國立陽明交通大學

電控工程研究所

碩士論文

Department of Institute of Electrical and Control Engineering

National Yang Ming Chiao Tung University

Master Thesis

使用端到端學習及強化學習方式進行夾取姿態預測實現人對機器人之交遞行爲

Learning End-to-End 6DoF Grasp Choice of Human-to-Robot Handover using Affordance Prediction and Deep Reinforcement Learning

研 究 生：張博凱 (Chang, Po-Kai)

指導教授：王學誠 (Wang, Hsueh-Cheng)

中華民國一一〇年九月

September 2021

使用端到端學習及強化學習方式進行夾取姿態預測實現人對機器人之交遞行爲

Learning End-to-End 6DoF Grasp Choice of Human-to-Robot Handover using Affordance Prediction and Deep Reinforcement Learning

研 究 生：張博凱　　Student: Po-Kai Chang

指導教授：王學誠 博士　　Advisor: Dr. Hsueh-Cheng Wang

國立陽明交通大學

電控工程研究所

碩士論文

A Thesis

Submitted to Institute of Electrical and Control Engineering

National Yang Ming Chiao Tung University

in Partial Fulfilment of the Requirements

for the Degree of

Master of Science

in

Engineering

September 2021

Hsinchu, Taiwan

中華民國一一〇年九月

使用端到端學習及強化學習方式進行夾取姿態預測實現人對機器人之交遞行爲

學生：張博凱　　指導教授：王學誠 博士

國立陽明交通大學 電控工程研究所

摘　　要

最近發生的 COVID-19 疫情已影響到大多數國家，並推動了全球努力開發機器人來幫助一線醫務人員。人機之間的交遞行爲是服務機器人的一項關鍵能力。最近的研究通常利用現有的手部和物件分割，以及姿勢估計算法來生成抓握。在不考慮人手的情況下，直接根據傳感器數據進行端到端預測的抓取已成功地應用於物流任務中，但是尚不清楚端到端抓取如何實現人類對機器人的交遞行爲。然而，這些方法都旨在抓握正確，不觸碰人類的手，並沒有考慮人類對於這些抓握的感受。我們提出用於人機之間交遞行爲的端到端 6DoF 抓取選擇。我們首先利用現有的端到端抓取網絡主幹，並透過深度強化學習微調抓取姿態。使用各種基線進行綜合評估，並表明，所提出的方法對部分遮擋更具魯棒性，並且使用適合人類的 6DoF 抓取策略，而不透過固定抓取姿態進行分類抓取。用於人機之間交遞行爲的預訓練模型及端到端抓取、軌跡數據集可在 https://arg-nctu.github.io/projects/socially-aware-handover.html 獲得.

Learning End-to-End 6DoF Grasp Choice of Human-to-Robot Handover using Affordance Prediction and Deep Reinforcement Learning

Student: Po-Kai Chang Advisor: Dr. Hsueh-Cheng Wang

Institute of Electrical and Control Engineering
National Yang Ming Chiao Tung University

Abstract

The recent COVID-19 outbreak has affected most countries, and motivated global efforts for developing robots to help the frontline medical staff. Human-to-robot handover is a key capability of service robots and human-robot interaction. Recent work takes advantage of existing hand and object segmentation, and pose estimation algorithms to generate grasps. End-to-end grasping directly from sensor data without object models has made tremendous progress in logistic tasks, but has not been used for human-to-robot handover. However, both approaches aim for grasping without inducing harm, but neither consider which types of grasps may be intrusive to human users. We present our end-to-end 6DoF grasp choice for human-to-robot handover. We first leverage existing end-to-end grasping for the network backbone, and then finetune for preferred grasps using deep reinforcement learning. Comprehensive evaluations are carried out against various baselines using multi-stage hand and object prediction and subsequent planning. We show that the proposed approach was more robust to partial occlusions, and executed human preferred 6DoF grasps without hard-coding the correspondence of hand grasp classification. A dataset of end-to-end grasping and trajectories for human-to-robot handover and all pretrained models are available at `https://arg-nctu.github.io/projects/socially-aware-handover.html`.

Acknowledgement

首先感謝王學誠老師用心栽培，從大學三年級到研究所這段期間在專業領域的教導以及做研究的態度。從實驗室剛開始王老師很努力花心力在建構實驗室的教學上，到後期試著將學界的研發應用到業界，發揮影響力，過程中投資很多心力和金錢在我們身上。目前實驗室正處於快速成長期，即便事情越來越多，王老師繼續堅持著做研究的態度，用心照顧每個學生。特別感謝王老師提供我們很多機會參加眾多國際會議以及比賽，增加我的視野，也更能看到國外頂尖學者做研究的心態以及最頂尖的研究內容。

再來感謝口試委員楊谷洋教授、陳鴻文教授以及柯立偉教授，你們在專業的方面提供了很多建議，爲了讓我的論文更加嚴謹且完整。也感謝所有教導過我的老師，你們無私的提拔讓我有機會完成我的碩士學位。

接下來要感謝全體 ARG 實驗室的學長姐、同學、學弟妹還有專題生。感謝一路走來幫過我的同學，從我專題生碰起 Duckietown 到後來 SubT 國際競賽，大家一起從無到有走上國際舞台，一起經歷各種喜怒哀樂，感謝有你們在我的研究生涯中佔了很重要的一部分，期待大家未來都能夠到自己適合的領域並有很好的發展。

最後要感謝我爸媽在我就學的過程中給我很多建議以及鼓勵，指引我人生的方向，同時也讓我自由的發展。謝謝你們讓我無後顧之憂地繼續往前努力、衝刺。

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Although advances in medicine have enabled people to live longer, global work and travel have facilitated the spread of infectious diseases, including SARS-Cov-2. More than 212 million people have been infected worldwide, and more than 4.44 million have died from COVID-19 as of August 24, 2021. Robotics researchers have played an important role in managing the pandemic [6, 7], and substantial effort has been made to adapt preexisting devices to the new challenges brought about by the pandemic. Numerous useful solutions have been developed. In particular, the lack of time for designing and developing medical robots in response to the pandemic has prompted experts to consider methods of responding to the infectious disease crisis by analyzing the challenges and opportunities for advancements in these technologies [8].

Hospital cluster infections are more dangerous than community cluster infections are. Therefore, researchers in various communities have rapidly mobilized to reduce hospital infection risk, and service robotics is a pioneering method in this field [6]. Research reports [9] have revealed that among the top 10 industrial projects funded by the Digital Innovation Center for Medical Robotics (DIH-HERO) project in Europe, robots providing disinfection and logistical services have the greatest short-term impact. Remote-controlled robots operated by humans can provide services to patients. For example, nurses can use these robots and their mobile manipulators for medicine delivery and pairing or for the disinfection of patient areas. Autonomous robots are critical for maintaining separation between medical staff and isolated patients. Wheeled mobile robots can be used to disinfect contaminated places in hospitals without the need for any contact between the patient and medical staff [10, 11, 12].

Many non-care core hospital tasks such as delivering medicine, collecting bedsheets, and disinfecting the environment can be performed by autonomously moving Robotic Nursing As-

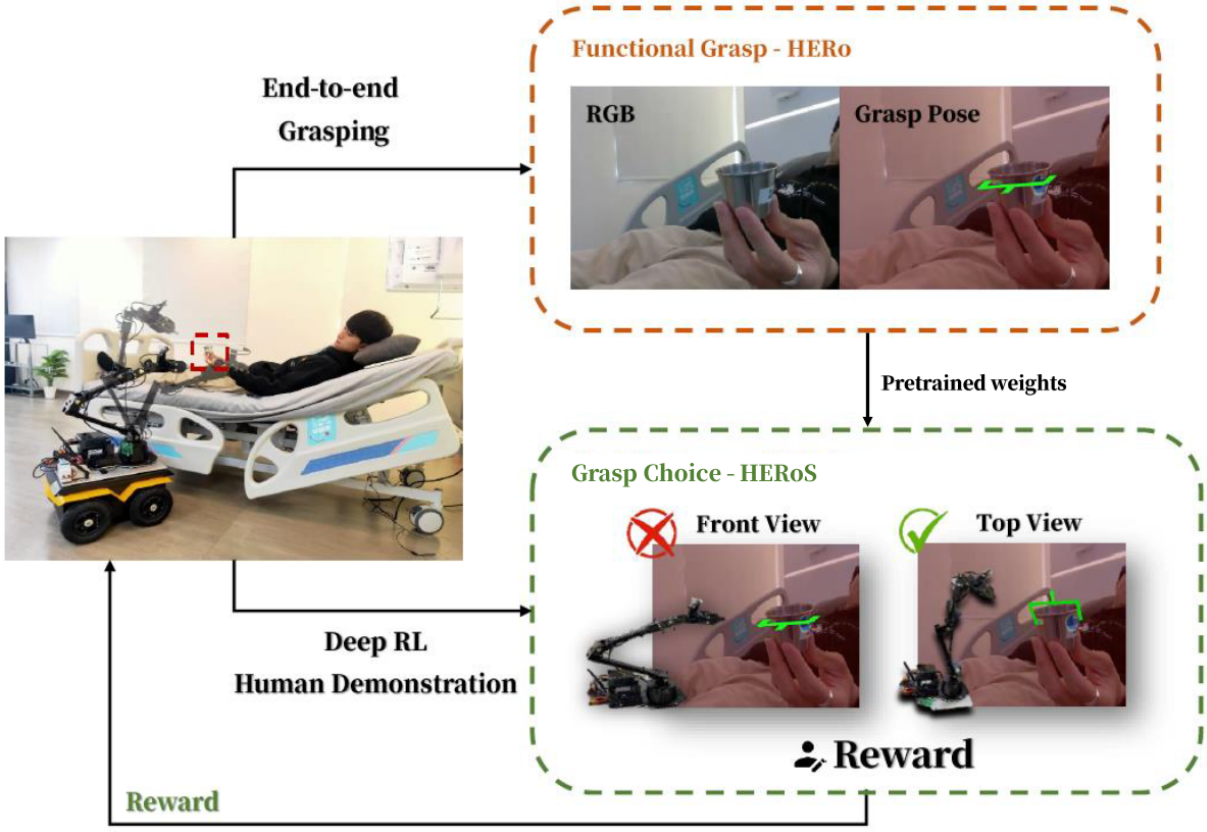

Figure 1.1: Our approach allows human-to-robot handovers using end-to-end grasp choice, learning-based algorithms. When the program is executed, the agent will generate an action. Through the end-to-end approach we proposed, there is a 93.5% probability that the target object can be grasped. In addition, we also incorporated reinforcement learning rewards, so that the agent can learn human grasping preferences to make the robot more anthropomorphic.

sistants (RoNAs)[13]. The use of robots instead of medical personnel to enter isolation area to perform tasks, in addition to reducing unnecessary contact with patients, can also reduce the risk of infection caused by contact with objects that patients have used.

Thus, in this study, our goal was to formulate an efficient and customizable method for meeting medical needs by enabling a robot to work with a human (human-to-robot handover) and to apply this method to a hospital medicine delivery robot with a manipulator. A RGB-D camera was used to accurately predict the gripper position between the object and the human fingers, and the manipulator then avoided the fingers when attempting to grasp a target object, such as a medicine cup, medicine bottle, or medicine box.

1.1.1 Human-to-Robot Handover

Human-robot collaboration in handover is mainly divided into robot-to-human (R2H) handovers and human-to-robot (H2R) handovers. In previous researches, most of them focused on robot-to-human (R2H) handover. Human can actively track the position of the robot arm and take the object. However, human-to-robot (H2R) handovers have tried to address challenges in enabling seamless autonomous handovers, from human and object perception to motion control during handovers. In the most recent research, only two papers [14, 5] had more than ten objects to overcome the diversity and uncertainty of the environment. Therefore, the current research and development aims to allow humans to achieve human-robot collaboration in the most natural and safe way.

Human-to-robot handovers, involving the transfer of an object from a human giver to a robot receiver, are fundamental capabilities that allow for service robots in our daily lives. There has been increasing interest in studying human experiences [15, 16], predicting human hand poses or object affordance (represents how the object is grasped to complete the handover task) for grasping selections [17, 14], and expanding the types of handovers for a wider range of objects [14, 5] in order to improve human-to-robot handover.

Grasp choice may be influenced by several constraints, such as object shape, task requirements, gripper types, and social convention [16]. By observing and analyzing the human grasp, the robot grasp could be adapted accordingly. Therefore, previous works tend to utilize human grasp classification [17] or hand pose estimation, in order to generate a corresponding grasp pose.

Several deep learning-based techniques have been applied in handover pipelines, including object detection [18], and object pose estimation [19].In general, human-to-robot handovers are successful as long as the objects and hands are accurately tracked.

1.1.2 End-to-End Grasping

Recent advances in end-to-end grasping prediction have shown promising progress. [20] Used a ResNet-101 backbone end-to-end network to generate the affordance map for either

suction or two-finger parallel grippers in heavily cluttered scenarios during the Amazon Robotics Competition. The series of DexNet models [21, 22, 23, 24] used large amounts of simulation-only datasets with thousands of objects to train the end-to-end CNN networks. Although the above research was successful in overcoming occlusions in cluttered scenarios, this has not yet been used to tackle input images involving objects that are held in human hands, nor has their been consideration of what grasp may be intrusive for users during human-to-robot handovers.

1.2 Challenges

Figure 1.2: Context comparison between logistics grasp and handover grasp.

Compared with other existing end-to-end grasping networks [25, 4, 20, 24], human-to-robot handovers requires the following adaptations.

Background: In previous works, the objects are placed in tote or on tables and therefore the backgrounds are often relatively plain. It has been shown that background may largely affect deep network performance, such as replacing the background by a checkerboard [26]. [5] modified the grasp prediction network GG-CNN [4] by adding a planar surface background. (Fig. 1.3)

Occlusion: Given that the target object is held by a human hand, there are a certain number of occlusions that may affect the success rate of segmentation and pose estimation of the object and the hand. In prior end-to-end grasping work, grasp selection networks are usually programmed to pick the objects from the top, and thus are not occluded by other objects in cluttered scenes. Each object is then removed from the tote one by one. In this context, the grasping behavior

only has to deal with background clutter, but not occlusion challenges. In contrast, depending on how the object is held in the human's hand, occlusion may be higher and thus existing segmentation and pose estimation algorithms may fail. Therefore, although there are only two objects (a single target object and the human hand) in the input images, the occlusion challenge remains. (Fig. 1.4)

Planar vs. 6 DoF Grasping: In prior research, end-to-end grasping is usually formulated as planar grasping [27], whereby a grasping point (x, y) and an angle θ of a two-finger gripper are generated from an input image. However, human-to-robot handovers involve grasping in 3D-space, meaning that there are several grasping and trajectory solutions.

Grasp Choice: Human grasp types are known to associate with different robot grasp during handover [16]. Previous work [17] needs to classify human grasp so that the *hard-coded* robot grasp could be carried out. Learning for such associations in a data-driven manner is not yet studied in previous end-to-end grasping work.

1.3 Contributions

In this thesis, we propose to address the problem of grasp poses in human-to-robot handovers through an end-to-end network. The contributions of our work are as follows:

- **An end-to-end grasping approach of affordance prediction for human-to-robot handovers.** Building upon the work in [20], we collected and manually labelled a handover dataset, and trained a model taking RGB-D images as inputs and generating an affordance map for grasping. In contrast to existing methods, this approach bypasses the separate stages of predicting the pose of the hand and the object. This is especially useful when more than 40% of the target object is occluded, and we systematically analyzed the grasping success rate across several baseline methods.

- **A data-driven approach for learning grasp choice without hand pose nor hard-coded human-robot grasp associations.** Given that a grasp choice is affected by object, task, gripper, or social convention constraints, we learn the implicit associations directly from image inputs. We trained grasping behaviors based on human feedback as rewards through

deep RL.

- **An open dataset for human-to-robot handover.** The data and pretrained weights of the above end-to-end grasping follow widely used end-to-end grasping in [28]. We also provide the grasp choice of human annotations for deep RL. Both are available open access for reproducing this study.

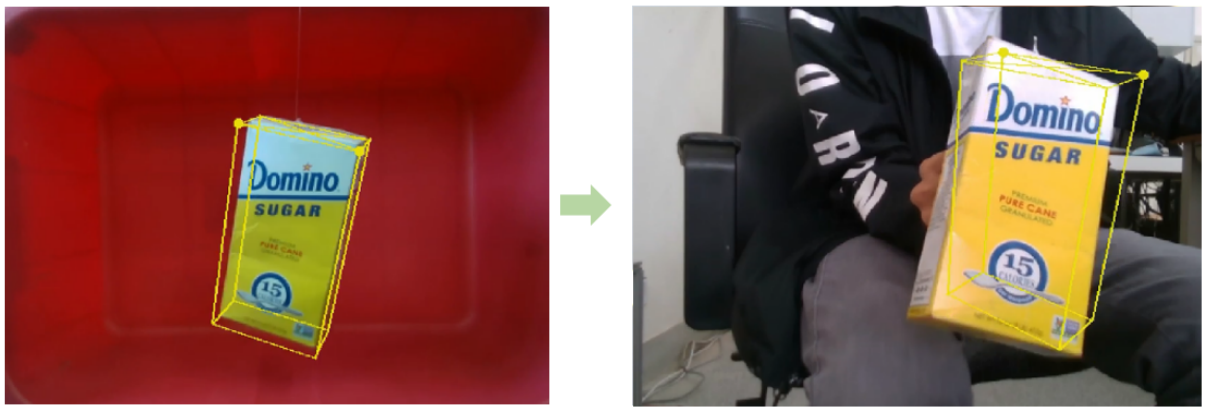

Figure 1.3: Background will affect predict results. (Predict by DOPE.)

Figure 1.4: Handover occlusion will affect predict results. (The occlusion threshold is 44.5%.)

Chapter 2

Related Work

2.1 Hand and Object Pose Estimation

Recent state-of-the-art hand and object pose estimation methods have been developed using a variety of real or simulated datasets. MANO (hand Model with Articulated and Non-rigid defOrmations) [29] was trained on 1,000 high-resolution 3D scans of hands. DOPE (Deep Object Pose Estimation) [30] took advantage of synthetic data for training deep neural networks for YCB household objects. This demonstrated that the combined domain randomized and photorealistic synthetic data closed the reality gap in the context of 6-DoF pose estimation of known objects from a single RGB image. Moreover, ObMan [2] leveraged a large-scale synthetic dataset for joint reconstructions of hands and objects. With manipulation constraints, a model exploited plausible hand-object constellations. We will use these prior methods as our baselines, and systematically evaluate scenarios that include objects with high rates of occlusions.

2.2 Handovers by Estimating Hand and Object Poses

Prior research on human-to-robot handovers have used establishing hand and object detection and pose estimation algorithms. Yang et al. [17] trained a deep neural network using PointNet++ [19] to classify point clouds around human hands into one of seven pre-defined grasping categories. A subsequent motion plan was then carried out to complete the handovers. Consequently, [14] further tackled the challenges of unseen objects. Closed-loop designs refined the tracking of segmented hands and objects over time. A grasp selection model based on the 6-DoF GraspNet [31] was then performed. Similarly, Rosenberger and colleagues [5] proposed a method for grasping generic objects, using an YOLO V3 object detector [18] trained on 80 object categories from the COCO [32] dataset. They simultaneously predicted hand and body segmentation, which were then excluded from a modified GG-CNN [4] model to generate

safe grasps. Our work is different from this prior work, incorporating a data-driven end-to-end grasping model that does not require hand and object detection and pose estimation algorithms.

2.3 End-to-end Grasping

End-to-end grasping has attracted much attention. [27] used a real robot to collect a dataset of 50k grasps in a self-supervised fashion, and trained a deep neural network classifier to predict grasp success. [33] further collected a grasping dataset with more variability by using images from individual's homes instead of only in lab settings using a low cost robotic platform. DexNet [21, 22, 23, 24] is a series of end-to-end approaches that used a 6.7 million dataset which was generated entirely through simulation. The proposed GQ-CNN method evaluated the quality of each grasp configuration from the previous step [0,1], and outputs the highest quality grasp configuration. In contrast,[20] used human annotations for suction grasping and parallel grasping. They used a ResNet-101 backbone network to implement an end-to-end affordance prediction method, without any pre-processing of object segmentation and classification.

Among these methods, self-supervised learning may not be suitable for handovers which include humans. Although simulation offers an abundance of data, it has been known to fall short in certain application setups due to the domain gap between synthetic and real data. How to incorporate human knowledge into the grasping algorithms remains an open question. In our work, the affordance prediction method [20] for human-to-robot handovers was adapted using human annotations.

Chapter 3

Approach

3.1 Problem Formulation

3.1.1 End-to-end Grasping using Affordance Prediction

We follow the affordance prediction [20] formulated as planar grasping. During pre-processing we filter backgrounds that have a depth greater than 75 cm. Given RGB-D images (I_{RGB} and depth I_{Depth}) of the scene, a fully convolutional network is trained to infer the affordances ($I_{Affordance}$) across a dense pixel-wise sampling of end-effector orientations and locations. Each pixel correlates to a different position by which to execute the grasping. By rotating the same input and running inference for N times, where N was set to 8 for 0, 30, 45, 60, 90, 120, 135, and 150 degrees, the largest area among the $224 \times 224 \times N$ affordance map is selected as the grasp by a two-finger parallel gripper. We also map the grasp pose (${}^{O}X^{G_{Grasp}}$) from the camera frame to a world frame (${}^{W}P^{P_i} = {}^{W}X^{C}\,{}^{C}P^{P_i}$). We then use MoveIt[34], open-loop motion planning in real-time, to execute the grasping.

3.1.2 Learning Grasp Choice using Deep RL

Through pixel-wise affordance prediction, we can predict the planar graspable positions and gripper angles. However, grasp choice may be influenced by object or human givers' different preferences when interacting with the robot. We modify the training process to include the human in the loop through deep reinforcement learning.

We formulate our task as a standard Markov Decision Process (MDP), which is defined as $M = \{S, A, R, P, \gamma\}$. We reuse the pre-trained weight obtained in our affordance prediction method, which share the RGB and depth image inputs as the state S. The action $A \in R^{224\times224\times N\times M}$ is a discrete space that maps to the 224×224 grasping position, N as different

orientations of the end effector, and M as the number of 6DoF grasping. R is the reward signal, and P is the transition function to the next state, $p(s_{t+1}|s_t, a_t)$, finally the γ is a discount factor for future value estimation. The objective is to find a state-action value estimator using rewards as a guide. Here we transform our value estimator into $N \times M$ affordance maps, and the largest area is selected for the 6DoF grasping task.

3.2 Pixel-Wise Affordance Prediction

3.2.1 End-to-end Adaptation Challenges

Table 3.1: Summary of Grasping Success Rates for the Experiments by Fine-Tuning the Method

Method/Object	Water Cup	Small Cup	Medicine Bottle	Medicine Box	Success Rate	Δ	Plan Failed	Touch Fingers
Base Grasping	75%	80%	80%	75%	77.5%		22.5%	0%
Hand Grasping	30%	5%	25%	5%	16.25%	-61.25%	46.25%	37.5%
Improve Background	55%	50%	55%	55%	53.75%	-23.75%	26.25%	20%
Improve Annotation	65%	65%	75%	65%	67.5%	-10%	21.25%	11.25%
HERoS	**95%**	**95%**	**85%**	**100%**	**93.75%**	**+16.25%**	**6.25%**	**0%**

To our knowledge, the end-to-end concept, which performed excellently in logistical grasping, has not been used for handovers. We thus fine-tuned the end-to-end concept to adapt it to handovers, as illustrated in Fig. 3.1 and Table 3.1. The difference between the two scenarios was that in the logistics grasping scenario, the background was smooth and consistent and there was no need to avoid human hands. In the handover scenario, only one object was grasped at a time, and objects would thus not be stacked. However, human fingers may occlude the object.

Base Grasping

We referred to the MIT–Princeton [20] end-to-end architecture that ranked first in 2017 Amazon Robotics Challenge. In accordance with their method, we used a dataset of four medical objects (Section. 3.2.2), annotated the positive grasp points as green, and annotated the negative grasp points (the part of the object containing human hands) as red to train the end-to-end model. Due to the configuration of the manipulator, we changed the target angle from horizontal to vertical. We grasped one object in each trial and grasped each object over 20 trials. The success

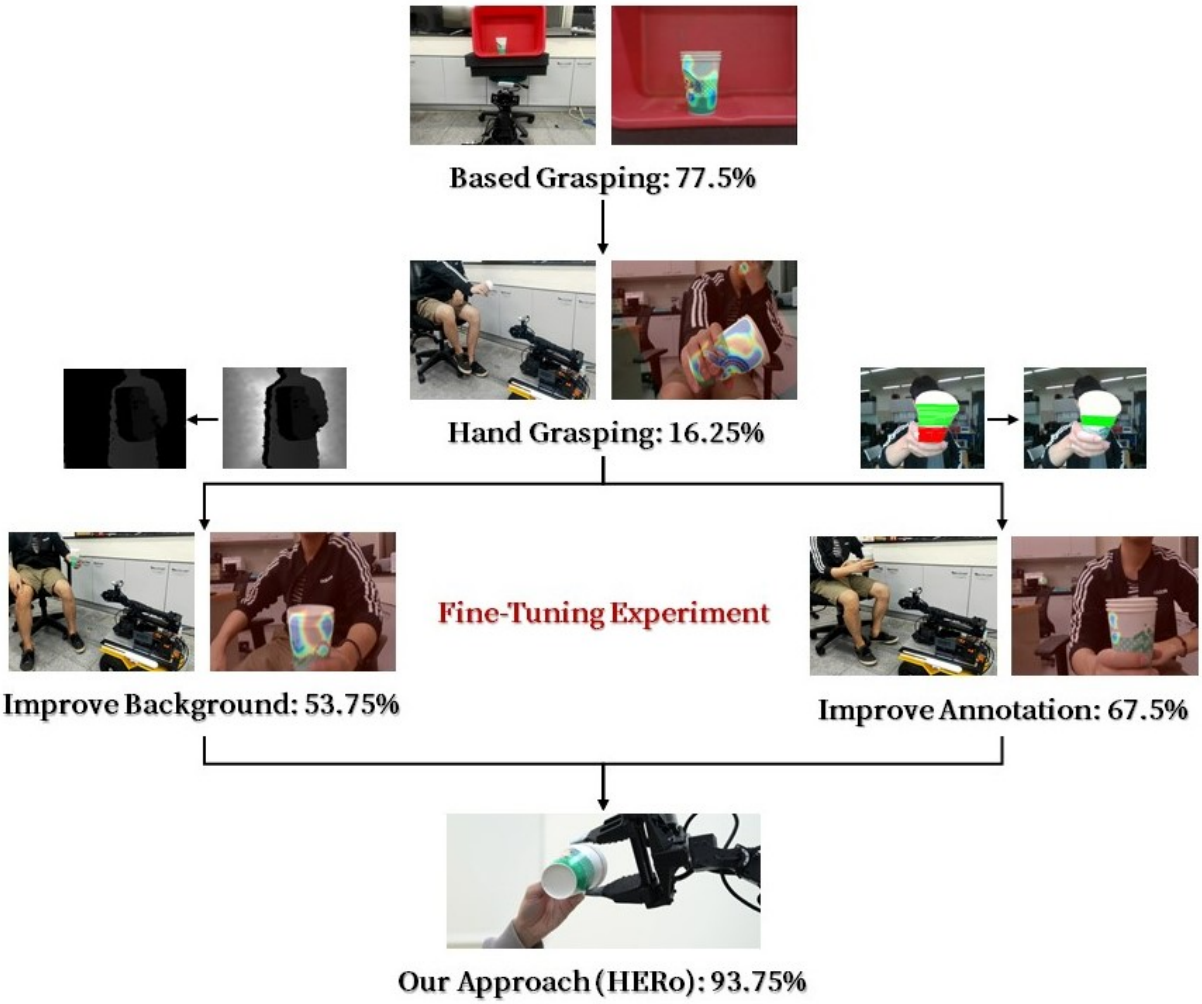

Figure 3.1: Success rate of end-to-end model in handovers. Objects were first grasped in a box. Due to the consistent background and the lack of human hands, the success rate was 77.5%. Next, the object was held in a human hand and handover was performed; the success rate was reduced to 16.25%. After depth processing, the success rate increased to 53.75%. In addition to depth processing, we tried improving only the annotation method; the success rate increased to 67.5%. Finally, we propose an approach combining processing depth and the annotation method; the success rate increased to 93.75%.

rate was 77.5%. The primary reason for failures was that the manipulator path through the Moveit path planning was unusual, causing collision with objects during movement.

Hand Grasping

After demonstrating that effective target grasping in the logistical task, we attempted handovers. Objects previously placed in the box were held by human hands, and the experiment was performed. The success rate decreased from 77.5% to 16.25%. During the experiment, problems such as collisions with human fingers and depth errors frequently occurred, resulting in a failure to grasp handheld objects.

Improve Background Grasping

The experimental results revealed that, in both scenarios, background depth values presented the greatest challenge. Depth information is simple (i.e., flat) for the target in the box but complicated for targets held in human hands, causing noise that interferes with the prediction results. Therefore, we filtered depths greater than 75 cm (the longest reach of the manipulator) in the depth image to simulate the box environment and filtered the noise to clarify the target feature. After a total of 80 grasping trials, the success rate increased from 16.25% to 53.75%.

Improve Annotation Grasping

In the experiment, we observed that the annotated part of the object containing the hand as a negative grasp point (red) interfered with model learning, and the two layers of the network output affected each other. Experiments have demonstrated that, using this annotation method, the model cannot effectively learn that the human hands cannot be grasped; this result had not been previously verified for logistical grasping. Therefore, we modified the annotation method to only label grasp points. The background and human hands were not labeled and were regarded as ungraspable points. The success rate for these experiments increased from 16.25% to 67.5%.

HERoS Grasping (Current approach)

We combined the annotation method and background depth processing to propose an approach (Section. 3.2.3) that is more adapted to handover tasks and with a success rate increased from 16.25% to 93.75%. The approach was applied to interact with humans in a medical setting.

3.2.2 Pixel-Wise Classification Dataset

Figure 3.2: Handover Dataset: (1) water cup, (2) small cup, (3) medicine bottle, (4) medicine box, (5) SPAM, (6) banana, (7) lemon, (8) strawberry, (9) peach, (10) pear, (11) plum, (12) mustard, (13) sugar. Fruit objects were plastic.

To overcome the lack of sufficient training data for the robot to operate in the ward, we collected a dataset covering four representative objects —a water cup, small medicine cup, medicine box, and medicine bottle (Fig. 3.3) These objects could represent the delivery of objects of different sizes and shapes in a ward. We also collected 9 additional YCB [35] objects, for total of 13 objects 1368 RGB-D images, for experiment to compare, as shown in Fig. 3.2.

Collection

We used a D435 RGBD camera recording dataset mounted at the end effector of a ViperX300s on a Jackal UGV. Data was collected in different scenes and with different light source angles, and objects were held in various postures with both left and right hands. During the recording period, humans only took the objects at an angle of 90° horizontal to the ground. This was to cause the predicted result of the object to only have a grasp area at 90° and to further obtain the rotation angle of the object in the horizontal plane.

Label

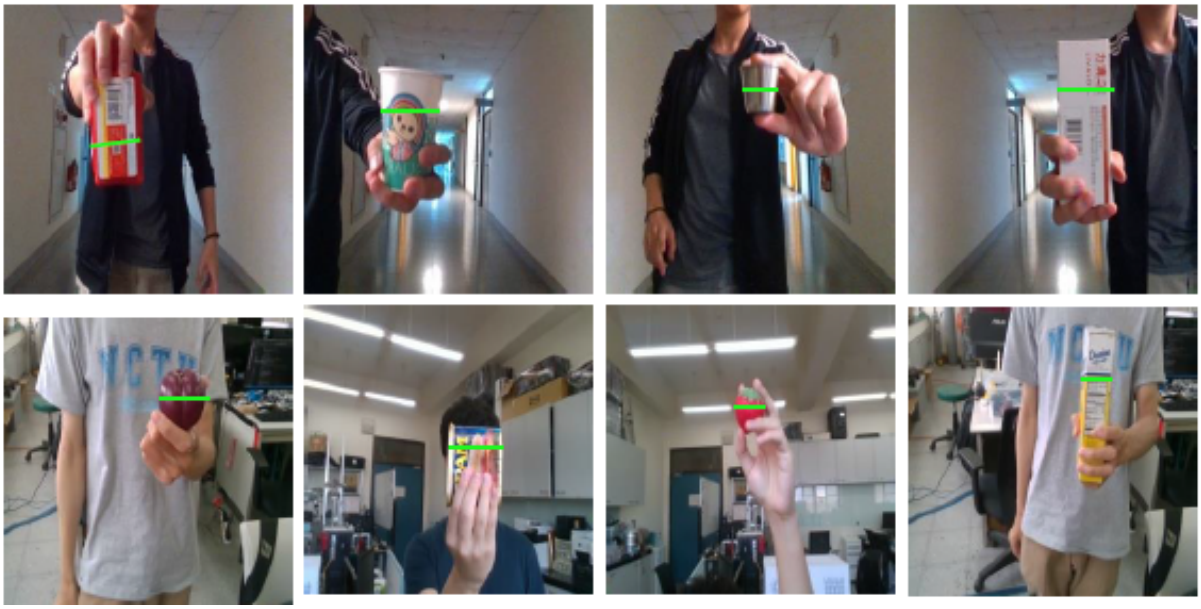

Figure 3.3: Handover Dataset: line annotation example.

We labelled the RGB images using the LabelMe[36] tool. The output size is also a 640×480 densely labeled pixel-wise map. Each pixel value of the pixel-wise map was normalized to between 0 and 1 in the form of a heat map. We followed the dataset in [20] and asked human annotators to label a line of where a two-finger gripper could grasp the object without touching the human hand. Similar to the labelling in [20], one object was densely labelled with the graspable positions and angles displayed in green, shown in Fig. 3.3. Each object included around 100 images, resulting in 1,368 RGB-D images; there were 1,368 annotations with 18,868 possible grasps (green lines).

3.2.3 Affordance Prediction

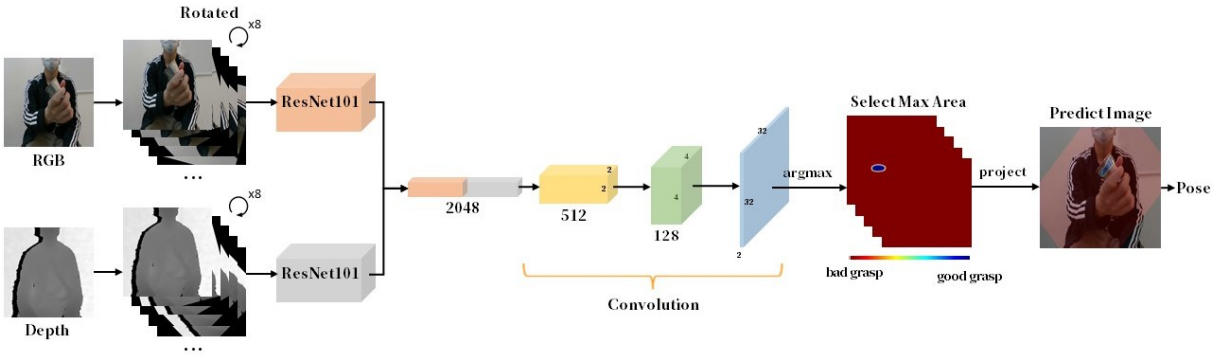

Figure 3.4: An overview of the HERoS framework. We used ResNet-101[1] as the main network architecture of HERoS. Given RGB-D images rotated eight times as input, the system predicted eight pixel-wise affordance maps. Then, the largest area of the affordance maps was selected as the rotation angle of the gripper (orientation). Finally, the affordance was projected to the original RGB image and combined with depth information for the grasp pose to be obtained.

To accomplish the handover task, we obtained the 3D position of the grasp point. The grasp point was defined as the point at which the object could be grasped stably without coming into contact with the human fingers. By using the labeled dataset, we allowed the model learn whether each pixel of the RGB-D image was a point that could be grasped.

In this thesis, we used ResNet-101[1] as the main network architecture of HERoS and used residual networks to solve the degradation of deep networks and reduce the required computation (Fig. 3.4). Although this dataset was small with respect to what is required to train a deep network, the use of ResNets pretrained on ImageNet[37] was sufficient to fine-tune our architecture. The architecture was a dual-stream network. The RGB images (RGB, 3 channels) and cloned depth images were normalized by subtracting the mean and dividing by the standard deviation (DDD, 3 channels) and were input to the ResNet-101 network separately. The depth was cloned across channels to use ResNet's pretrained weight ImageNet on 3-channel (RGB) color images to avoid convergence difficulties in the case of small datasets. Finally, the two ResNet-101 outputs (RGB and depth) were concatenated, and three additional spatial convolution layers were used to merge the features. Then, the images were spatially up-sampled bilinearly and softmaxed to output two pixel-wise layers (non-grasping layer and graspable layer) to represent the inferred affordances.

In this case, we used a D435 depth camera to capture 640×480 RGB-D images as input;

the output size was also a 640×480 densely labeled pixel-wise map. Each pixel value of the pixel-wise map was between 0 and 1, and it was displayed as a heat map.

We used cross-entropy as a loss function, a batch size of 10, a fixed learning rate of 10^{-3}, and a momentum of 0.99 to train HERoS through stochastic gradient descent of momentum. Our model was trained in PyTorch using an NVIDIA RTX 2080 GPU and an Intel Core i5-9400F CPU. Training with our dataset took approximately 1 hour.

For application, we used a trained HERoS network to predict the RGB-D image, generate an affordance map, and project the RGB-D depth information to obtain the real-world 3D grasp point position. For the orientation, because the training data included only objects perpendicular to the horizontal ground (Section. 3.2.2), if the object was in a skewed state during the prediction process, the grasp region of the affordances map was reduced. The same frame was rotated eight times (0°, 30°, 45°, 60°, 90°, 120°, 135°, 150°) and the angle with the largest affordance map area was selected to obtain the orientation of the object in a grasp plane.

3.3 Human-to-Robot Handover Task

Our handover system was divided into three states, **on standby**, **plan grasp pose**, and **execute manipulator**. In this section, we introduce the operation process of the entire task, how to transmit messages through the robot operating system (ROS), and how to process and execute messages after receiving them.

3.3.1 On Standby

In this state, the manipulator does not perform any actions and is stationary on the UGV. The vehicle is free in this state. The UGV may be a shuttle carrying objects between the wards and the pharmacy. If the handover task is triggered through the ros service, the manipulator enters the **plan grasp pose** state.

3.3.2 Plan Grasp Pose

The RGB-D camera running at 15 Hz mounted at the end effector captures images (I_{RGB}, I_{Depth}) used by all of the perception modules; the modules run concurrently in real-time. In this module, the best grasp pose is obtained through a series of processes. The best grasp pose is that which can accurately grasp objects and avoid any contact with human fingers. After this state is entered, HERoS inference is activated by receiving RGB-D messages from the D435 depth camera. The images I_{RGB} and I_{Depth} are input to the HERoS network and are used to predict the pixel-wise affordance map ($I_{Affordance}$), to indicate the grasp value that is graspable for each pixel, and to obtain the orientation by rotating the input eight times (because the largest grasp area made by the prediction result is obtained only when the object is at a 90° angle to the horizontal ground). Then, the affordance map with the largest grasp area is mapped to the original RGB image by rotating the obtained angle of the object. The center of the largest area in the $I_{Affordance}$ is selected and combined with I_{Depth} to obtain the grasp pose. The grasp pose is obtained through the perspective of the camera. To drive the manipulator, the grasp pose must be mapped to the manipulator coordinate system through a transform conversion (${}^{W}P^{P_i} = {}^{W}X^{C}\,{}^{C}P^{P_i}$) to drive the manipulator directly to the grasp pose (${}^{O}X^{G_{Grasp}}$). The aforementioned process is a single-frame image processing procedure. To avoid interference from human hand shaking and the effects of image noise, the final target pose (${}^{T}X^{G_{Grasp}}$) is obtained using five consecutive frames; the position distance between each frame is less than 5 cm. Finally, the mean in x, y, and z of the five frames is selected for the handover pose. Once chosen, the target pose is unchanged until **execute manipulator** is completed.

$$
{}^{T}X^{G_{Grasp}} = \begin{cases} \overline{\sum_{t=0}^{4} {}^{O}X^{G_{Grasp}}(t)} & if \quad \text{distance between} \\ & \text{frames} \leq 5cm \\ 0 & else \end{cases}
$$

3.3.3 Execute Manipulator

This module translates the output of **predict grasp pose** into joint and gripper actions by using the Moveit[34] path planning node. This process is mainly based on the best grasp pose generated by HERoS. When the manipulator VX300s reaches 5 cm in front of the target grasp point, the velocity of the manipulator becomes $\frac{1}{5}$ times the normal speed, and the manipulator slowly approaches the target point. This reduction in speed protects humans during the handover and ensures that the patient does not change the position of the original object or change the grasp pose due to fear of being touched by the manipulator. After reaching the target point, the gripper is closed and the robot detects whether the object has been grasped correctly by evaluating the gripper position. If the robot determines that grasping has failed, the manipulator returns to the ready pose and **predict grasp pose** is repeated. If the object is held tightly, the manipulator returns to the home pose by changing to the **on standby** state.

3.4 Learning Grasp Choice using Deep RL

By using pixel-wise affordance prediction, we could predict the graspable area and could use rule-based methods to grasp the object. However, people have different preferences when interacting with a robot. We wanted our robot grasping to be optimized for users and to be comfortable to interact with. Thus, we modified the training process to include humans by using a deep reinforcement learning (DRL) formulation. However, no simulator exists for the human–robot hand-over task to bootstrap the DRL training. Additionally, directly training the DRL agent from scratch in the real world to interact with humans is inefficient and could cause physical injury to human testers. Therefore, we used a collected human demonstration replay buffer and an offline reinforcement learning algorithm to start training another affordance prediction model.

3.4.1 Human Demonstration Replay Buffer

We collected an offline training replay buffer containing medical items and 9 additional YCB [35] objects for a total of 13 objects and 1127 transitions. Transitions can be represented

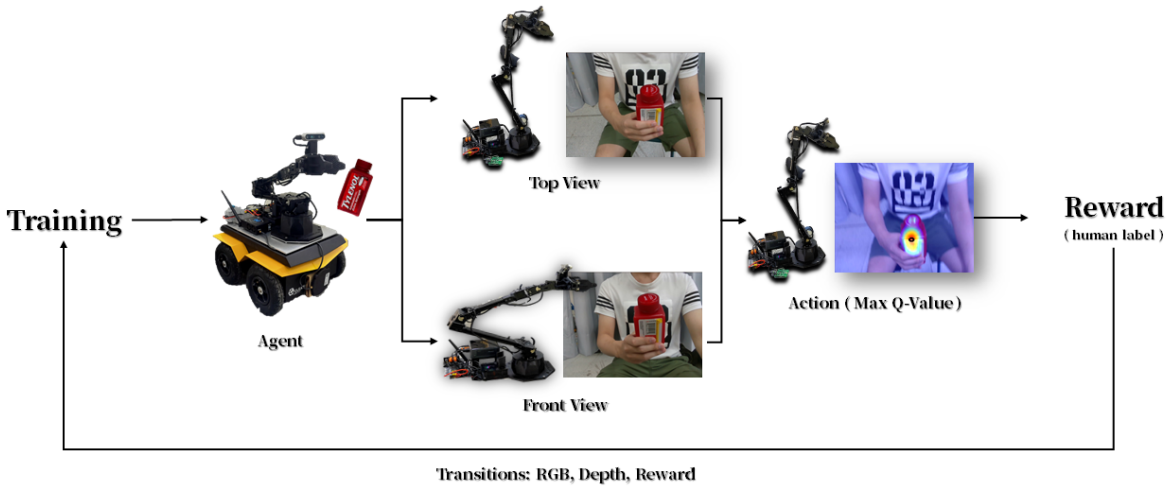

Figure 3.5: Grasp choice system architecture diagram. When the image is received, it will control the manipulator from two different perspectives for prediction, selecting the perspective with a larger Q-value to perform the grasping, and receives the reward through the human label after the grasping is completed, so that the agent can learn human preferences.

as a tuple $\{S_t, S_{t+1}, a_t, r, E\}$ that indicate, from left to right, the initial state, end state (each state features a pair comprising an RGB image and a depth image), action (position and orientation), reward, and a flag indicating success or failure.

Collection

We used a D435 RGBD camera to collect RGB and depth images for different scenes and light source angles. Objects were held in various postures and in either the left or right hand. We held the objects at four angles (90°, 0°, −45°, 45°) to the horizontal ground, and the model predicted an action based on these data. If transition was successful, each state was a pair of images: one containing the held object and the other not containing the object. If transition fails, the pair of images in the state both contain the held object.

Label

To generate the gripper action, we labeled a graspable line in the RGB images by first using the Labelme[36] tool and subsequently marking the center position and orientation of the line as

the grasping position and angle, respectively. We also labeled a True or False flag and a reward of 5 or -5 depending on whether the transition was or was not a successful grasp.

3.4.2 Deep RL Value Prediction

We formulated the task as a standard Markov decision process (MDP) defined as $M = \{S, A, R, P, \gamma\}$. In our case, the state S was the RGB and depth image taken by the robot before the start of grasping, the action $A \in R^{224\times224\times4}$ was a discrete space mapping to the 224×224 grasping position and 4 different orientations of $\{-45°, 0°, 45°, 90°\}$ for the end effector, the reward signal R was 5 for a success grasp and -5 if otherwise, P was the transition function to the next state $p(s_{t+1}|s_t, a_t)$, and, finally, γ was a discount factor for future value estimation.

Basically, we need heatmap to select an action to execute, so we need a value based algorithm. The chosen algorithm was Double Q-learning (DDQN)[38], which is an off-policy learning algorithm that can work with our prestored experience replay buffer. This algorithm solves the over estimation problem of DQN[39] through the greedy method. The algorithm was designed to train a Q network with parameter θ to predict the future value of each action by using reward signals. DDQN leverages a separate target network with parameter θ' that gradually updates from θ to stabilize the training process. We also used a prioritized experience replay buffer to increase efficiency by sampling important transitions to update the gradient. The objective of the value network Q was to minimize the Bellman error:

$$R_t + \gamma Q_{\theta'}(s_{t+1}, \underset{a}{\text{argmax}}\, Q_\theta(s_{t+1}, a)) - Q_\theta(s_t, a_t).$$

Our Q network was based on previous affordance prediction networks; only one additional convolution layer was used to output the Q-value. We also rotated the input images according to every possible end effector orientation to obtain the Q-values for each action in the action space. The full value map can be used as an affordance map to identify the best grasping point and rotation. The training process was performed using an NVIDIA RTX 2070 GPU for 5000 steps and took 3 hours.

Chapter 4

Experiment

4.1 System Overview

4.1.1 Hardware System

The medicine delivery robot system (Fig. 4.1) comprises four parts, each defined by its function of perception, movement, grasping, or calculation. These parts enable the robot to complete a series of medicine delivery tasks from the pharmacy to the ward.

Perception: Perception was realized using an Intel RealSense D435 RGB-D camera with a resolution of 640×480 mounted on the end effector of the manipulator VX300s.

Movement: For the UGV, we used the small mobile platform Clearpath Jackal. The size of the robot was $508 \times 430 \times 250$ mm^3, the maximum load was 20 kg, the maximum speed was 2 m/s, and the four-wheel differential drive can be used for more flexible travel in a multiperson indoor environment.

Grasp: We mounted an Interbotix ViperX300s manipulator on the Jackal. The manipulator had 6 DOFs and full rotation of 360°, a maximum reach of 750 mm, a maximum load of 750 g, and a two-finger parallel gripper as its end effector.

Calculation: A mounted Intel NUC computer on the Jackal was used to control the VX300s manipulator control and receive images from the D435 camera. The image information was transmitted using ROS over the local area network to another workstation equipped with an NVIDIA GTX 1080 GPU for prediction, and the prediction results were returned to the manipulator.

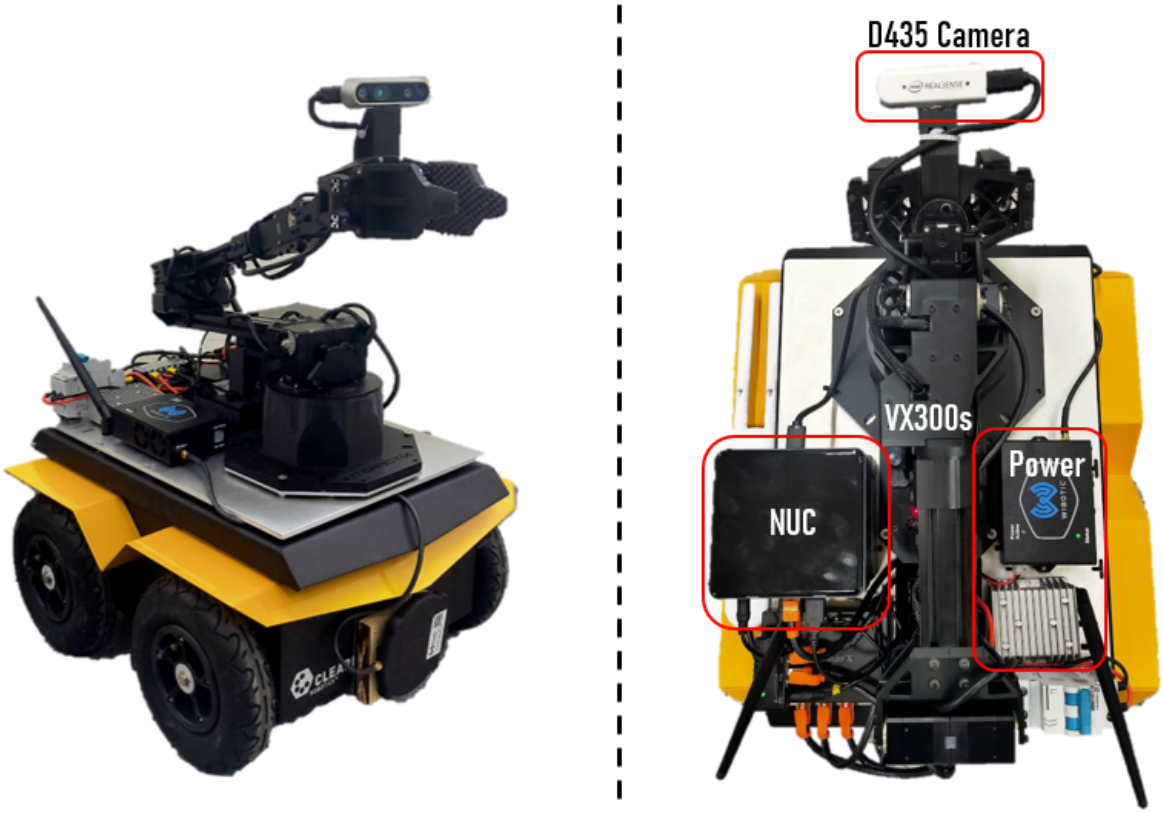

Figure 4.1: Medicine Delivery Robot. Perception, movement, grasping, and calculation functions were integrated on the Jackal, and it can shuttle between the ward and the pharmacy to complete handover and medicine delivery tasks.

4.1.2 Graphical User Interface

To facilitate user operation, we designed a graphical user interface as presented in Fig. 4.2. The user can click on each action while related pictures are displayed on the screen. The user interface can be used for manipulator control, gripper control, vehicle movement, and task execution selection.

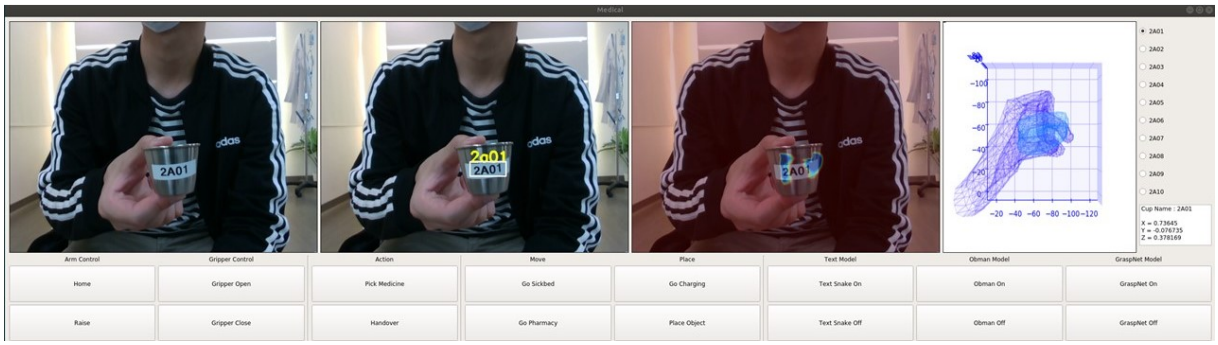

Figure 4.2: Graphical User Interface. Images, from left to right, are of the original image, the predicted brand name of the grasped medicine cup, the prediction result of the HERoS model, and the prediction result of the Baseline II. Joint Reconstruction [2] (ObMan model). The buttons below are for the movement control, manipulator control, and trigger tasks.

4.2 Human-to-Robot Handover Experiment

In this section, we demonstrate the effectiveness of our approach with experiments. We compare the concepts and effects of various human-to-robot handovers, including the methods we propose. Further apply this method to practical application, complete the medicine delivery robot system, and compare the effects of manual operation and learning-based approaches.

4.2.1 Experimental Setup

We used a mobile manipulator platform, and put the manipulator system in a fixed area, prohibiting moving vehicles to increase the fairness of the experiment. Testers arbitrarily handed over objects to the robot. During the experiment, we tested the following 5 approaches, conducting 20 trials for each object, and recorded the number of successes, and the reason for failures.

Baseline I. Hand [29] + Object Poses [30] (DOPE & MANO): This approach represented the intuitive method of handovers, which predicted the pose of the object and the human hand

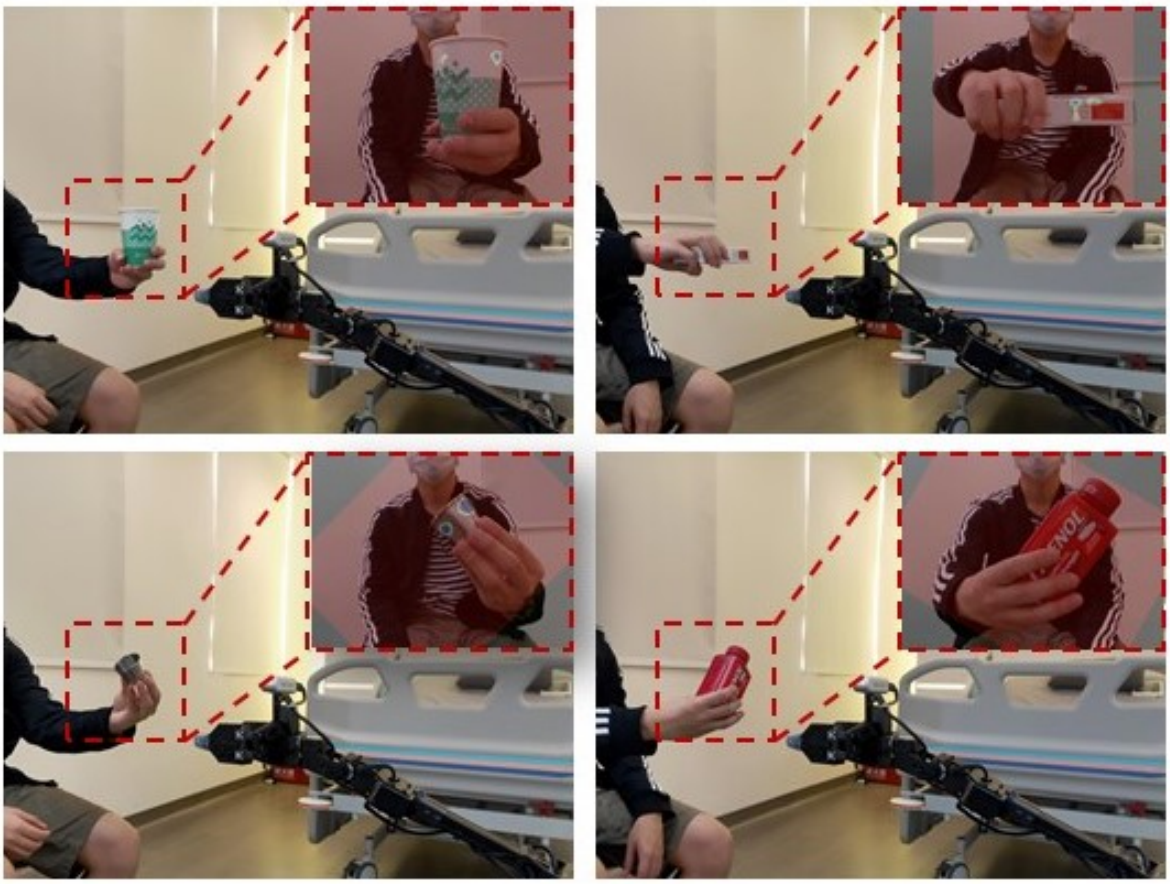

Figure 4.3: We performed 20 trials for each objects and classified the tasks with respect to four orientations: vertical, horizontal, diagonally to the right, diagonally to the left (taking each orientation of five trials in random poses); we did so to ensure fairness when the outputs of various models were compared with the object taken by the personnel.

separately, where the algorithm excluded the hand to grasp the target. The object pose prediction component used the state-of-the-art DOPE model [30], and the hand prediction component used the MANO model [29]. We combined the two and perform post-processing, and finally selecting the grasp pose for the handover task. (Fig. 4.4)

Baseline II. Joint Reconstruction [2] (ObMan): As prediction of the object and hand separately led to occlusion problems, we tried to use the ObMan [2] model to directly generate the pose of the object and the hand through joint reconstruction in order to obtain their relative positions. The reconstructed result was mapped to real-world coordinates by coherent point drift methods [3], and the grasp pose was generated through the algorithm to complete the handover task. (Fig. 4.5)

Baseline III. Modified GG-CNN [4, 5] (Rosenberger et al.): The approach described in Chapter. 2.2 was similar to the goal we want to achieve. We used this method as the baseline for this experiment and applied it to the D435 Camera and ViperX300s manipulator.

HERoS (Current approach): The proposed system called **HERoS** was described in Chapter. 3, used RGB-D images to predict the position and orientation that can be grasped.

Teleoperation: In order to compare the success rate of the manual control and the autonomously grasping methods, we reported the results using a joystick to control the manipulator to complete this task.

The algorithm mentioned in the Dope & Mano and ObMan approaches, meets the following conditions and Algorithm 4.1 in order to achieve the purpose of safety and accurate grasping:

- Distance: Within 10-70 cm from the manipulator.
- Protection: The finger joints are not accessible by the manipulator trajectory.
- Confidence: The grasp point of each frame must be less than 5 cm for 5 consecutive times.

All approaches use the same processes for recording and evaluation. During the experimental handover task, we performed 20 trials for each object with four orientations: vertical, horizontal, diagonally to the right, and diagonally to the left (Fig. 4.3). For each orientation, five trials with random poses were performed. Because these approaches ought to achieve real-time grasping, if the manipulator did not begin to perform the grasp action within 12 seconds after

Algorithm 4.1 Dope & Mano / ObMan Grasp Selection Algorithm

Input:
Five grasp pose in continuous time, ${}^{O}X^{G}(t)$;
Five hand joint points in continuous time, ${}^{H}X^{G}(t)$;
Output:
Target pre-grasp pose, ${}^{T}X^{G}$;

for $t = 0; t < 5; i++$ **do**
  if $10 <$ the depth of ${}^{O}X^{G}(t) < 70$ and ${}^{O}X^{G}(t) \cap {}^{H}X^{G}(t) = \emptyset$ **then**
    if $t > 0$ and $d({}^{O}X^{G}(t), {}^{O}X^{G}(t-1)) < 5$ **then**
      ${}^{T}X^{G} = ({}^{T}X^{G} + {}^{O}X^{G}(t))/2$
    else if $t = 0$ **then**
      ${}^{T}X^{G} = {}^{O}X^{G}(t)$
    else
      return $None$
    end if
  else
    return $None$
  end if
end for
return ${}^{T}X^{G}$

the start of the experiment, the tester changed the posture of the object and continued to wait for the prediction calculation. After the manipulator had performed the grasp action, whether the grasp was a success or failure was recorded.

In the experiment, it is currently only applicable to Asian hands, without gloves. The background is not specially regulated, mainly based on the clothes worn in daily life and the space of the laboratory. If we want to consider hands of different skin colors, we can increase the diversity of dataset.

4.2.2 Evaluation Metrics

We used a set of indicators to evaluate the performance of the system and analyze it by recording the number of successes, failures and their reasons.

Success: How often the robot was able to successfully take the object from the human's hand safely.

Planning Fail: The robot failed to predict the object in the human hand, resulting in an inability to grasp the object but the human's fingers were not touched.

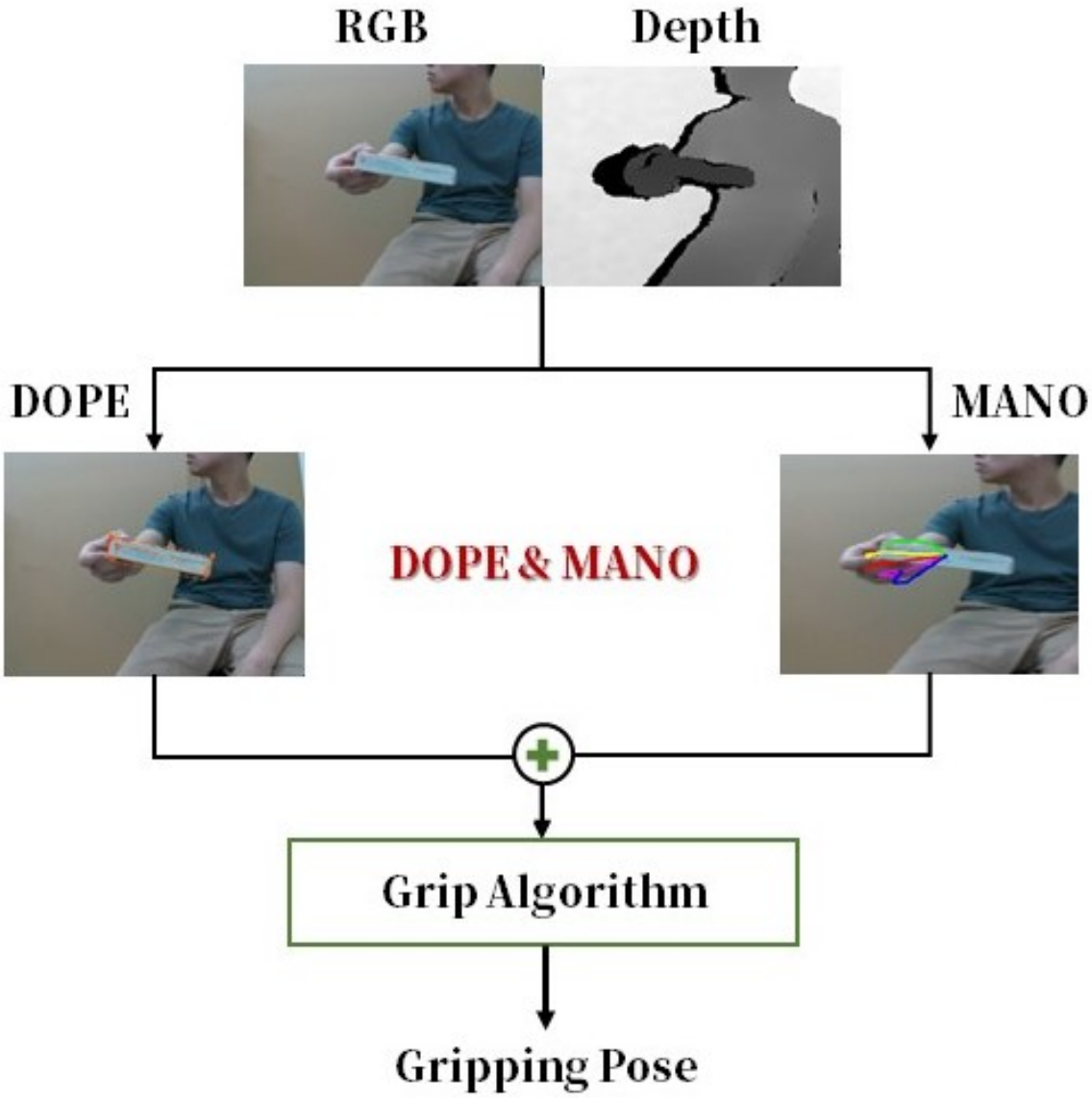

Figure 4.4: Baseline I. (DOPE & MANO). We used DOPE to predict the object pose and MANO to obtain the hand pose. Then, the grasping algorithm 4.1 was used to generate a suitable grasp pose and complete the handover task.

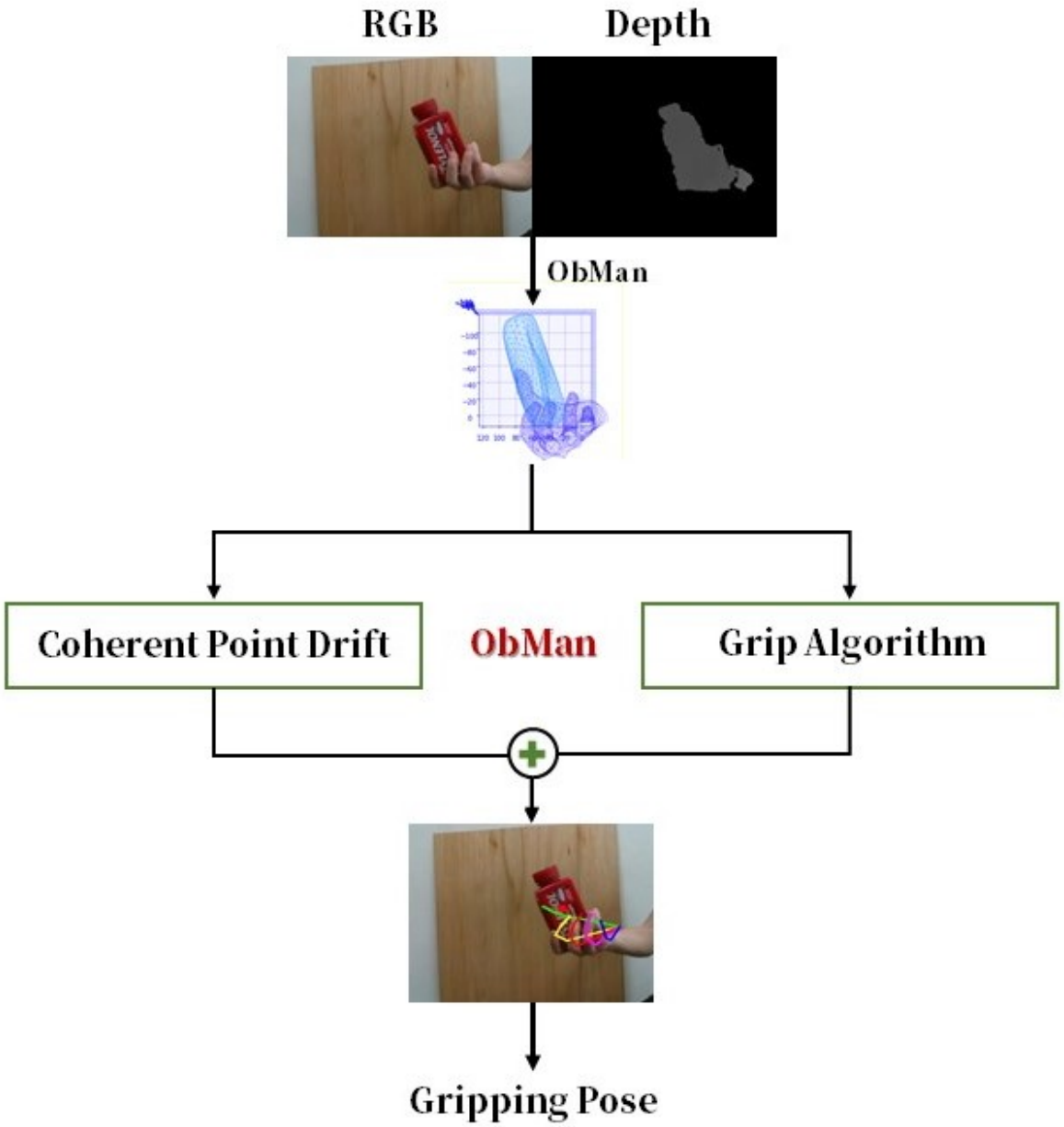

Figure 4.5: Baseline II. (ObMan) approach. We used ObMan reconstruction to predict the object and the human hand to avoid occlusion problems. Then, the coherent point drift method [3] was used to project coordinates in the real world and the grasp algorithm, as described in Algorithm 4.1, was used to generate a suitable grasp pose to complete the handover task.

Touched Fingers: Regardless of whether an object is grasped or not, if the robot grasps the human fingers, it will be considered a failure and recorded.

Hand Occlusion:

To ensure fairness of the comparison of various approaches and quantitative data analysis, we captured the current frame of the grasp execution and human label to obtain the occlusion relationship between the hand and the object. We categorized hand occlusion into either $< 40\%$ or $> 40\%$ (Fig. 4.6).

$$occlusion = \frac{(object\ area\ \cap\ hand\ area)}{object\ area}$$

Figure 4.6: Example capture of the current frame of the grasp execution for labelling hand/ object segmentation and estimating the occlusion ratio.

4.2.3 Results

Table 4.1: Individual object grasping results using YCB dataset in comparison with modifed GG-CNN[4, 5] baseline. An object is carried out 20 trials, a total of 360 handovers

	Hand Occlusion $< 40\%$									Avg.	
Method \Object	SPAM	Banana	Lemon	Strawberry	Peach	Pear	Plum	Mustard	Sugar	Success Rate	σ
Baseline III. Modified GG-CNN	80%	75%	75%	90%	70%	85%	80%	85%	80%	80%	$\pm 5.77\%$
HERoS (Current approach)	**95%**	**85%**	**80%**	**95%**	**90%**	**90%**	**90%**	**95%**	80%	**88.89%**	$\pm 4.82\%$

We first compared with Baseline III. (Rosenberger et al. [5]), which used a modified GG-CNN for grasping. 9 YCB objects with $< 40\%$ occlusion were evaluated, and we replicated the success rate reported in [5] using our manipulation platform. Our proposed affordance prediction method achieved higher success rate, shown in Table. 4.1.

Table 4.2: Comparison of the success rate of grasping four objects (400 handovers) across approaches.

$Method$ \Hand Occlusion	$< 40\%$	$\geq 40\%$	σ
Baseline I. Hand [29] + Object Poses [30]	61.6%	45%	$\pm 9.01\%$
Baseline II. Joint Reconstruction [2]	75%	60.97%	$\pm 9.01\%$
Baseline III. Modified GG-CNN [4, 5]	62.5%	50%	$\pm 3.53\%$
HERoS (Current approach)	**93.87%**	**93.54%**	$\pm 5.44\%$
Teleop	95.55%	91.43%	$\pm 6.49\%$

In a separate experiment, Table. 4.2 further showed the results of each of the main metrics in our system evaluation process. We first reported that our hardware equipment is sufficient to complete the handover task by teleoperation. The success rate of other three benchmark methods (DOPE & MANO, ObMan, and Rosenberger et al. [5]) were found decreased due to hand occlusions, but our method consistently outperformed the success rate, suggesting the efficacy and reliability of our method to severe hand occlusion ($> 40\%$).

The overall success rate of our method for all objects was 93.5%. It is noted that the success rate in Baseline III decreased to 62.5%. This may be caused by the model for detecting objects uses a YOLO V3 object detector [18], which was prone to false positives with a complex background and occlusion problems.

In the Baseline I. (DOPE & MANO) experiment, this method was used to predict the positions of the objects and fingers separately and then performed post-processing calculations. Common problems were that the hand occludes the objects and results in pose estimation errors. As the object cannot be detected if there was an occlusion, the DOPE model was only able to successfully predict the pose of the object during the experiment when the fingers are placed on the sides of the object.

Compared with the Baseline II. (ObMan) approach, the method of using hands and objects for joint reconstruction deals with the problem of not being able to detect objects. However, during post-processing, the reconstructed 3D-point cloud needs to be projected onto objects in real world coordinates to obtain the grasp points, which often causes errors such as skew.

For HERoS, no fingers were touched. This means that we can effectively avoid touching human fingers when handing over objects, and it also proves that the end-to-end method performs better than reconstruction or separate predictions of the hands and objects. In addition to solving the occlusion problem, it also reduces reaction time and improves stability.

4.3 Complete System Experiment

For infectious diseases, hospital cluster infection is more dangerous than community cluster infection. Therefore, isolation and medical care in the hospital are key issues. The purpose of this experiment is to reduce the contact between medical staff and patients. Use the medicine delivery robot to automatically shuttle between the wards. Therefore, we design a situation where we can use the robot with handover to complete the medicine delivery task that interacts with the patient. 1. Pick up a small medicine cup at the pharmacy. 2. Move to the side of the sickbed. 3. Handover interaction with patients. 4. Recycle the medicine cup.

4.3.1 Experimental Setup

On the basis of human-to-robot technologies and approaches, this thesis proposes an end-to-end medicine delivery robot system to reduce the risk of infection between medical staff and patients; thus, this technology has practical applications. This experiment systematically evaluated success rates and execution times using both manual teleoperation and a learning-based approach. The medicine delivery robot system was divided into four parts, as presented in Figure. 4.7; these parts were connected in series to achieve a complete medicine delivery task. The experiment was performed over 10 trials in a simulated ward.

Grasp: The first part begins at the pharmacy. The text on a small medicine cup was identified by using the Moran model [40]. If grasping of the medicine cup was achieved, this part was considered successful.

Movement: After successfully grasping the medicine cup, the robot proceeds to the second part of transporting the medicine cup from the pharmacy to the sickbed. This part includes all of the robot platform mobile tasks, which were completed through ultrawideband (UWB) localization and a prebuilt map combined with the A* path planning algorithm.

Handover: The robot hands the medicine cup to the patient. The approach of this thesis was to grasp the medicine cup from the patient's hand to achieve the human-to-robot handover task.

Placement: The vehicle returns to the pharmacy and accurately places the medicine cup in a specific location by using the Apriltag2 system [41] to obtain a pose to complete this task.

During the operation of the system, RGB-D images can be transmitted over a wireless network to perform calculations on a computer equipped with an NVIDIA GTX 1080 GPU. In the experiment, a set of indicators was used to evaluate the performance of the system. We evaluated and analyzed the success rate of each part and the overall time and the success rate of completing the entire task for each trial.

4.3.2 Results

The recorded times and success rates for various tasks demonstrated that the system is robust and reliable.

Grasping a small medicine cup from the shelf was difficult because the gripping point was very narrow. Even when joystick control was used, the cup was dropped often or collisions occurred. The success rate was lower for the joystick than for learning-based method, indicating that our hardware and software can be combined to achieve accurate control and outstanding performance in the following handover tasks.

For the movement part, we used UWB for localization and planned a path using the A* algorithm. However, when moving to the sickbed to perform handover, the robot must be moved to a place within 75 cm of the human because the manipulator range is only 75 cm. If the human is lying on a sickbed, slight deviations in the accuracy of the target point (of 10 or 20 cm) may occur. These deviations may cause the failure of the handover task. After the medicine delivery robot reaches its destination, the joystick was used to manually adjust the position of the robot and ensure the fairness of the task. Thus, in the movement part, more effort is necessary to achieve precise localization and reach the specified target point.

The handover task is the most cruicial for the system. A success rate of over than 80% was achieved, and the handover was completed within 20 seconds. Thus, the handover method could be used in various situations.

The placement task required the robot to find the pose for the placement point through Apriltag2 [41]. The robot placed the small medicine cup into a recycling box on the ground. The suc-

cess rate was 100%. In future studies, the robot could be challenged to put the small medicine cup on a recycling shelf.

Figure 4.7: Upper left picture displays the **Grasp** action; the robot is grasping a small medicine cup. The upper right picture displays the **Movement** action, in which the robot delivers medicine cups. The lower right picture presents the **Handover** action, which is key focus of this thesis. The lower left picture presents the **Placement** of the medicine cup to recycle the medicine cup and complete the series of tasks.

Table 4.3: The success rate of each part of the complete system process.

Method/Task	Grasp	Movement	Handover	Placement	Success Rate	Average Time(s)
Teleoperation	70%	100%	100%	100%	70%	117.8
HERoS	90%	100%	80%	100%	72%	152.9

4.4 Grasp Choice Experiment

In the previous experiment in Chapter. 4.2, we showed that the target object can be successfully grasped, but some successful grasp may violate some grasp choice constraints (such as

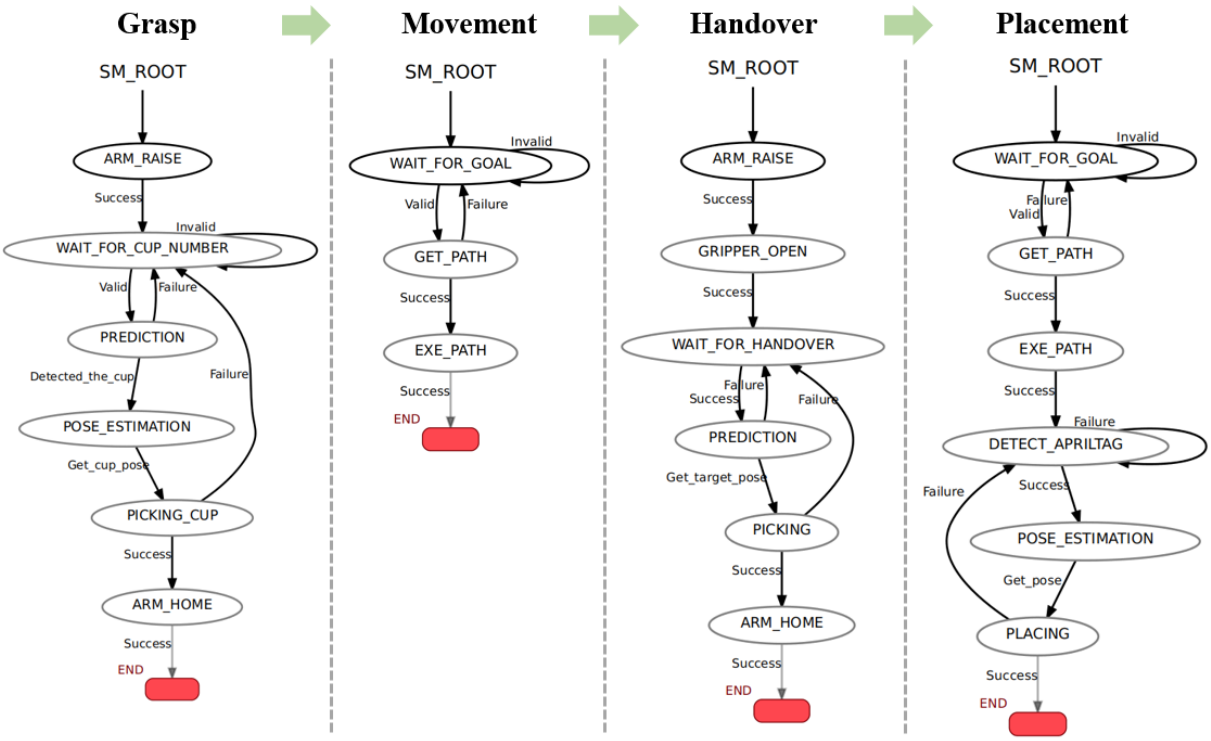

Figure 4.8: Medicine delivery robot system smach. The whole system will start from automatic charging and go through the four main parts Grasp, Movement, Handover, Placement. If it fails during execution, the action will be repeated until it succeeds. After the process is over, it will automatically return to the wireless charging station.

social convention). In this experiment, we further used deep RL to learn the associations of raw image inputs and grasp choice of human preferences. We set up our robot arm using the model described in Chapter.3.4 and continuously updated the model through the guiding of rewards given by users.

4.4.1 Experimental Setup

We designed fined two grasp types that were suitable for one of the two 6DoF grasp choices (front and top views), as shown in Fig. 4.10. The robot arm moved to the initial positions of the two grasp choices positions, and collected two input RGB-D images. The inputs were then pre-processed with 4 planar rotations, resulting in $224 \times 224 \times 4 \times 2$ before feeding into the deep model. The model predicted the Q-values of the outputed 2×4 6DoF affordance map, and the best grasping choice and orientation were selected. It is often to have a successful grasp from both grasp choices. However, we have set preferences for the robot to learn. We programmed the model to learn that when the user was holding the bottom of object, the robot should approach

from the top to stay away from the user’s fingers. Noted that no human hand detector nor human grasp classifier were used.

4.4.2 Training-stage

The deep RL model was trained while the robot interacted with human users, and a total number of 100 handovers was carried out. During each handover the reward given by the user. For every trial where the handover task was completed, it sampled from the prioritized reply buffer for training. 6-10 points were given when the grasping pose matched the human preference, 1-5 points were given when the pose grasped the object but did not match the human preference, and -5 points was given if the grasping fails or collision with a finger.

4.4.3 Results for Known and Novel Objects

We carried out a pre-test and a post-test to evaluate the grasp choice of the pre-trained DRL model before and after the training with 100 handovers. We had 40 handovers interacting with the robot arm, but without updating the weights. All trials usded known objects shown in the training stage. As mentioned the model generated 4 orientations $\times$ 2 grasp choice, and the best grasp position and orientation among the 8 affordance maps were selected. According to the experimental results, we found that the overall success grasp achieve 82.5%, and the probability for the preference was 47.5%.

The same procedure was performed for the model trained with deep RL. We tested 40 handovers for known objects (shown in the training stage) and 20 handovers for novel objects, which were not included in any of our training dataset HERoS. After obtaining rewards from humans and training for 100 trials, the average probability of grasping using the human preference was 82.5%. This means that the model can be updated with human-preferred grasping through learning to achieve a more preferred grasp choice, but a trade-off was found that the success rate (based on predicted positions and orientations from the affordance map) slightly decreased to 73.33%.

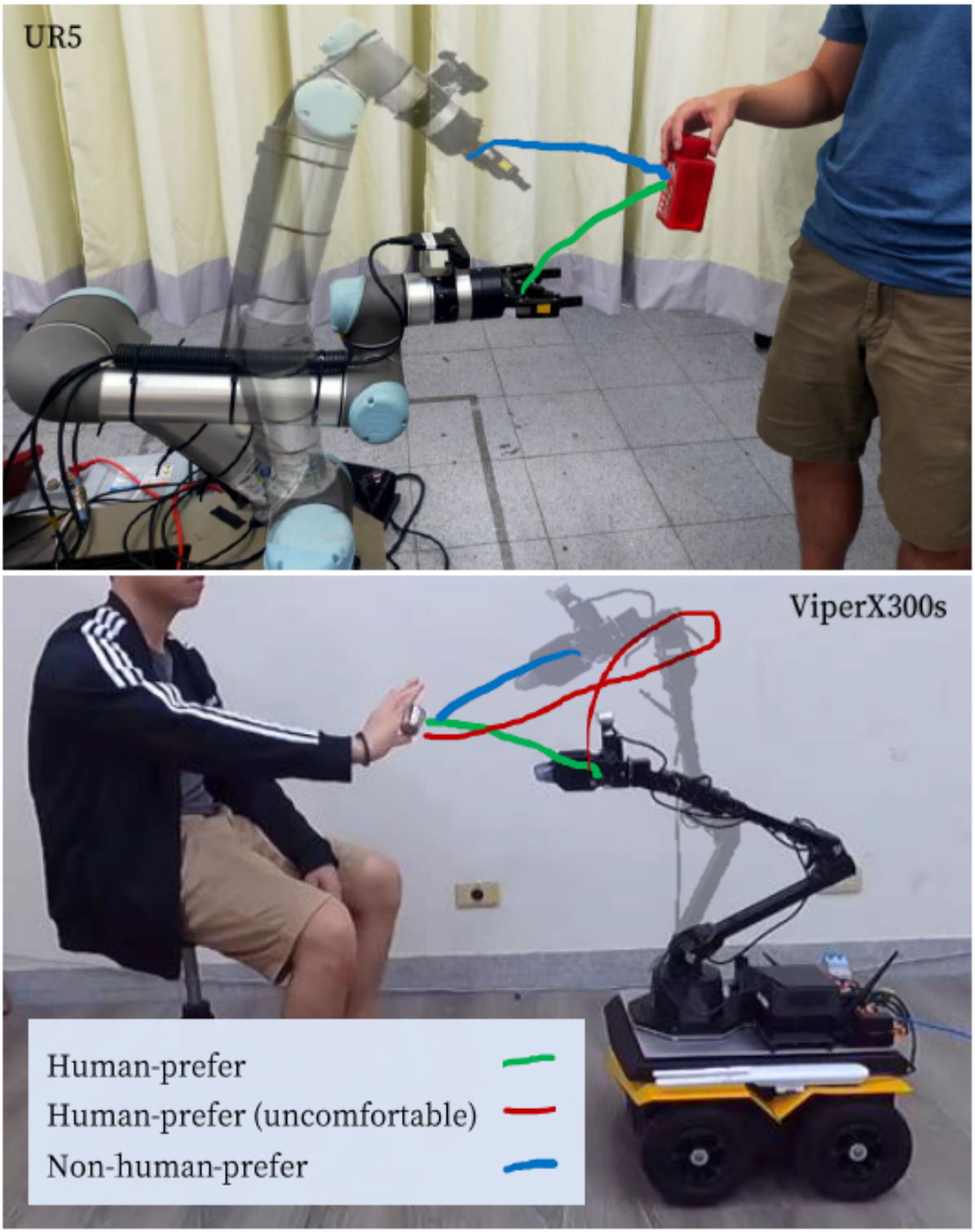

Figure 4.9: The trajectory of the manipulator during the handover task. The green line is the human-prefer trajectory, the red line is the human-prefer but not comfortable trajectory, and the blue line is the non-human-prefer trajectory.

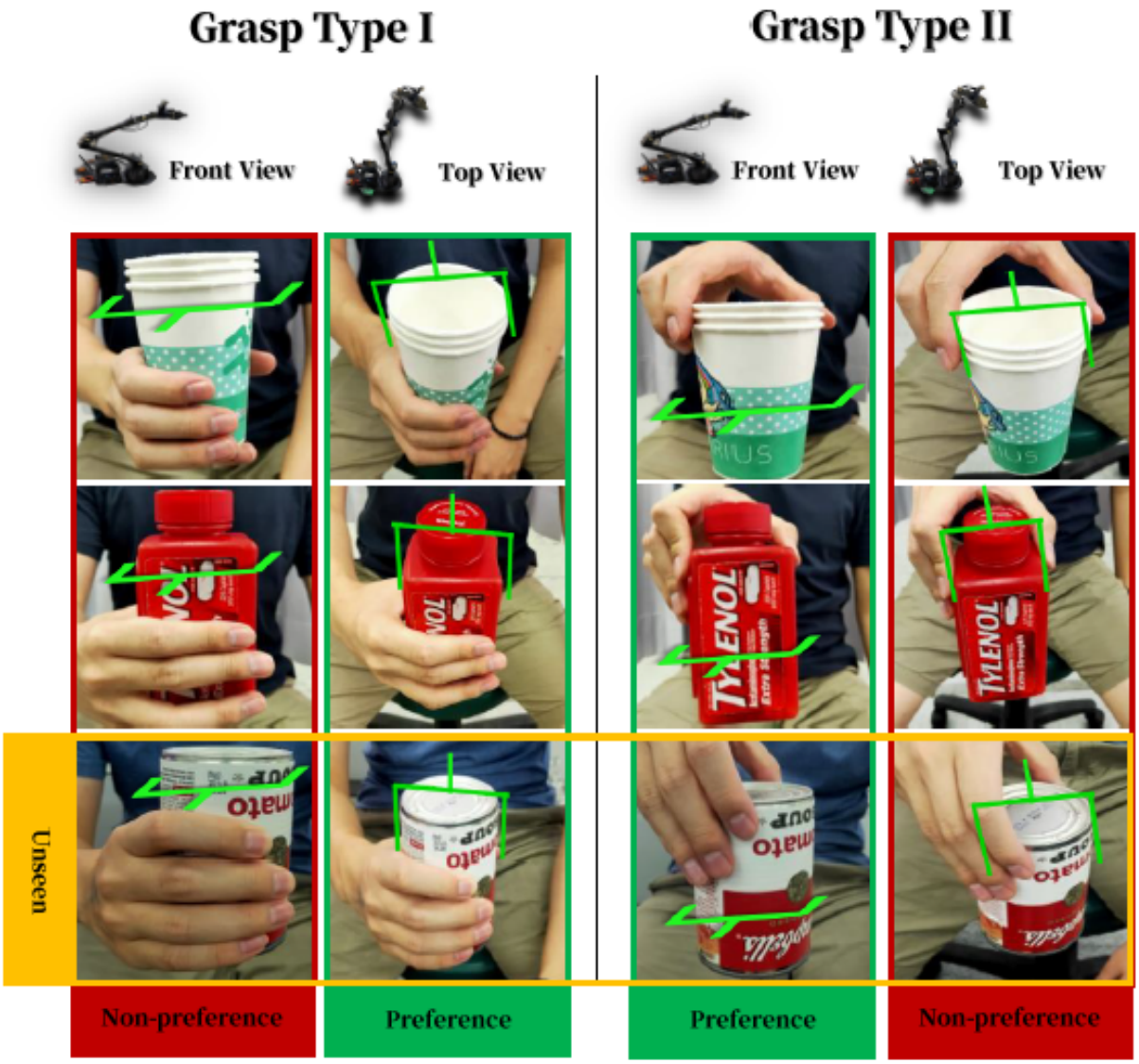

Figure 4.10: There are Grasp Type I and Grasp Type II preferences for picking objects. Grasp Type I is picking the middle of objects, and Grasp Type II is picking the top of objects, using different perspectives to predict objects, and finally selecting the grasping perspective preferred by humans.

Chapter 5

Conclusion

Human-to-robot handover has many challenges. Considering the issues of complex backgrounds, occlusion and trajectory, we presented an end-to-end approach for handover. We filtered out depth noise to increase the robustness of the model. Without using separate predictions of the hands and objects, we used RGB-D images as input to generate affordance maps for selecting the grasp point, thus solving the occlusion problem. Then we used deep reinforcement learning to train our prediction model. After each grasping trial, the user gives a point reward, allowing the robot to learn the grasp choice preferred by humans. In addition, we also proposed a medicine delivery machine system that integrates ground vehicles and a manipulator to enable it to move around the hospital and complete the medicine delivery task.

In the future, we believe the same approach could also be applied to many applications of human-robot collaboration. In future work, we will 1) use off-line RL training to jointly train affordance map with preference, and 2) use end-to-end methods to modify the deep RL network architecture and generate a close-loop 6D pose using RGB images, which can make the manipulator trajectory more legible and friendly.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[2] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid, "Learning joint reconstruction of hands and manipulated objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 807–11 816.

[3] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 12, pp. 2262–2275, 2010.

[4] D. Morrison, P. Corke, and J. Leitner, "Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach," *arXiv preprint arXiv:1804.05172*, 2018.

[5] P. Rosenberger, A. Cosgun, R. Newbury, J. Kwan, V. Ortenzi, P. Corke, and M. Grafinger, "Object-independent human-to-robot handovers using real time robotic vision," *IEEE Robotics and Automation Letters*, vol. 6, no. 1, pp. 17–23, 2020.

[6] G.-Z. Yang, B. J. Nelson, R. R. Murphy, H. Choset, H. Christensen, S. H. Collins, P. Dario, K. Goldberg, K. Ikuta, N. Jacobstein *et al.*, "Combating covid-19—the role of robotics in managing public health and infectious diseases," 2020.

[7] G. Hager, V. Kumar, R. Murphy, D. Rus, and R. Taylor, "The role of robotics in infectious disease crises," *arXiv preprint arXiv:2010.09909*, 2020.

[8] A. Di Lallo, R. R. Murphy, A. Krieger, J. Zhu, R. H. Taylor, and H. Su, "Medical robots for infectious diseases: Lessons and challenges from the covid-19 pandemic," *arXiv preprint arXiv:2012.07756*, 2020.

[9] K. Jovanovic, A. Schwier, E. Matheson, M. Xiloyannis, E. Rozeboom, N. Hochhausen, B. Vermeulen, B. Graf, P. Wolf, Z. Nawrat *et al.*, "Digital innovation hubs in health-care

robotics fighting covid-19: Novel support for patients and health-care workers across europe," *IEEE Robotics & Automation Magazine*, 2021.

[10] M. Tavakoli, J. Carriere, and A. Torabi, "Robotics, smart wearable technologies, and autonomous intelligent systems for healthcare during the covid-19 pandemic: An analysis of the state of the art and future vision," *Advanced Intelligent Systems*, vol. 2, no. 7, p. 2000071, 2020.

[11] S. O'Meara, "Hospital ward run by robots to spare staff from catching virus," *New Scientist (1971)*, vol. 245, no. 3273, p. 11, 2020.

[12] E. Ackerman, "Autonomous robots are helping kill coronavirus in hospitals," *IEEE Spectrum*, vol. 11, 2020.

[13] J. Ding, Y. Lim, M. Solano, K. Shadle, C. Park, C. Lin, and J. Hu, "Giving patients a lift - the robotic nursing assistant (rona)," in *2014 IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, 2014.

[14] W. Yang, C. Paxton, A. Mousavian, Y.-W. Chao, M. Cakmak, and D. Fox, "Reactive human-to-robot handovers of arbitrary objects," *arXiv preprint arXiv:2011.08961*, 2020.

[15] C.-M. Huang, M. Cakmak, and B. Mutlu, "Adaptive coordination strategies for human-robot handovers." in *Robotics: science and systems*, vol. 11. Rome, Italy, 2015.

[16] F. Cini, V. Ortenzi, P. Corke, and M. Controzzi, "On the choice of grasp type and location when handing over an object," *Science Robotics*, vol. 4, 2019.

[17] W. Yang, C. Paxton, M. Cakmak, and D. Fox, "Human grasp classification for reactive human-to-robot handovers," *arXiv preprint arXiv:2003.06000*, 2020.

[18] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[19] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *arXiv preprint arXiv:1706.02413*, 2017.

[20] A. Zeng, S. Song, K.-T. Yu, E. Donlon, F. R. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo *et al.*, "Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 3750–3757.

[21] J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, and K. Goldberg, "Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 1957–1964.

[22] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," *arXiv preprint arXiv:1703.09312*, 2017.

[23] J. Mahler, M. Matl, X. Liu, A. Li, D. Gealy, and K. Goldberg, "Dex-net 3.0: Computing robust robot suction grasp targets in point clouds using a new analytic model and deep learning," *arXiv preprint arXiv:1709.06670*, 2017.

[24] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg, "Learning ambidextrous robot grasping policies," *Science Robotics*, vol. 4, no. 26, p. eaau4984, 2019.

[25] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[26] R. Julian, B. Swanson, G. S. Sukhatme, S. Levine, C. Finn, and K. Hausman, "Never stop learning: The effectiveness of fine-tuning in robotic reinforcement learning," *arXiv preprint arXiv:2004.10190*, 2020.

[27] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 3406–3413.

[28] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser, "Learning synergies between pushing and grasping with self-supervised deep reinforcement learning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[29] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 6, pp. 1–17, 2017.

[30] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield, "Deep object pose estimation for semantic robotic grasping of household objects," in *Conference on Robot Learning (CoRL)*, 2018. [Online]. Available: https://arxiv.org/abs/1809.10790

[31] A. Mousavian, C. Eppner, and D. Fox, "6-dof graspnet: Variational grasp generation for object manipulation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2901–2910.

[32] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[33] A. Gupta, A. Murali, D. P. Gandhi, and L. Pinto, "Robot learning in homes: Improving generalization and reducing dataset bias," in *Advances in Neural Information Processing Systems*, 2018, pp. 9112–9122.

[34] S. Chitta, I. Sucan, and S. Cousins, "Moveit! [ros topics]," *IEEE Robotics Automation Magazine*, vol. 19, no. 1, pp. 18–19, 2012.

[35] B. Calli, A. Singh, J. Bruce, A. Walsman, K. Konolige, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Yale-cmu-berkeley dataset for robotic manipulation research," *The International Journal of Robotics Research*, vol. 36, no. 3, pp. 261–268, 2017.

[36] K. Wada, "labelme: Image Polygonal Annotation with Python," https://github.com/wkentaro/labelme, 2016.

[37] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[38] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, no. 1, 2016.

[39] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[40] C. Luo, L. Jin, and Z. Sun, "A multi-object rectified attention network for scene text recognition," *arXiv preprint arXiv:1901.03003*, 2019.

[41] J. Wang and E. Olson, "Apriltag 2: Efficient and robust fiducial detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4193–4198.