xu, J., Chen, R., and Yi, L. Enhancing generalizable 6d pose tracking of an in-hand object with tactile sensing. *IEEE Robotics and Automation Letters*, 2023.

Movshovitz-Attias, Y., Kanade, T., and Sheikh, Y. How useful is photo-realistic rendering for visual learning? In *ECCV Workshops*, pp. 202–217, 2016.

Peng, S., Liu, Y., Huang, Q., Zhou, X., and Bao, H. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4561–4570, 2019.

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in Neural Information Processing Systems*, 30, 2017.

Qi, H., Yi, B., Suresh, S., Lambeta, M., Ma, Y., Calandra, R., and Malik, J. General in-hand object rotation with vision and touch. In *Conference on Robot Learning*, pp. 2549–2564, 2023a.

Qi, L., Kuen, J., Shen, T., Gu, J., Li, W., Guo, W., Jia, J., Lin, Z., and Yang, M.-H. High quality entity segmentation. In *IEEE/CVF International Conference on Computer Vision*, pp. 4047–4056, 2023b.

Rezazadeh, A., Dikhale, S., Iba, S., and Jamali, N. Hierarchical graph neural networks for proprioceptive 6d pose estimation of in-hand objects. In *IEEE International Conference on Robotics and Automation*, pp. 2884–2890, 2023.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, 2015.

Röstel, L., Sievers, L., Pitz, J., and Bäuml, B. Learning a state estimator for tactile in-hand manipulation. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4749–4756, 2022.

SimLab. Allegro Hand Overview. https://www.simlab.com/products/allegro-hand, 2016. [Online; accessed 25-May-2020].

Smith, E., Calandra, R., Romero, A., Gkioxari, G., Meger, D., Malik, J., and Drozdzal, M. 3d shape reconstruction from vision and touch. *Advances in Neural Information Processing Systems*, 33:14193–14206, 2020.

Suresh, S., Qi, H., Wu, T., Fan, T., Pineda, L., Lambeta, M., Malik, J., Kalakrishnan, M., Calandra, R., Kaess, M., et al. Neural feels with neural fields: Visuo-tactile perception for in-hand manipulation. *arXiv preprint arXiv:2312.13469*, 2023.

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

Tolgyessy, M., Dekan, M., Chovanec, L., and Hubinsky, P. Evaluation of the azure kinect and its comparison to kinect v1 and kinect v2. *Sensors*, 21(2):413, 2021.

Tu, Y., Jiang, J., Li, S., Hendrich, N., Li, M., and Zhang, J. Posefusion: Robust object-in-hand pose estimation with selectlstm. *arXiv preprint arXiv:2304.04523*, 2023.

Turpin, D., Zhong, T., Zhang, S., Zhu, G., Liu, J., Singh, R., Heiden, E., Macklin, M., Tsogkas, S., Dickinson, S., et al. Fast-grasp'd: Dexterous multi-finger grasp generation through differentiable simulation. *arXiv preprint arXiv:2306.08132*, 2023.

Tyree, S., Tremblay, J., To, T., Cheng, J., Mosier, T., Smith, J., and Birchfield, S. 6-dof pose estimation of household objects for robotic manipulation: An accessible dataset and benchmark. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 13081–13088, 2022.

Wan, W., Geng, H., Liu, Y., Shan, Z., Yang, Y., Yi, L., and Wang, H. Unidexgrasp++: Improving dexterous grasping policy learning via geometry-aware curriculum and iterative generalist-specialist learning. *arXiv preprint arXiv:2304.00464*, 2023.

Wang, C., Xu, D., Zhu, Y., Martín-Martín, R., Lu, C., Fei-Fei, L., and Savarese, S. Densefusion: 6d object pose estimation by iterative dense fusion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3343–3352, 2019.

Wang, G., Manhardt, F., Liu, X., Ji, X., and Tombari, F. Occlusion-aware self-supervised monocular 6D object pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Wang, S., Lambeta, M., Chou, P.-W., and Calandra, R. Tacto: A fast, flexible, and open-source simulator for high-resolution vision-based tactile sensors. *IEEE Robotics and Automation Letters*, 7(2):3930–3937, 2022.

Wen, B., Mitash, C., Soorian, S., Kimmel, A., Sintov, A., and Bekris, K. E. Robust, occlusion-aware pose estimation for objects grasped by adaptive hands. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6210–6217. IEEE, 2020.

Xiang, Y., Schmidt, T., Narayanan, V., and Fox, D. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017.

Xu, W., Yu, Z., Xue, H., Ye, R., Yao, S., and Lu, C. Visual-tactile sensing for in-hand object reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8803–8812, 2023.
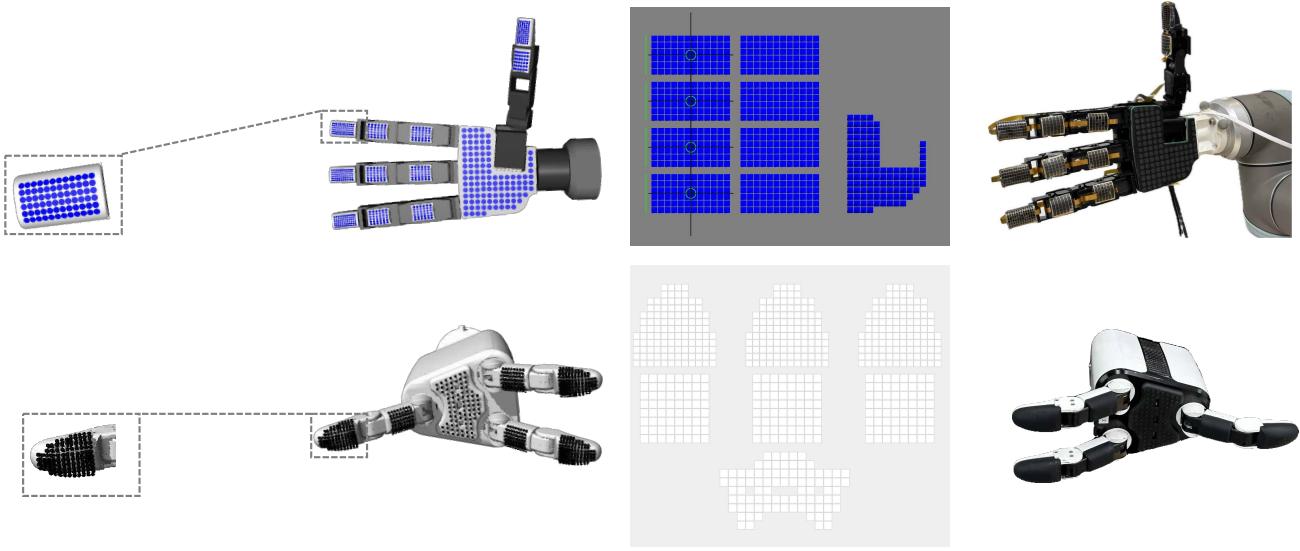
Figure 9. **Simulating touch follows the real-world setup.** In the upper row, we have an AllegroHand equipped with whole-hand tactile sensors, while in the lower row, we have a Trx-hand with sensors. The distribution of these sensors closely follows that of the real-world setup.

A. Appendix.

A.1. VinT-Sim

A.1.1. SIMULATION SETUP

Given the influence of finger configurations on visual and tactile data acquisition, we have included datasets for both three-fingered and four-fingered hand configurations in our simulation. We use our three-fingered TRX-Hand and Wonik's four-fingered AllegroHand, both equipped with pressure-based touch sensors on each finger.

A.1.2. SIMULATING TOUCH

To narrow the sim-to-real gap in touch simulation, we pose two critical questions: (1) How can we represent tactile readings in a sensor-agnostic manner, accounting for different types of tactile sensors and their performance variation over time? (2) How can we accurately simulate the distribution of all taxels (tactile unit sensors)?

To tackle these queries, we first reviewed recent pioneering approaches in tactile simulation, primarily using: (1) (Suresh et al., 2023; Qi et al., 2023a; Smith et al., 2020; Xu et al., 2023) Directly using the simulator (Wang et al., 2022) to generate RGB images for vision-based tactile sensors. (2) (Dikhale et al., 2022; Li et al., 2023; Rezazadeh et al., 2023) pressure-based tactile sensors simulated by attaching depth cameras to each joint of an AllegroHand, with the point cloud serving as tactile feedback. However, these approaches may introduce a significant domain gap compared to real-world tactile sensors, primarily because they do not account for the curved surface of fingers and the typically smaller, more localized actual touch area.

In our real-world setup, as shown in Figure 9, each finger and the palm of the robotic hands are equipped with tactile sensors featuring a piezoresistive array of taxels on a curved, continuous surface. The three-fingered hand has 620 taxels, while the four-fingered hand comprises 679 taxels. Unlike traditional flat tactile arrays, these curved surface sensors distribute tactile elements over an irregular surface, and slight deviations exist in each sensor's tactile element distribution due to manufacturing precision constraints. The response of each taxel is proportional to the applied normal force, displaying a non-linear, positively correlated response pattern. Over time, a greater force is required for the same level of activation.

For individual taxels, we meticulously modeled each taxel as a force sensor within the 3D robotics MuJoCo physics engine (Todorov et al., 2012). Each simulated tactile element is represented as a small, lightweight cuboid cylinder on the finger
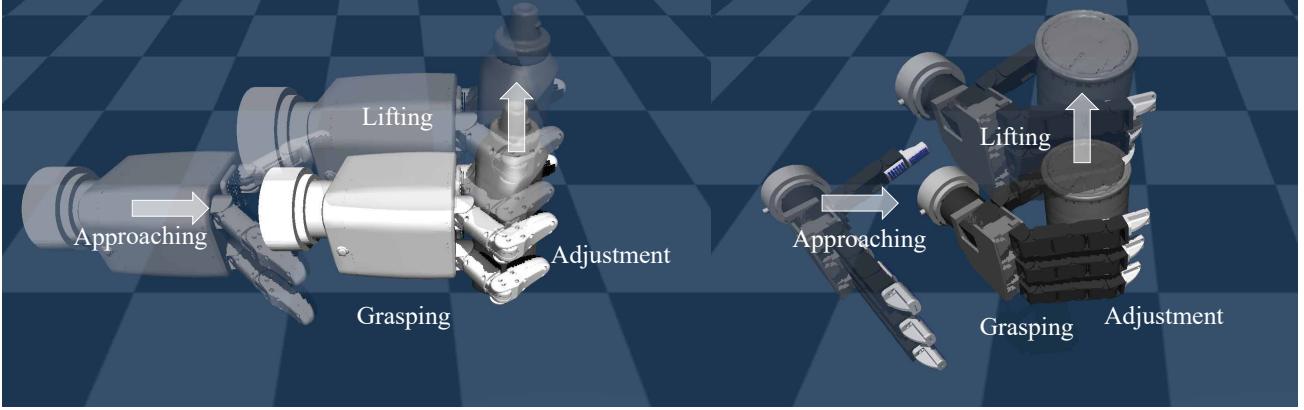
Figure 10. **Simulating object-grasp interaction**. Selected stable grasping requires stability and preparedness for manipulation. Our simulation-based grasping method replicates real-world scenarios to ensure effective tactile engagement.

surfaces, topped with a blue hemisphere, as depicted in Figure 9. Upon tactile contact, the simulation generates a touch signal (T) and corresponding sensor readings (R) at the contact point. These hemispheres, when activated, denote the local surface topology being touched, facilitating interaction with the tactile sensors. Using forward kinematics, the activated location is computed as the touch signal, providing a versatile representation suitable for both pressure-based and vision-based sensors, and accommodating time-varying characteristics.

We aimed to replicate these tactile distributions accurately in our simulations for the overall distribution of taxels. We captured detailed mesh and texture information by meticulously scanning each sensor part with 3D scanning equipment. After exporting the calibrated positional and normal data for all tactile elements, we precisely aligned each element on the hand in the simulation. This approach mirrors real sensor features and distributions, effectively bridging the gap between simulated and real-world environments. We recorded all activated positions as a local touch point cloud (D_{touch}) relative to the palm frame, included for use in the dataset.

A.1.3. SIMULATING OBJECT-GRASP INTERACTION

What types of interactions are essential between a robotic hand and an object it touches? How to narrow down the sim2real gap in proprioception? Our approach to selecting object-grasp interactions is guided by two primary considerations: the ability to hold the object stably in hand and the readiness of the grasp pose for subsequent robotic manipulation tasks. We posit that an effective in-hand grasp should prevent the object from dropping and position it optimally for future manipulative actions.

Notably, recent advancements in state-of-the-art multi-finger object-grasp generation, such as those proposed by (Turpin et al., 2023) and (Wan et al., 2023), have emerged. However, these approaches often do not address the critical aspect of stable holding in hand and readiness for manipulative actions, which are key focuses of our research.

Simulating object-hand setup: To simulate object-grasp interactions that adhere to these specific criteria, we have implemented a setup involving a robotic hand and an object within the MuJoCo physics engine (Todorov et al., 2012). The detailed grasping process is depicted in Figure 10. Initially, the Trx-Hand (or AllegroHand) and the object are placed into the simulation environment in predefined initial poses. The robotic hand is carefully positioned into a pre-grasp stance, ensuring its palm is tangentially aligned with a randomly chosen point adjacent to the object's graspable surface. The proximity between the hand in its pre-grasp position and the object is assigned randomly, adhering to a set ratio. Utilizing inverse kinematics, we articulate the finger joints to facilitate contact between the tactile sensors on the fingers or palm and the object's surface. Upon establishing contact, the robotic hand executes a power grasp and lifts the object to a predetermined height. This action is dependent on the successful and physically realistic integration of the object being held. The grasping process is repeated for each combination of object and hand until a dataset comprising 2,500 successful grasps, each marked by effective tactile contact, is compiled. This setup is designed to closely mirror real-world grasping scenarios, ensuring the grasp's stability and utility for subsequent manipulations. We meticulously record the object pose (P_{obj}), hand pose (P_{hand}), and the poses of all finger knuckles $(P_{knuckle})$ for subsequent vision rendering.
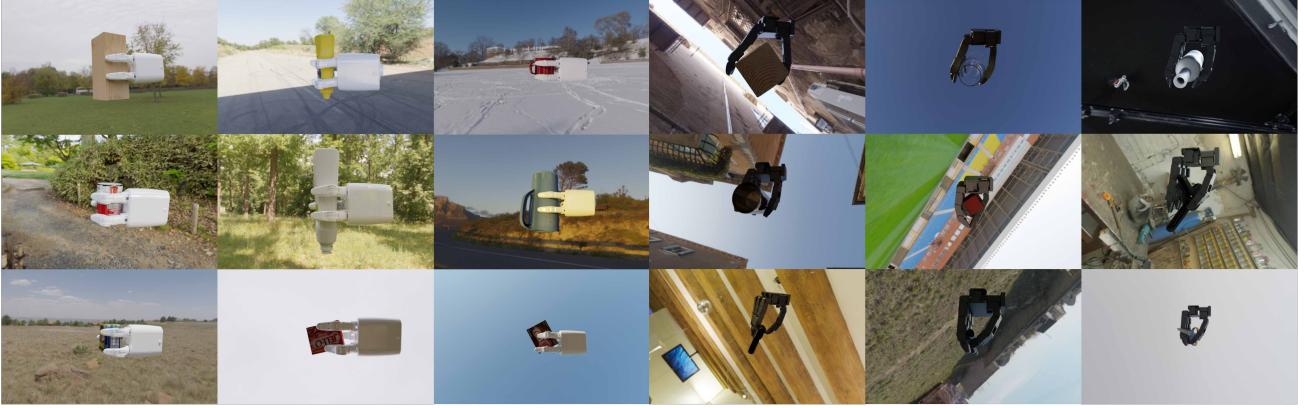
Figure 11. **Photorealistic Rendered RGB Images.** We present a set of photo-realistic rendered images that depict objects held in hand.

A.1.4. SIMULATING VISION

Drawing from our prior experience, we have recognized that images captured in MuJoCo for training frequently exhibit a significant domain gap. This issue predominantly arises from naive renderings, particularly for transparent or reflective objects. To obtain more realistic visual representations of object-hand interactions, we utilize Blender to render two types of images: an RGB image and a depth image, both depicting an object occluded by the robotic fingers grasping it.

Simulating RGB images. We employ Blender's integrated Cycles rendering engine to produce photorealistic scenes, each featuring an object held in hand. Initially, we import both an object and a hand model into Blender, setting their poses as well as the poses of all finger knuckles based on the information recorded during the object-grasp interaction phase. For each object-in-hand configuration, we opt for different HDRI backgrounds (pol) to ensure adequate and realistic illumination and diverse backdrops. We render each scene from multiple viewpoints within a hemispherical range, focusing on the object-in-hand configuration. To achieve realistic rendering, as shown in Figure 11, we restrict the number of ray bounces and the number of glossy reflection bounces.

Simulating Depth images. From a given camera viewpoint, a depth image was initially rendered using Blender's internal depth estimation. In our real-world setup, we equipped the real-world robot with a Kinect Azure TOF camera. The depth readings from this camera are subject to various factors, including the distance between the camera and the object, environmental lighting conditions, and the object's surface specularity. Even under optimal conditions, the depth measurements from the Kinect Azure are influenced by a multitude of hyperparameters, significantly impacting the accuracy of the depth ground truth. To replicate these real-world conditions in our simulation, we introduced noise, created holes, and applied smoothing to the rendered depth image. This methodology, following the approach outlined by Tolgyessy et al. (2021), is specifically adapted to emulate the unique noise characteristics of the Kinect Azure sensor. We convert the depth image for visualization into a point cloud, aligning it with the local touch point cloud at the camera frame, as depicted in Figure 12.

A.2. VinT-Real

A.2.1. ACQUIRING EQUIPMENT

As illustrated in Figure 13, our customized robot platform integrates a comprehensive perception system, including vision, tactile, and proprioceptive systems. The vision system features two industrial color cameras for binocular stereo vision, enhanced by a TOF depth camera from Azure Kinect placed strategically between them. For tactile sensing, we have developed our own pressure-based sensors embedded in the fingers and palms. These self-developed sensors, uniquely designed for irregular surfaces, respond rapidly at frequencies exceeding 250 Hz, providing nuanced tactile feedback. The proprioceptive capabilities are enabled by two calibrated ABB arms and Trx-Hands, which precisely track movements of both arm and finger joints. Additionally, our configuration includes a motion capture system[1] (Figure 14), ensuring highly

---

[1]https://www.vicon.com/software/tracker/

Figure 12. **Well-Aligned Vision and Touch Data Visualization.** The image on the left shows a color-rendered image of the object being held, while the right side depicts the point cloud generated by the depth camera. The gray points represent the depth, while the red points indicate the touch points that have been activated.
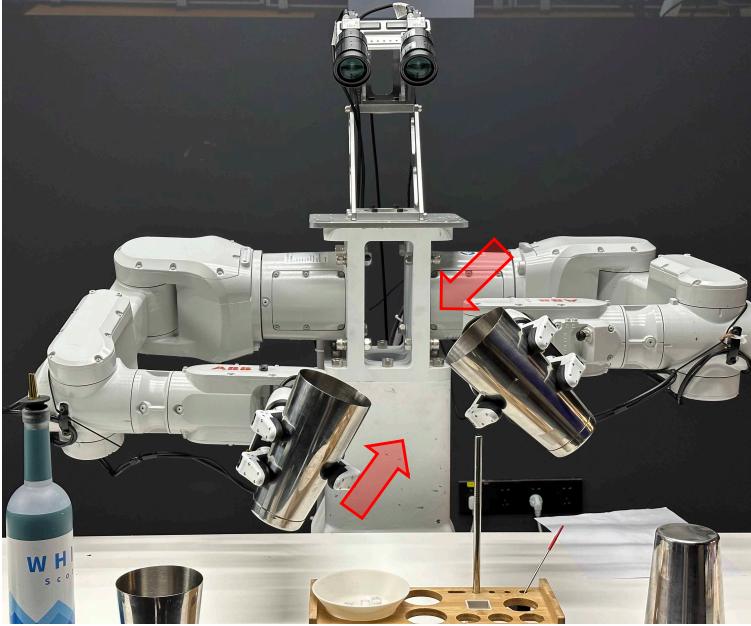

Figure 13. **Robotic Pouring Task.** Demonstrates the "Perception-Planning-Control" paradigm in a robotic pouring task, highlighting challenges in unstructured household environments due to potential vision obstruction by the robot's hand. Essential for success is the integration of tactile feedback from finger and palm sensors and proprioceptive data from arm and hand joints.

accurate tracking of objects and hand poses.

A.2.2. ACQUIRING TOUCH

Acquiring touch point cloud. As highlighted earlier, the Trx-hands are outfitted with pressure-based sensors, as illustrated in Figure 9. Each hand is equipped with sensors that yield 620 values from various points across the knuckles and palm, providing precise contact pressure readings at each touch point. Our bespoke tactile sensors, designed to mimic human fingers' dexterity, can conform to irregular surfaces. Each sensing point on these sensors demonstrates remarkable responsiveness, which is evident through their low activation force, extensive measurement range, and high sampling
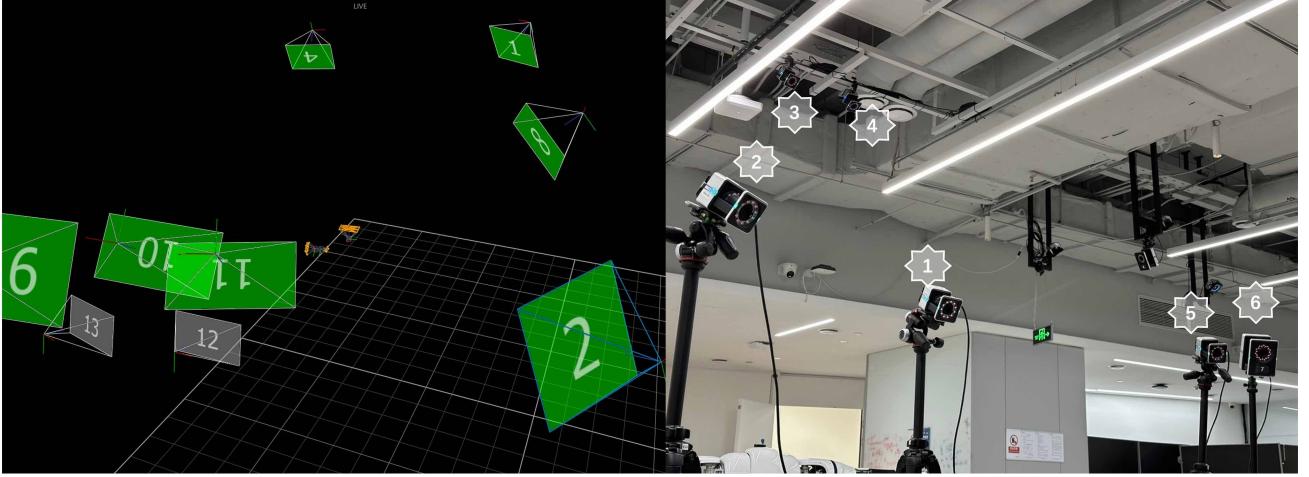
Figure 14. **Motion Capture System.** The motion capture system accurately captures the poses of the object, hand, and robot base.

frequency. During the simulation phase, touch positions are aggregated to form a local touch point cloud. However, translating this into a real-world application presents significant challenges due to inherent system biases.

We initially calculate the hand pose relative to the robot's base frame using forward kinematics to address this. This calculation is extended from the finger knuckles to the hand frame. We then identify the activated touch sites and map them onto the finger knuckles' positions to form the local touch point cloud. Given the complex series of calculations involved in proprioception, achieving the necessary precision in the touch point cloud requires a highly calibrated robotic system. While one common method to minimize hand pose errors involves attaching ArUco tags (Garrido-Jurado et al., 2014), as demonstrated in (Dikhale et al., 2022), (Li et al., 2023), (Rezazadeh et al., 2023), and (Xu et al., 2023), this approach can lead to centimeter-level inaccuracies, which fall short of our precision requirements.

To overcome this, we equipped the Trx-hand with a flange plate featuring markers and utilized a motion capture system to record the hand pose, achieving sub-millimeter accuracy. Furthermore, both the hand and arm were rigorously calibrated. The complete touch points were visualized and meticulously aligned with the vision system, as detailed in Figure 4.

A.2.3. ACQUIRING VISION

Aligning RGB-D images. This process necessitates aligning the depth images with the RGB images. We obtain RGB images from the right camera of the stereo setup, which has a resolution of 1200 x 960, and depth images from the central TOF camera with a resolution of 640 x 565. Using a chessboard, we repeatedly recalibrated both cameras' intrinsic and extrinsic parameters to ensure precise alignment. After distortion correction, we reproject the depth image into the right camera's frame, resulting in two well-aligned vision images with resolutions of 640 x 400,.

Acquiring segmentation labels. When an object is held in hand, it is often obscured by occlusions caused by multiple fingers, tightly intertwining the object and the hand. This complexity poses a significant challenge in segmenting the object from the hand in aligned images. Despite the advancements in large-scale segmentation models, like (Qi et al., 2023b) and (Kirillov et al., 2023), segmenting objects in hand still necessitates specific prompts for guidance. Manual hover-and-click methods for adding and removing prompts in our extensive dataset are impractical. Our pivotal insight involves using touch and proprioception as cues for SAM (Kirillov et al., 2023). With our vision and touch data precisely aligned to millimeter accuracy, we leverage touch data and proprioception for segmentation. Specifically, using the activated touch points and the camera matrix, we calculate the pixel position of each touch point through robotic kinematics. As demonstrated in Figure 16, we assign 'add' prompts to activated touch points on the thumb fingertip and 'remove' prompts to those on the index and middle fingertips. We employed the ViT-H SAM model with 636 million parameters for our experiments.
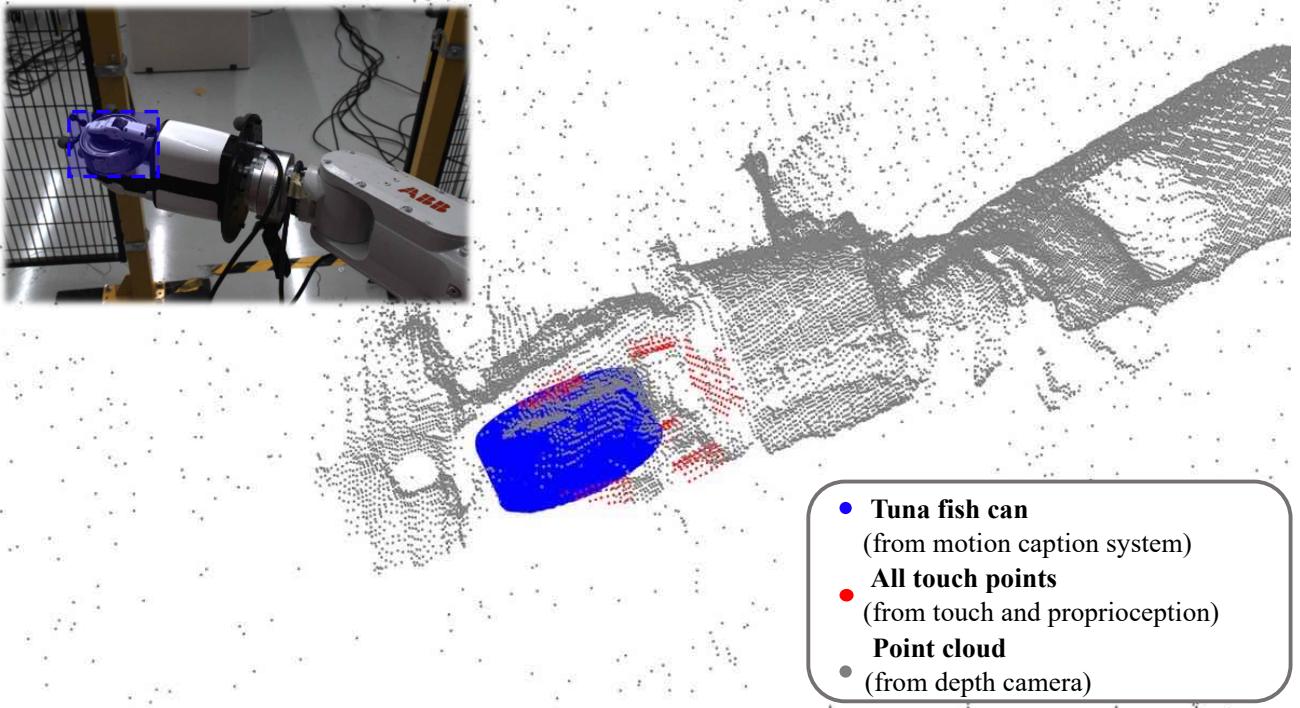
- • **Tuna fish can**
  (from motion caption system)
- • **All touch points**
  (from touch and proprioception)
- • **Point cloud**
  (from depth camera)

Figure 15. **Visualization of full tactile points aligned with vision in VinT-Real.** Gray points represent the point cloud from the depth camera, blue points represent the transformed model from the motion capture system, and red points indicate the full touch points of the hand.

A.2.4. ACQUIRING OBJECT-IN-HAND POSE GOUND TRUTH

Obtaining ground truth for object-in-hand poses is straightforward in simulations but presents significant challenges in the real world. Unlike object pose estimation datasets in computer vision, such as those in (Calli et al., 2017) where objects are simply placed on a table and their poses estimated via QR codes, we adopt a motion capture system similar to (Dikhale et al., 2022), which attaches markers directly to the objects. However, we believe that simply sticking markers on objects may not yield high-quality pose data due to the uncertainty in marker positions relative to the object frame. To address this, we have custom-designed fixtures for each object, affixing markers on these fixtures. We then securely attach these marker-equipped fixtures to the objects, ensuring that the anchoring is both visually and physically feasible when the object is held in hand, as illustrated in Figure 17. All models of these fixtures will be made publicly available on our website.

A.2.5. OBJECT CATEGORIES

In constructing the VinT-6D, we applied the following criteria for object selection: (1) *Held in Hand*: Objects of an appropriate scale that can be comfortably held in hand. (2) *Commonality*: Objects should be commonplace in everyday life and easily accessible. (3) *Variety*: The selection should cover a diverse range of materials, including plastics, metal, and glass. The field of robotic manipulation is increasingly focusing on domestic applications, such as household robots designed for tasks in laundry rooms or kitchens. These innovations are particularly significant for supporting independent living for the elderly, assisted living facilities, and addressing various healthcare challenges. In the VinT-Sim dataset, while we initially chose 21 objects from the Yale-CMU-Berkeley (YCB) dataset (Calli et al., 2017), noted for their handheld size and frequent use in robotics research, we realized that some objects are difficult to find in stores, especially for those researchers not in US. Therefore, they are not enough for us to provide a generic object-in-hand benchmark for robotic perception researchers to design and test their algorithms. To enhance practicability, we incorporated 5 high-quality scanned objects from daily life, featuring transparent and reflective properties, as rendered in Figure 18. For VinT-Real, we selected ten objects, such as a tomato soup can, power drill, and mustard bottle, detailed in Figure 19. These everyday items, selected for their specific functions, require particular grasp poses for tactile interaction. Additionally, objects like a metal shaker and
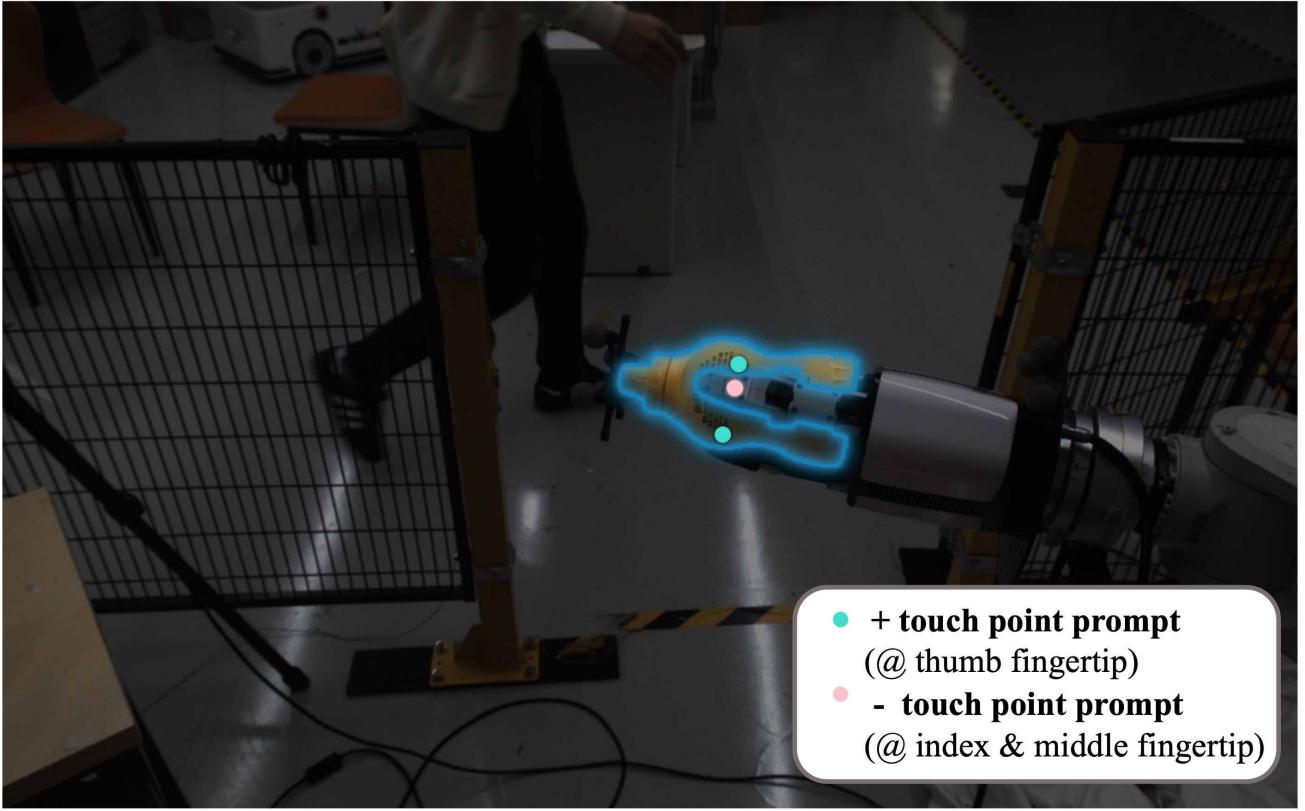
Figure 16. **Object-in-hand segmented by large vision model. We use touch and proprioception as cues for the segmentation of a large model. We assign 'add' prompts to the activated touch points on the tip of the thumb fingertip while assigning 'remove' prompts to those on the index and middle fingertips. This approach helps us to effectively and efficiently segment the model.**
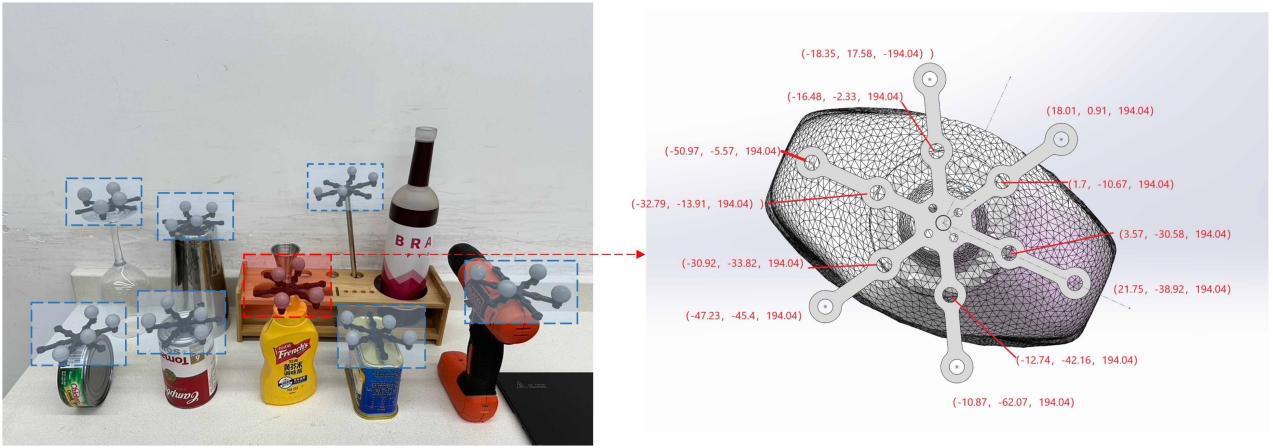

Figure 17. **Real-world objects with markers.**

wine glass vary in size and material distribution, with some, like the stir, presenting a challenge in fingertip touch during rotational holding to prevent dropping.

Figure 18. **Synthetic rendering of 25 object models in VinT-Sim.**

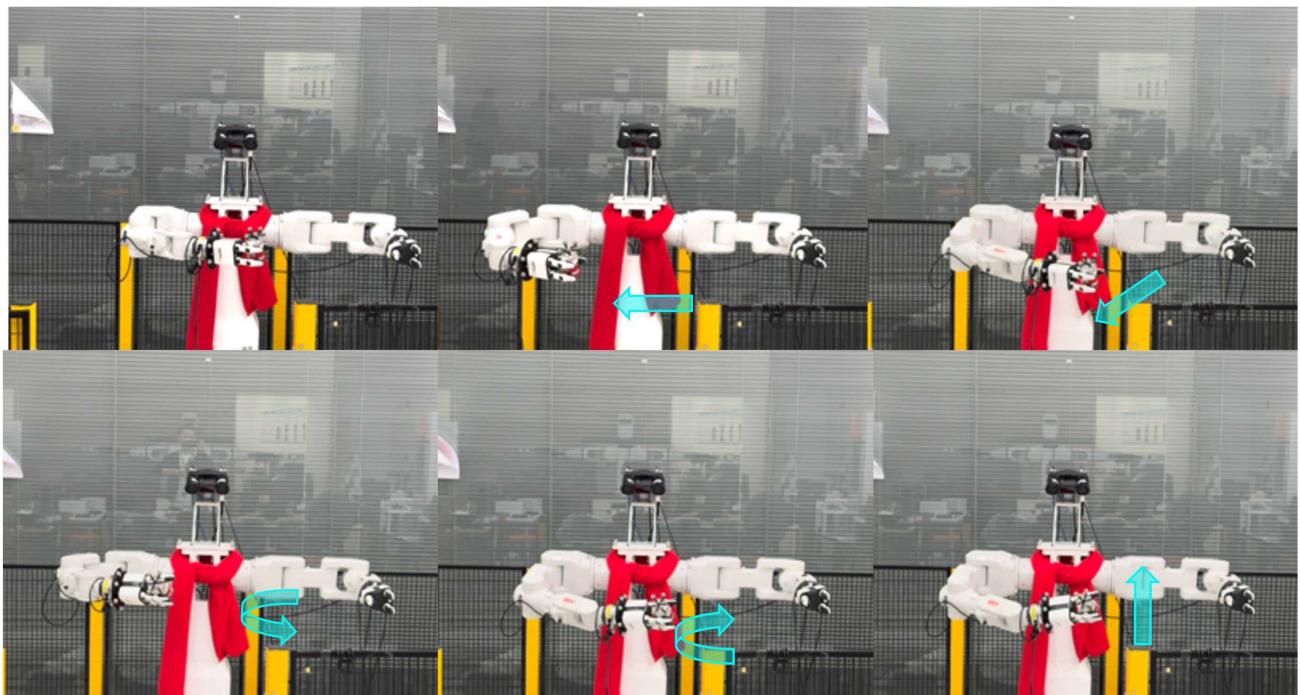*Figure 19.* **Selected 10 objects used in the VinT-Real dataset.**

Figure 20. **Data collection mirroring toddler-like exploration.**