actuators

MDPI

Article

Data-Driven Robotic Tactile Grasping for Hyper-Personalization Line Pick-and-Place

Zhen Xie *, Josh Ye Seng Chen, Guo Wei Lim and Fengjun Bai

Advanced Remanufacturing and Technology Centre (ARTC), Agency for Science, Technology and Research (A*STAR), 3 Cleantech Loop, #01/01 CleanTech Two, Singapore 637143, Singapore
* Correspondence: xie_zhen@artc.a-star.edu.sg

Abstract: Industries such as the manufacturing or logistics industry need algorithms that are flexible to handle novel or unknown objects. Many current solutions in the market are unsuitable for grasping these objects in high-mix and low-volume scenarios. Finally, there are still gaps in terms of grasping accuracy and speed that we would like to address in this research. This project aims to improve the robotic grasping capability for novel objects with varying shapes and textures through the use of soft grippers and data-driven learning in a hyper-personalization line. A literature review was conducted to understand the tradeoffs between the deep reinforcement learning (DRL) approach and the deep learning (DL) approach. The DRL approach was found to be data-intensive, complex, and collision-prone. As a result, we opted for a data-driven approach, which to be more specific, is PointNet GPD in this project. In addition, a comprehensive market survey was performed on tactile sensors and soft grippers with consideration of factors such as price, sensitivity, simplicity, and modularity. Based on our study, we chose the Rochu two-fingered soft gripper with our customized force-sensing resistor (FSR) force sensors mounted on the fingertips due to its modularity and compatibility with tactile sensors. A software architecture was proposed, including a perception module, picking module, transfer module, and packing module. Finally, we conducted model training using a soft gripper configuration and evaluated grasping with various objects, such as fast-moving consumer goods (FMCG) products, fruits, and vegetables, which are unknown to the robot prior to grasping. The grasping accuracy was improved from 75% based on push and grasp to 80% based on PointNetGPD. This versatile grasping platform is independent of gripper configurations and robot models. Future works are proposed to further enhance tactile sensing and grasping stability.

Keywords: robotic grasping; data-driven; hyper-personalization; tactile sensing; FMCG

check for updates

Citation: Xie, Z.; Chen, J.Y.S.; Lim, G.W.; Bai, F. Data-Driven Robotic Tactile Grasping for Hyper-Personalization Line Pick-and-Place. *Actuators* **2023**, *12*, 192. https://doi.org/10.3390/act12050192

Academic Editors: Jing Wang, Zhijie Xu, Zhenyu Lu and Jonathan Gomez

Received: 10 January 2023
Revised: 22 February 2023
Accepted: 23 February 2023
Published: 1 May 2023

1. Introduction

Robotic grasping is an area of study in which collaboration between grasping software and hardware is key. Figure 1a,b demonstrates bin-picking and conveyor-picking applications for robotic grasping, respectively, while Figure 1c shows another more complex picking problem—shelf picking and place [1]. The co-development between grasping software and hardware provides many opportunities to address the challenges present in this area [2]. This project aims to overcome challenges faced by pre-programmed robots operating within the manufacturing or logistics industry. These robots are programmed to perform very specific tasks, and thus they require reprogramming whenever they are assigned to handle new applications [3]. They are not capable of manipulating novel objects in high-mix low-volume production scenarios. This inhibits the widespread adoption of robotic technology within the FMCG and logistics sectors, which are heavily loaded with fast-changing processes. In these sectors, the robotic system has to handle highly mixed stock-keeping units (SKUs), which include products possessing various physical properties such as being heavy, light, flat, large, small, rigid, soft, fragile, deformable, translucent, etc.

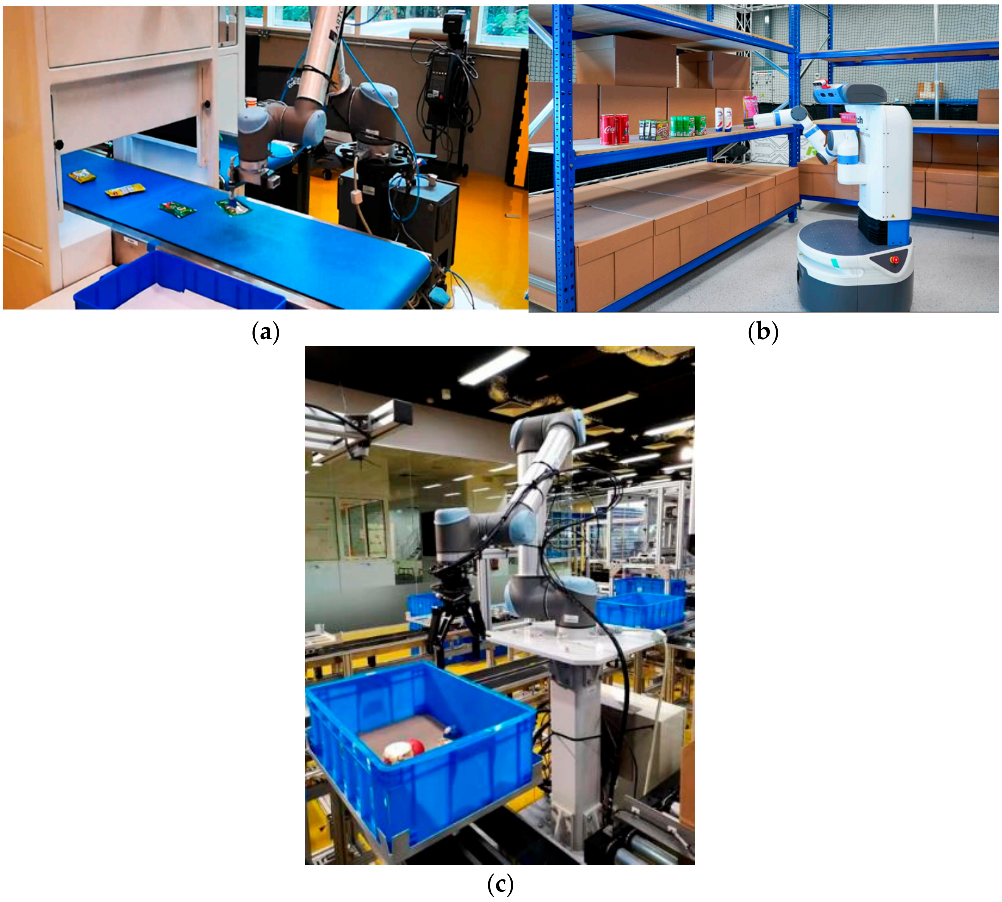


Figure 1. High-mix low-volume robotic grasping scenarios: (**a**) conveyor belt pick-and-place; (**b**) shelf picking based on versatile grasping; (**c**) robotic bin picking from a cluttered environment.

In this research, we have developed a robotic system to enable high-mix high-volume picking and packing of hyper-personalized products. This consists of a 3D camera-based robotic vision to capture RGB and point cloud data for object detection and point cloud segmentation, respectively. The vision system generates the information required for the robot to identify and locate each product within the robot's workspace. Using this information, the grasp pose detection module produces pose estimates of viable picking grasps, while the packing algorithm determines the feasible placing pose through the calculation of packing spaces of the products with respect to the carton boxes. With the picking and placing poses determined, the motion planning module consisting of kinematics and dynamics modeling of the robotic system plans a collision-free trajectory to ensure the products are successfully transferred from the tote boxes and packed into the carton boxes.

To be more specific, we propose tactile sensing based on Mask R-CNN and PointNetGPD [4], a further enhancement from PointNetGPD. Previously, a robotic grasping system consisting of a Robotiq 2f-140 gripper mounted on a UR10e robot (see Figure 2) based on push and grasp [5] and evaluated against novel non-textured and textured objects resulted in limited grasping accuracy and success rates of 75%. Hence, this research seeks to improve the development of PointNetGPD through the enhancement of grasping capability for novel objects with varying shapes and textures by utilizing soft grippers and deep-learning-based grasping algorithms. Our research focuses on the development of a flexible grasping technique for soft grippers modeled using a deep learning algorithm, the usage of tactile sensor information as feedback for the vision data to validate robotic grasping stability, and the improvement of the accuracy and success rate of the grasping model to 80%. The design and development of new soft grippers are not within the scope of this research. In terms of payload limitation, the official payload of the UR10e robot is 10 kg, but after factoring in the end-of-arm tooling (EOAT) weight and reachability, the product handling payload for the robot is around 8 kg. However, the soft gripper is driven by pneumatics with a payload of only 400 g, and the soft fingers can expand to accommodate up to a maximum size of 18 cm and can pick up an object that has a minimum height of 2 cm. In terms of shape and materials, the soft gripper is very flexible. The setup is best suitable to handle FMCG or food products that are fragile and lightweight such as chips, juice bottles, vegetables, soup cans, etc.

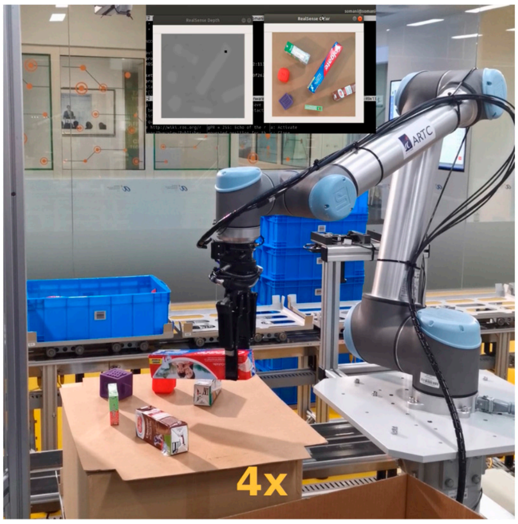


Figure 2. Push and grasping demonstration using Robotiq 2f-140.

This paper is an extended version of the conference paper titled "AI-Enabled Soft Versatile Grasping for High-Mixed-Low-Volume Applications with Tactile Feedbacks" [6], which was published at the 2022 27th International Conference on Automation and Computing (ICAC) in Bristol, United Kingdom.

2. Literature Review

In this literature, we will first cover state-of-the-art 3D grasping sampling benchmarks. These benchmarks contain different types of object data with labeled grasping poses based on specific gripper configurations, which can serve as the baseline for grasping learning. After that, market surveys on soft gripping and tactile sensing are discussed. A one-axis force sensor and Rochu gripper are selected for preliminary study in this stage. Last but not least, deep learning approaches and deep reinforcement learning approaches are compared with limitations being discovered.

2.1. 3D Grasping Pose Sampling Benchmarks

Robotic grasping starts with finding the region of interest (ROI) for grasping. After that, the learning-based approach will sample a large number of grasping poses around the object without prior information about the object [7]. This is the major advantage of deep-learning-based grasping. These kinds of approaches perform better when the object is novel or familiar. Then, the grasping pose with the highest score based on grasping evaluation metrics will be selected as the optimal grasping pose. There are considerably numerous benchmarks on grasping pose sampling. These benchmarks can be categorized into different groups based on the gripper configuration. For suction grippers, SuctionNet [8] is the most popular benchmark, while for the finger gripper, the Cornell dataset [9]; the Columbia grasp database [10]; DexYCB [11],as shown in Figure 3; Dex-Net [12]; and GraspNet [13] are the most prominent databases. Lastly, TransCG [14] can be used as the baseline for transparent object grasping.

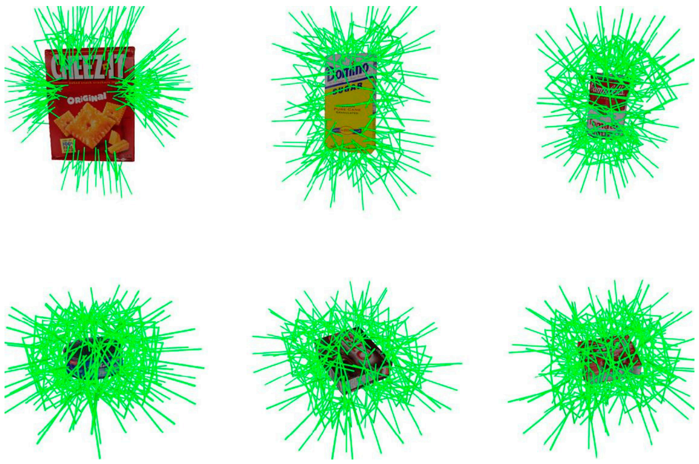


Figure 3. DexYCB dataset [11].

2.2. Soft Gripping Technology

In recent years, the soft robotic gripper industry has seen significant advancements, leading to the development of new types of soft grippers with improved performance and capabilities. These innovative soft robotic grippers offer a wide range of advantages over traditional rigid robotic grippers. One of the primary advantages of soft robotic grippers is their ability to handle delicate and fragile objects without causing damage. This makes them ideal for applications in the food, pharmaceutical, and electronics industries, where traditional robotic grippers may not be suitable. Soft robotic grippers also have the capability to conform to the shape of an object, creating a more secure grip than rigid robotic grippers. This makes them especially useful in handling objects that have complex or irregular shapes. Another advantage of soft robotic grippers is their safety. Unlike traditional robotic grippers, soft robotic grippers do not have sharp edges or hard surfaces that can cause injury to humans. This makes them suitable for use in environments where human interaction is likely, such as hospitals, research labs, and factories. Additionally, because of their pliable nature, soft robotic grippers can be programmed with a wide range of tactile feedback, allowing them to better sense their environment.

Soft gripping technology mainly can be categorized into three groups, actuation-based grasping, controlled stiffness, and controlled adhesion, as can be seen in [3]. Through lowering control complexity based on mechanical compliance and material softness, soft grippers are a great illustration of morphological computation. Researchers have been addressing advanced materials and soft components such as silicone elastomers and shape memory materials in recent years. In addition, gels and active polymers attract a lot of attention for the benefits of lighter and universal grippers due to their inherent material properties [15]. In addition, stretchable distributed sensors embedded in or on soft grippers greatly boost their interaction performance with gripping objects. However, research on soft grippers still faces a lot of challenges, including but not limited to miniaturization, sensor fusion, reliability, payload, control, and speed. In conclusion, soft gripping technology advances with material science, processing power, and sensing technology. Table 1 lists the state-of-the-art soft gripper solutions on the market, including key information such as capacity, grasping object type, and key technical specifications. There are several types of soft robotic grippers, including pneumatic, hydraulic, and dielectric. Pneumatic soft robotic grippers use air pressure to generate movement, while hydraulic soft robotic grippers use fluid pressure to generate movement. Dielectric soft robotic grippers use electrical fields to generate movement. Each type of soft gripper has its own unique advantages and disadvantages, and the choice of gripper type depends on the specific application and requirements. Despite the advantages, there are some disadvantages to soft robotic grippers. One disadvantage is their slower speed and lower precision compared to traditional robotic grippers. Soft robotic grippers are also more complex and difficult to design and manufacture, which can make them more expensive. Additionally, soft robotic grippers are less durable and have a shorter lifespan compared to traditional robotic grippers, resulting in higher maintenance costs.

Table 1. The state-of-the-art soft gripper solutions on the market.

Type/Brand	Payload	Grasping Objects	Tech Specs
Soft Robotics Inc. mGrip	Circular: diameter up to 145 mm × Parallel: width up to 100 mm, up to 3.4 kg	Food products of various sizes, shapes	Operating pressure −5 to 10 psi, IP67, weight 334 g(2-finger), weight 577 g(4-finger)
OnRobot Soft Gripper	2.2 kg max	Delicate/fragile objects, irregular shapes	0.77 + 0.168 kg total weight (base + attachment), work range: 11–75 mm (grip dimension)
Soft Robot Tech Co. (SRT) SFG	250–1550 g	Abnormal/fragile objects	Different models can be chosen according to weight/dimension specs
UBIROS Gentle Duo	1.5 kg (3.3 lbs) payload	Delicate objects or products that vary in size, shape, and weight during primary packaging	Fully electrical, weight 800 g (1.8 lb.), finger distance (base) 28 mm (1.1 in)
FESTO DHEF	1 kg	delicate products, undefined shapes	weight 475 g, position sensing accessories available
Rochu Soft Robotic Gripper	Catalog different finger modules that provide different max load	Food, textile, glass, logistics, and electronics	Pressure used; the weight of the gripper depends on the model used.
Piab piSOFTGRIP	250 g, 10–30 mm objects	Suitable for sensitive and lightweight objects of odd geometries and/or an unusual surface	Weight 170 g
Roplus UnisoGrip	Up to 2.5 kg, 10–30 cm gripping width	FMCG products	UR/ABB arm compatible, product weight 2.5 kg
SoftGripping by Wegard GmbH SoftGripper	200–400 g	Rectangular, cubic objects, 10–90 mm	SoftGripper: 120 mm height, 150 mm diameter
Applied robotics Flexible Smart Gripper	1.5 kg	Delicate objects	Fully electric, compatible with collaborative robots, Operating temperature: 5–60 °C, enhanced version provides force control
Gripwiq SofTouch	200 g to 6 kg	Packaged goods, food	4 fingers principle, round, Ø 40–220 mm, 2 fingers principle, square, 40–250 mm width, and 100–350 mm length
The Gripper Company 4-Finger Lip	25–96 mm diameter objects	Fragile objects	Max push/pull 100 N
Schunk ADHESO	N.A.	Fragile objects, electronics	UR compatible, adhesion based

Based on our soft gripping systems market review, as illustrated in Table 1, Soft Robotics, Applied Robotics, Soft Gripping, OnRobot, Rochu, and FESTO are the major market players. Most of them offer various types of soft grippers compatible with collaborative robots. As shown in Table 1, Rochu soft gripper [15] was down-selected as the design is modular and is capable of grasping fragile products such as food, product packaging, etc.

2.3. Tactile Sensors and Slipping Detection

A tactile sensor, also known as a touch sensor, is a device that detects physical contact or pressure. It can be used in various applications, including robotics, human–machine interaction, and industrial control systems. The performance of tactile sensors is evaluated based on factors such as sensitivity, accuracy, resolution, linearity, and repeatability. The type of material used in the sensing element, as well as the method of measurement, can also impact the performance of the sensor. When choosing a force sensor, it is important to select a sensitive, modular, and flexible pressure sensor. In this phase, we explore a one-axis tactile sensor that is capable of detecting grasping force. Based on surveys, there are many one-axis tactile solutions on the market, such as PUNYO [16], Gelsight DIGIT [17], BioTac [18], and HEX-O-SKIN [19]. However, one-axis tactile sensors will be explored in the next phase. As can be seen in Table 2, multiple types of pressure sensors are compared, including resistive, capacitive, piezoresistive, piezoelectric, magnetic, fiber optics, and ultrasound. The pros and cons are discussed in Table 2 as well. The main reason for not choosing the rest of the pressure sensors include the measurement complexity, temperature sensitivity, complex computations, and cost of the conditioning electronics. Keeping these considerations in mind, resistive force sensors are the most ideal for robotic grasping as they balance the above-mentioned considerations well despite not leading in any of them. These sensors are classified as normal pressure sensors and are sufficiently sensitive, simple, flexible, and inexpensive, causing them to be the most favored sensors in spite of generally detecting only a single contact point and being quite energy inefficient. In contrast, capacitive force sensors possess excellent sensitivity and spatial resolution, as well as a large dynamic range, but they are mostly complex and suffer from hysteresis-related problems. The most suitable resistive force sensors for our setup are FSRs FA201, FA400, and FA402 force-sensing resistors, as well as those found in the Arduino Kit. The sensor dimension is approximately 15.7 mm, which is a comfortable fit for the 24 mm Rochu soft fingertip, as demonstrated in Figure 4. Refs. [20,21] presents a mathematical model and a control strategy for regulating the slipping behavior of a planar slider and shows that the proposed control strategy can effectively improve the stability and performance of the system. Finally, accurate object manipulation is achieved by detecting and controlling slipping between the parallel gripper and the object. Ref. [22] designed a distributed control structure to overcome the uncertainty of object mass and soft/hard features. A model-free intelligent fuzzy sliding mode control strategy is employed to design the position and force controllers of the gripper. Ref. [23] described an online detection module based on a deep neural network that is designed to detect contact events and object material from tactile data, and a force estimation method based on the Gaussian mixture model is proposed to compute the contact information from the data.

Table 2. One-axis force sensor comparison based on various types of technologies.

Type	Classification	Advantages	Disadvantages
Resistive	Normal pressure	Good sensitivity; low cost; simple; flexibility	Generally detects single contact; high power consumption
Capacitive	Normal pressure	Excellent sensitivity; good spatial resolution; large dynamic range	Hysteresis; the complexity of measurement electronics
Piezoresistive	Skin deformation	High spatial resolution; high scanning rate in a mesh; structured sensors	Lower repeatability; hysteresis; temperature sensitive
Piezoelectric	Skin deformation	High-frequency response; high sensitivity; high dynamic range	Poor spatial resolution; dynamic sensing only; temperature sensitive
Magnetic	Skin deformation	High sensitivity; good dynamic range; no mechanical hysteresis; physical robustness	Suffer from magnetic interference; complex computations; high power consumption
Fiber optics	Skin deformation	Flexibility; small size; high sensitivity; biocompatibility; remote sensing capability	Fragility; relatively high cost of initial investment of interrogator
Ultrasound	Skin deformation	Fast dynamic response; good force resolution	Complex electronics; temperature sensitive; limited utility at low frequency

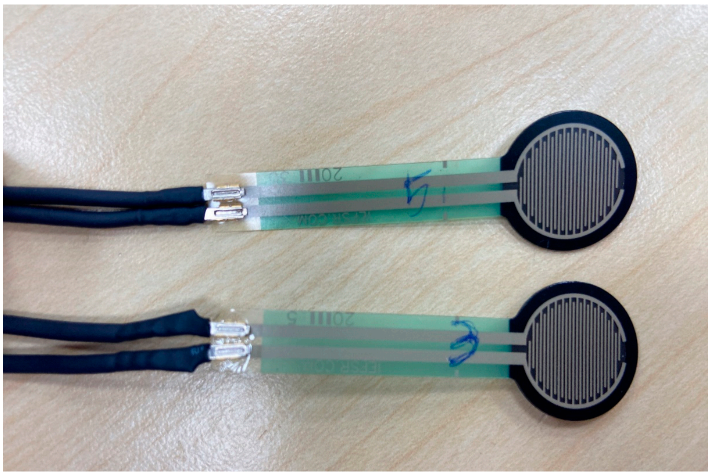


Figure 4. FSR resistor-based force sensor.

2.4. Data-Driven Approach

Generally, objects used for grasping can be categorized into three groups: unknown, familiar, and known [24,25], as can be found in Figure 5. For unknown objects, their models are not available to the robot initially. Hence, the first task most often includes the identification of the objects' structure or features from sensory data prior to performing the grasping action [26,27]. After that, specific features can be extracted that are indicative of good grasping. At the other end of the spectrum, known objects are objects that have been encountered by the robot and their grasps have been planned previously. The grasping pose generation is relatively straightforward when pre-programmed information about the object's size, shape, and surface texture is used to determine the best grasping strategy. In the middle are familiar objects that possess similar attributes to known objects but have never been encountered by the robot previously. The grasp planning requires the robot to find an object representation and a similarity metric to allow it to learn based on its previous experience with familiar objects [7]. As this is a very complex problem, current research is mostly focused on developing deep-learning models for grasping unknown objects. Some of the prominent research utilizes DCNNs [28], RGB-D images, and depth images of a scene [25]. These methods are generally successful in determining the optimal grasp of various objects [29], but they are more often than not restricted by practical issues such as limited data and testing.

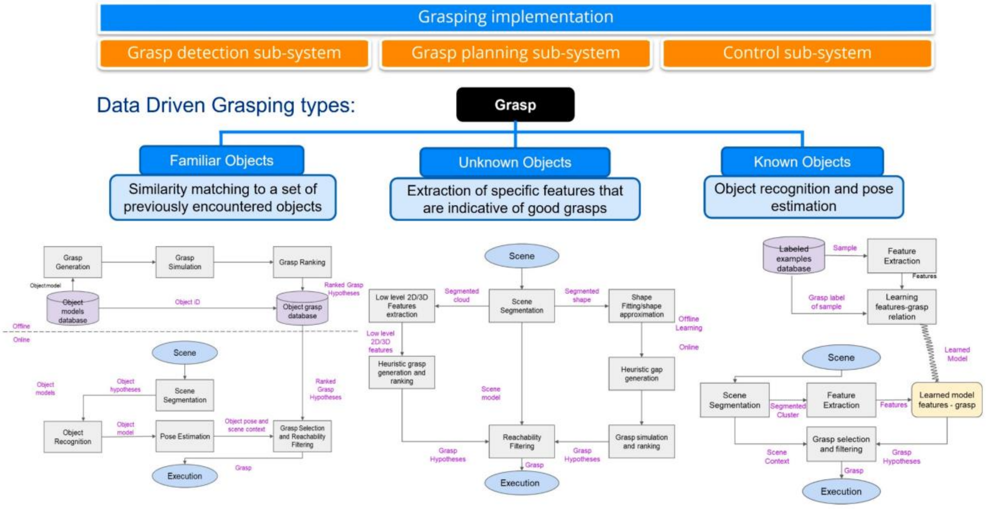


Figure 5. Data-driven grasping types [30].

Two of the most well-known data-driven approaches for robotic grasping are PointNet GPD and GraspNet [13,31], as illustrated in Figure 6. GraspNet is an open project for general object grasping that is still undergoing development. It contains 190 cluttered and complex scenes captured by a Kinect A4Z and a RealSense D435 camera giving rise to a total of 97,280 images. Each image is annotated with a 6D pose and dense grasp poses for each object. In total, the dataset contains over 1.1 billion grasp poses [31]. However, PointNet GPD performs better when the raw point cloud is sparse and noisy [4]. As a result, we proceed with PointNet GPD.

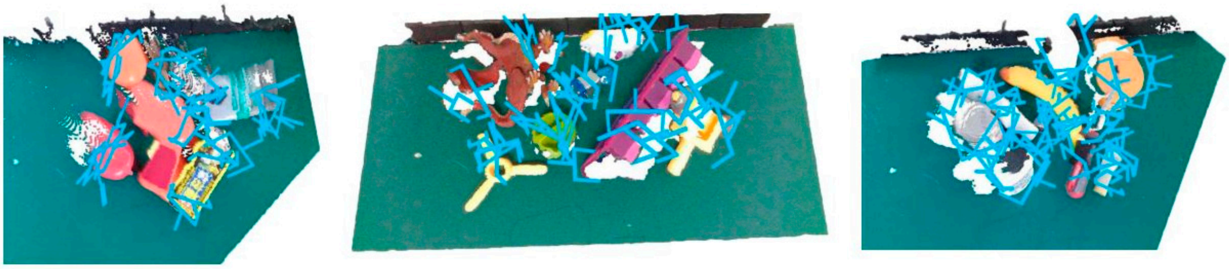


Figure 6. GraspNet [31].

2.5. Deep Reinforcement Learning Approach

One of the challenges in DRL for grasping is the sparse reward signal [32], as it can be difficult to provide meaningful rewards for successful grasps. To address this issue, researchers have explored different reward designs, including hand-designed reward functions and self-supervised methods [26] that generate rewards based on the progress of the grasp.

Another challenge is the high-dimensional observation space, which requires the use of complex models such as convolutional neural networks (CNNs) to process visual information [33]. Researchers have also explored combining DRL with other techniques, such as model-based methods, to improve the sample efficiency of the learning process [34].

Overall, DRL-based robotic grasping has shown promising results and continues to be an active area of research. However, further advancements are needed to make this approach more robust and scalable for real-world applications.

The three most popular deep reinforcement learning methods that bring about smarter and more adaptable robotic grasping algorithms are policy gradient methods, model-based methods, and value-based methods [35,36], as shown in Figure 7. These algorithms can be subdivided into two categories: off-policy learning and on-policy learning. The majority of DRL-based tactile sensing is based on off-policy learning, such as learning from demonstration.

An algorithm utilizing a tactile feedback policy can realize grasps provided by a coarse initial positioning of the hand above an object. In this DRL policy, the trade-off between exploration and exploitation is made through the use of a clipped surrogate objective within a maximum entropy reinforcement learning framework. Force and tactile sensing can provide proprioceptive knowledge about grasping stability and success, thus regrasping or rearrangement can be activated to improve grasping success rate and versatility [37,38]. Despite the improvement in DRL-based techniques, they continue to be data-intensive, complex, and collision-prone [37,39]. These drawbacks make them unsuitable for robotic manipulations in FMCG industries, especially in the area of efficient and lightweight object manipulation [40], further research might be required to be industry-ready. In comparison, some of the data-driven approaches have been successfully applied in high-mix low-volume industries and are relatively more industry-ready [41].

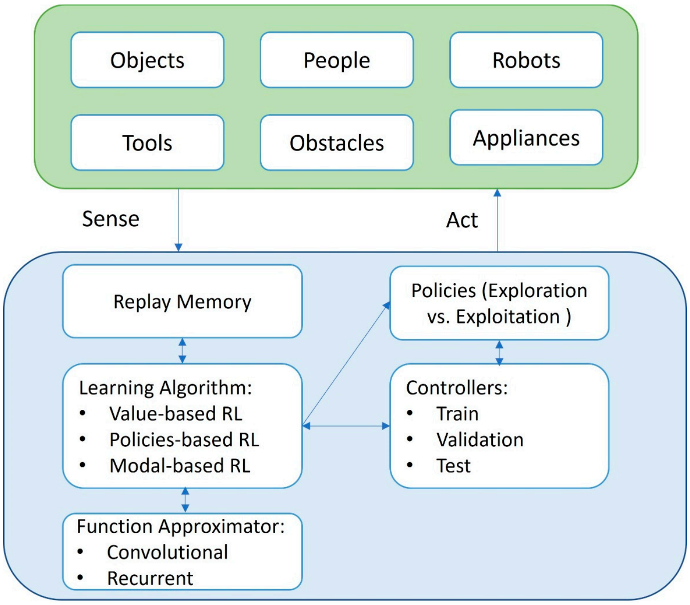


Figure 7. Deep reinforcement learning flow diagram [42].

3. Methodologies

Handling and packing mass-customized orders and products is one of the key challenges that the current warehouse or logistics industry is facing [43]. These challenges are very similar to the high-mix and high-volume problems faced by any production line across different industries currently. In existing fast-moving consumable goods packaging lines, packing various products within a delivery box is still very manual and capital-intensive [44] in order to maximize the space utilization rate and minimize the damage to products, especially when they are fragile or mishandled [45]. It is a labor-intensive process, which is a bottleneck in boosting the productivity of the e-commerce industry. Although there are existing automation and robotic solutions in the market, these solutions are developed to address either high-mix low-volume orders or low-mix high-volume orders. With more than a million products in today's consumer goods markets, achieving hyper-personalized packaging poses a huge challenge to implementing existing robotic or automation solutions. This is due to the lack of intelligent algorithms that enable these existing solutions to handle large amounts of product databases and to use the database effectively, without affecting the cycle time of high-mix product packaging.

In this work, we leverage intelligent software models such as machine learning algorithms to enable existing robotic platforms for handling the high-mix and high-volume problem that occurs in mass-personalized product packaging. The final goal of the work package is to develop a fully automated end-to-end robotic system that delivers hyper-personalized order fulfillment tasks.

Utilizing a combination of PointNet GPD and Mask R-CNN, we demonstrate versatile grasping in detail. This includes describing the results from each algorithm and investigating their advantages and disadvantages. The work that we have introduced is:

(1) Hyper-personalization line (HPL) software pipeline;
(2) ROI filtering based on Mask R-CNN;
(3) Force feedback on grasping stability and success rate;
(4) PointNetGPD grasp pose estimation using raw point cloud data.

3.1. HPL Software Pipeline

Our software system pipeline is shown in Figure 8. The vision system first acquires RGB-D images and feeds them into the perception module for object detection and point cloud segmentation. The segmented point cloud is used as the input to the grasping algorithm, which proposes and selects the best grasp candidate. In addition, the segmented point cloud can also be used for 6D pose estimation based on the surface normal of the target SKU. The detected picking pose is sent to the picking module for SKU picking and execution. The red dotted line depicts the exception-handling strategies. To be more specific, if the grasping is unstable, the grasping algorithm will put back the object on the tray and try to regrasping it again. The maximum number of regrasps is 2. The same goes for failed grasping; the grasping algorithm will reactivate the perception module again to locate the object and try to regrasp again for a maximum of 2 attempts. After the maximum attempts are reached, the system will pause and wait for human intervention. For the placing attempts, the same strategy is used to place for a maximum of 2 times with desired placing pose and no collision. Human intervention will be required if 2 consecutive failures are detected. Once the object is successfully picked up, it will be transferred to the target carton box. The picking and transfer module executes the motion planning algorithm, which searches for a collision-free trajectory to transport the SKU from the picking to the packing location. The SKU's packing position is determined by a packing algorithm that finds a feasible solution based on the geometric and space constraints of the SKU and the carton boxes. An intuitive graphical user interface (GUI) for online ordering is also developed that allows users to specify and control the bin picking and packing operation.

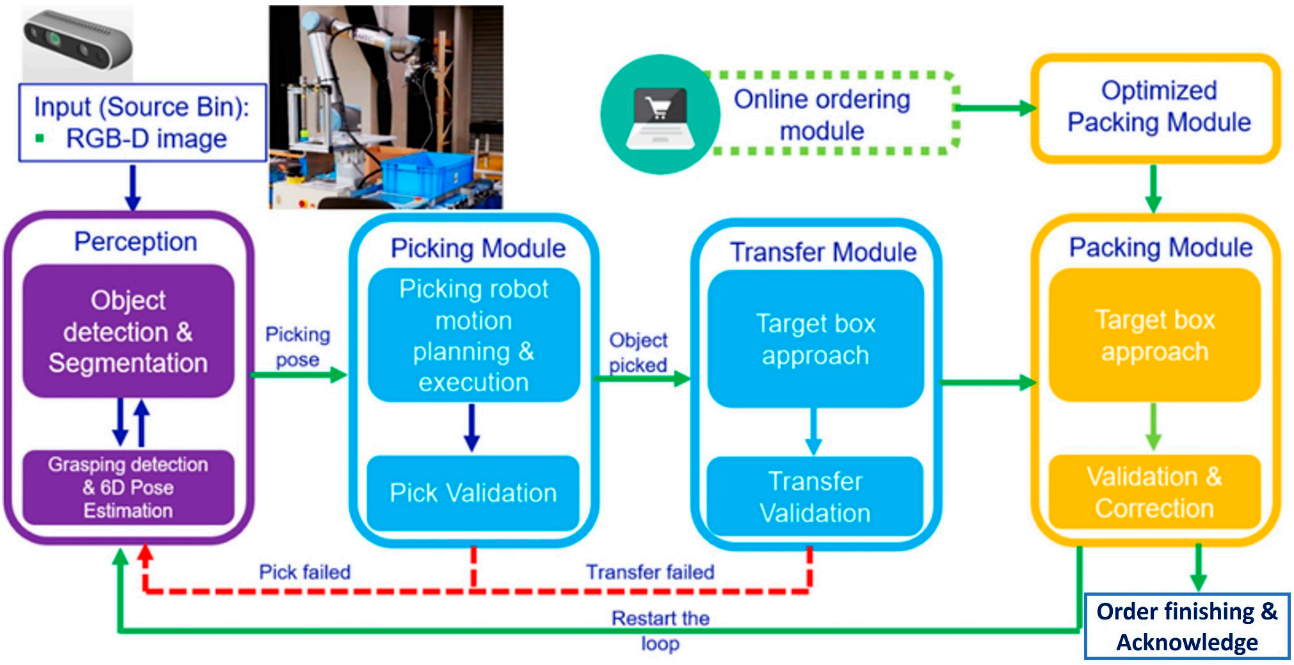
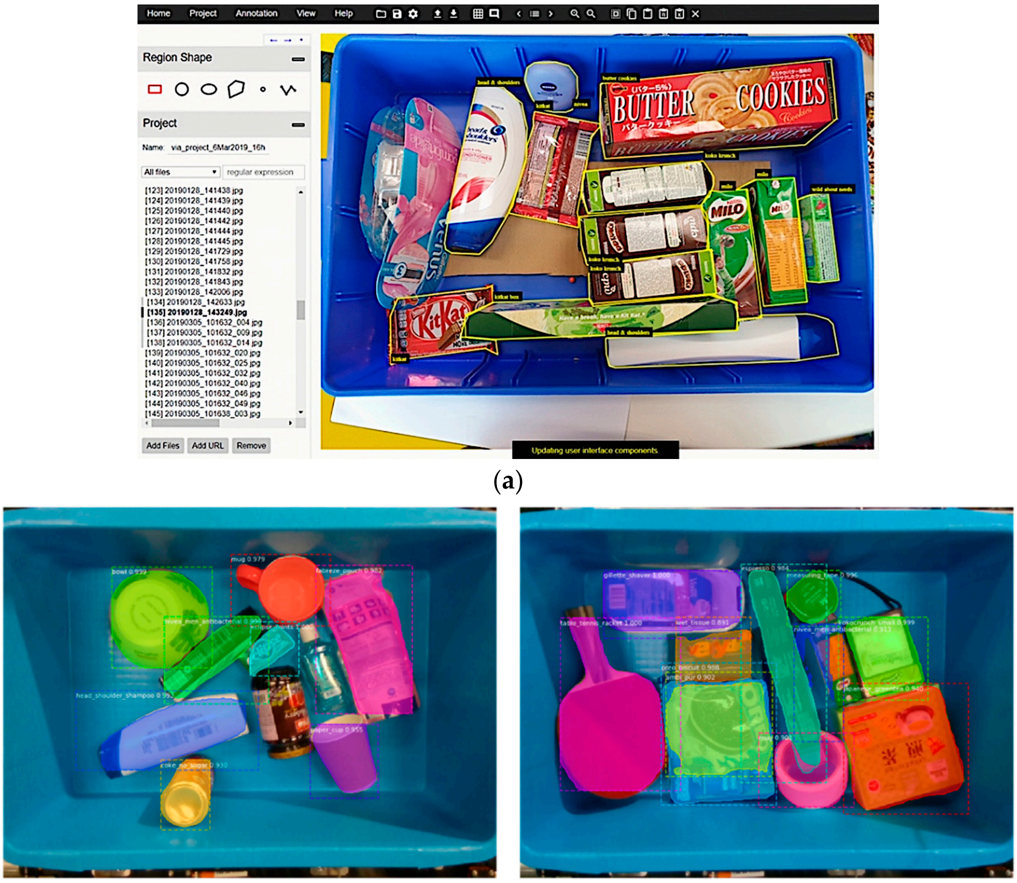


Figure 8. Hyper-personalization line pipeline.

3.2. Mask R-CNN Based Region of Interest (ROI) Filtering

For object segmentation and detection, we deploy Mask R-CNN, which is well-known as a state-of-the-art deep learning model used for instance segmentation applications [46]. The network generates output such as bounding boxes, segmentation masks, class names, and the confidence score of recognition for each object using the input RGB-D image. Mask R-CNN is a two-stage segmentation model where the input image is first scanned to generate proposals of fields possibly to contain an object and followed by classification of the proposed regions to predict the bounding boxes and masks. We use the Mask R-CNN network with weights pre-trained on the COCO baseline, which is a comprehensive object segmentation and detection dataset containing more than 80 object categories and 1.5 million object instances [47].

To customize the network to work with our SKUs, we captured and annotated our dataset for the 50 SKUs. During annotation, the SKUs are placed within a tote bin and positioned in a cluttered manner to resemble the actual working environment during model inference. An example of the image annotation process is shown in Figure 9a. In total, 200 images are collected and annotated to fine-tune the pre-trained Mask R-CNN framework using transfer learning. Based on the model training results, we can detect and recognize the SKUs with more than 97% confidence. Figure 9b,c show the model results of the SKUs placed in a cluttered environment. The target grasping object will have enough visual conspicuity to be separated from the cluttered environment [48,49]. The time it takes to train and fine-tune a machine learning model can vary depending on several factors, such as the hardware used, dataset size, and the specific training parameters. In this study, we use two NVIDIA 1080 Ti graphics processing units to do the fine-tuning and training. The number of iterations, batch size, and learning rate are all examples of training parameters that can impact training time.



Name: via_project_6Mar2019_16h

[123] 20190128_141438.jpg
[124] 20190128_141439.jpg
[125] 20190128_141440.jpg
[126] 20190128_141442.jpg
[127] 20190128_141444.jpg
[128] 20190128_141445.jpg
[129] 20190128_141729.jpg
[130] 20190128_141758.jpg
[131] 20190128_141832.jpg
[132] 20190128_141843.jpg
[133] 20190128_142006.jpg
[134] 20190128_142633.jpg
[135] 20190128_143249.Jpg
[136] 20190305_101632_004.jpg
[137] 20190305_101632_009.jpg
[138] 20190305_101632_014.jpg
[139] 20190305_101632_020.jpg
[140] 20190305_101632_025.jpg
[141] 20190305_101632_032.jpg
[142] 20190305_101632_040.jpg
[143] 20190305_101632_046.jpg
[144] 20190305_101632_049.jpg
[145] 20190305_101638_003.jpg

(**a**)

(**b**) (**c**)

Figure 9. (**a**) Data annotation of SKUs: (**b**) object detection and instance segmentation using fine-tuned Mask R-CNN in Cluttered Environment 1; (**c**) object detection and instance segmentation using fine-tuned Mask R-CNN in Cluttered Environment 2.

For the object detector, it took us around 3 h to fine-tune with around 50 objects from daily used products for 50 epochs. However, for the 3D grasping pose sampling, due to the size of the 3D point cloud data, the training and fine-tuning time significantly increased to 22 h. After the object mask is generated in the cluttered environment, we use the mask to project on the point cloud from a low-cost RGB-D camera to filter out the surrounding noise in order to obtain the region of interest (ROI) point cloud of the grasping object as can be seen in Figure 10.

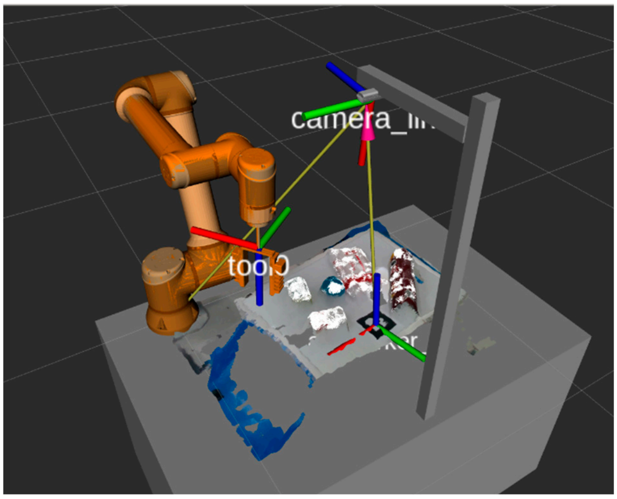


Figure 10. ROI filtering based on Mask R-CNN.

The scalability issue of object detection based on deep-learning approaches with fast-changing FMCG products is a challenging problem to solve, but there are several ways to address it. First of all, parallel computing and more powerful computers evolve over time due to Moore's Law [50]. Deep learning datasets take less time to train now compared to several years ago. Secondly, transfer learning enables reduced training time by tagging on pre-trained models of similar objects or features instead of training new datasets from scratch. Thirdly, new data can be created by data augmentation, such as rotating, adding randomness, scaling, and flipping. As a result, model performance can be enhanced by injecting randomness with shortened data collection time. Last but not least, steps can be taken to optimize the deep learning model, such as reducing the number of parameters or more efficient neural network architectures. By employing these techniques, you can improve the performance and speed of your model, making it more scalable and better suited to real-world applications.

3.3. Sensor Feedback on Grasping Success and Stability

The current gripper used in the setup is a 2-finger soft gripper configuration from the ROCHU soft robotic gripper group. The gripper is selected due to its grasping ability to suit a more diverse set of grasp objects; the contact surface of the gripper has the elasticity to slightly wrap around the targeted grasp object, which creates higher traction while grasping onto objects compared to normal rigid 2-finger grippers. The Rochu gripper is driven by pneumatics, and output air pressure can be adjusted based on the output voltage. This makes viable grasping force control possible through the UR10e Modbus.

These are the main conditions considered when we mount the force sensors on the pneumatic gripper:

- Sensor calibration;
- Actual use condition, grip stability, and slippage;
- Sensor placement for accurate position of gripping contact.

The force sensors are calibrated by measuring its raw voltage output against a calibrated load cell at multiple samples of force values; two examples of the calculated curve fitting of the sensor readings of varying characteristics are conducted. The calculated results show varying exponential curve trends among the set of sensors used, as illustrated in Figure 11. As a result, the general model $f(x) = a \cdot \exp(b \cdot x)$ is deployed to fit into the force/voltage datasets collected during experiments. Coefficients are calculated with 95% confidence bounds. After that, we chose a pair that has a similar force to voltage curve outputs for reading and visual interpretation consistency. For the force sensor on the left, the calibration fitting model is in Equation (1), while Equation (2) demonstrates the fitting model for the sensor mounted on the right finger. The force sensors were initially mounted on the pneumatic gripper with the force sensor contact area exposed for direct contact with the grasping objects to ensure that the measured force is true to the calibrated curve readings.

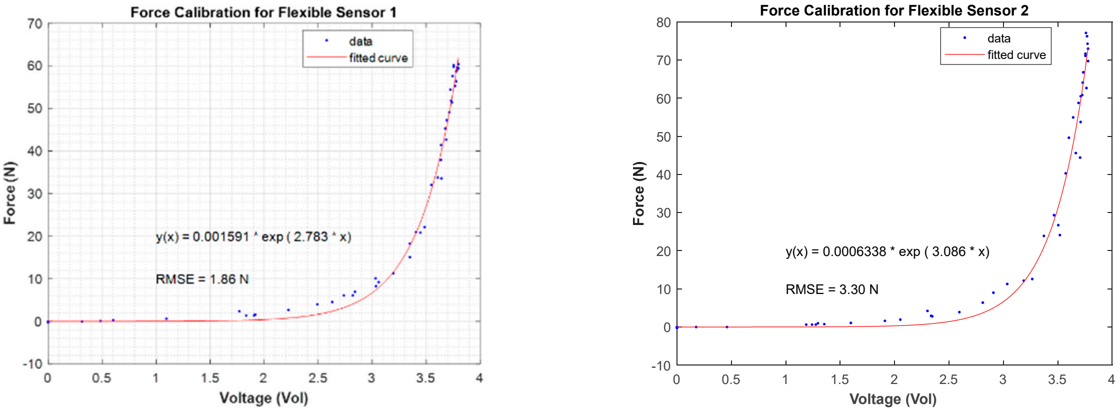


Figure 11. Force calibration for Flexible Sensors 1 and 2 (all additional resistors: 1k Ohm).

The fitting results are shown below:

$$f_{1(x)} = a_1 \cdot \mathrm{e}^{b_1 x} \tag{1}$$

where $a_1 = 0.001591$, and the upper and lower bound for a_1 is 0.00252 and 0.0006622, respectively. $b_1 = 2.283$ and the upper and lower bound for b_1 is 2.94 and 2.626, respectively. Root-mean-square deviation (RMSE) is equal to 1.86 N.

$$f_{2(x)} = a_2 \cdot \mathrm{e}^{b_2 x} \tag{2}$$

where $a_2 = 0.0006338$, and the upper and lower bound for a_2 is 0.001325 and -5.77×10^{-5} respectively. $b_2 = 3.086$ and the upper and lower bound for b_2 is 3.379 and 2.793, respectively. Root-mean-square deviation (RMSE) is equal to 3.3 N.

The sensors were positioned in different places on the gripper finger to determine the best point of contact between the gripper and the object being grasped. The first two positions, depicted on the left in Figure 12, either failed to register or only registered the grasping contact sporadically. The only position that consistently registered the grasping contact was one where the sensor was stretched across the curved surface of the gripper fingertip, as shown on the right in Figure 12.

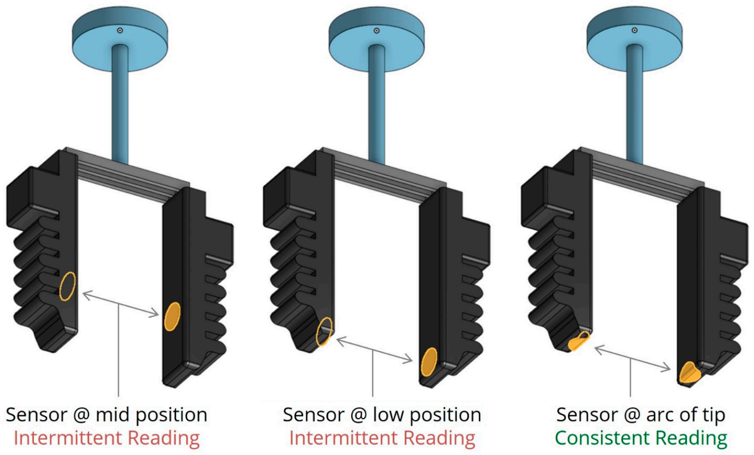


Figure 12. Different tactile sensor positions on the pneumatic gripper.

Grasping trials on all viable objects for picking, as shown in Figure 13, were conducted in a simple sequence of gripping the object (Step 1), followed by raising the object off the table (Step 2). The object is then either released to be dropped (Step 3a) or placed back on the table if the object is fragile (Step 3b).As can be seen in Figure 14, during the grasping sequence, the force sensor readings change abruptly, and we can count how many grasps based on the number of peaks occur during a certain time interval. The force sensor reading will reach a plateau after the grasping reaches a stable state. Based upon this pattern, we can train our stability evaluator based on the difference in sensor readings from both sides to decide on grasping stability. For example, when the grasping attempt is failed, we can predict the failure from the sensor readings and the robot does not need to go back to the home position and can revert back to the grasping scene to redo the grasping motion. As a result, the motion waste is greatly reduced. However, the contact plates have very low grip traction due to the flatness and rigidity of the contact surface. Objects that are grasped slip off the gripper and force sensor easily on the motion of lifting off the table surface. To increase the gripping traction, we wrapped silicone rubber (or grip) tape around the force sensor contact surface, as shown on the right of Figure 15. The silicone rubber tape dampens part of the measured contact force during the object grasping action. The rubber damping effect refers to the phenomenon of reduced sensitivity of a tactile sensor when a rubber material is placed between the sensor and the object being sensed. The effect occurs because the rubber material absorbs some of the energy from the contact force, reducing the magnitude of the signal that reaches the sensor. This affected the force value output from the grasping actions, and the readings here on were taken with the help of compensation algorithms to the force curve reading from the initial calibration. A compensation factor is calculated based on the difference between the actual contact force and the measured signal from the tactile sensor. This factor represents the reduction in sensitivity caused by the rubber damping effect. After that, the compensation factor $\mathtt{C} = 0.0176$ is applied to the sensor signal to correct for the reduced sensitivity. The corrected signal provides an estimate of the actual contact force, which is more accurate than the raw signal from the tactile sensor.

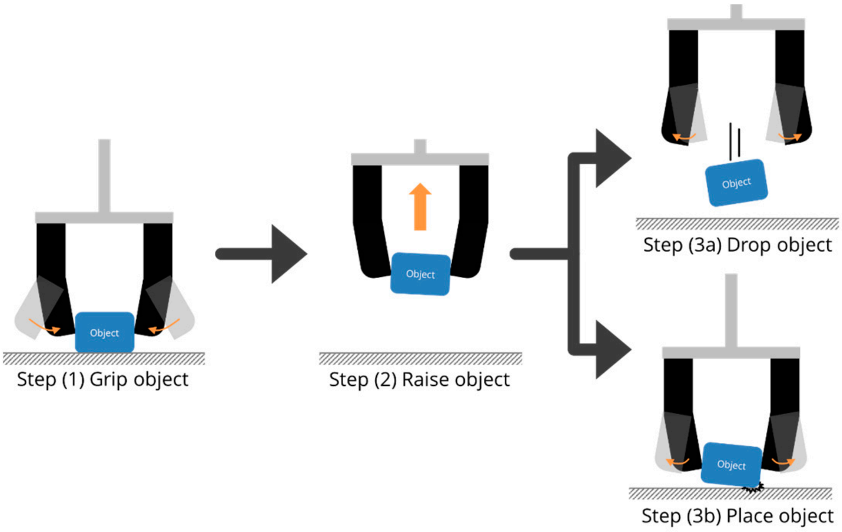


Figure 13. Grasp action sequence.

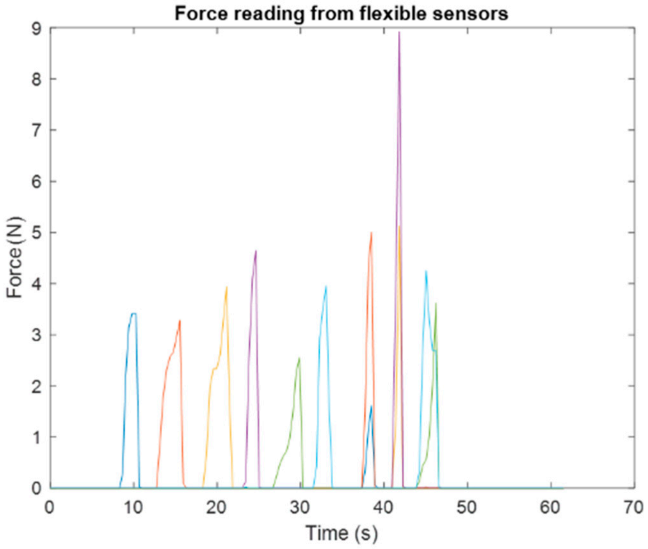


Figure 14. Pressure sensor readings during multiple grasping sequences.

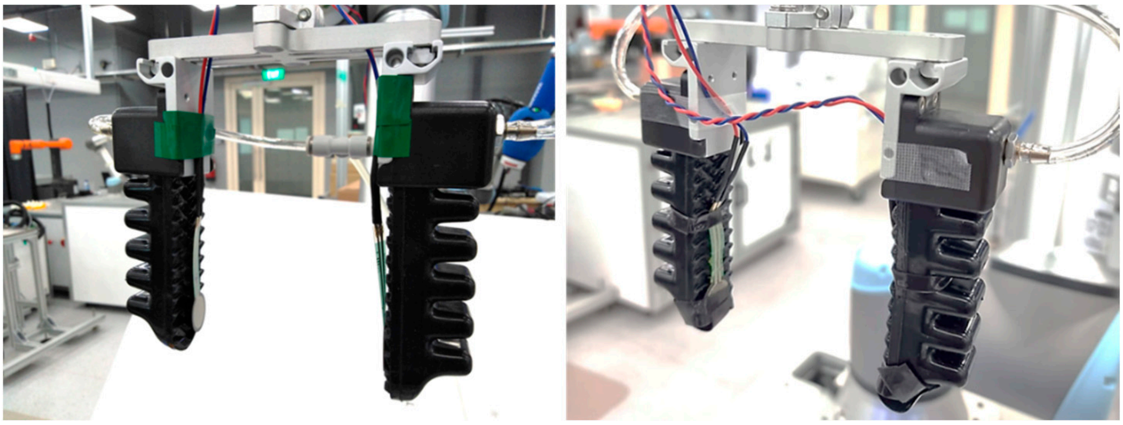


Figure 15. Force sensor on the pneumatic gripper exposed contact plate (**left**) wrapped with grip tape (**right**).

The general output readings from the force sensor are shown in Figure 16; results shown on the left are from objects that follow Steps 1, 2, and 3a (dropped) and on the right from objects that follow Steps 1, 2, and 3b (placed back on the table).

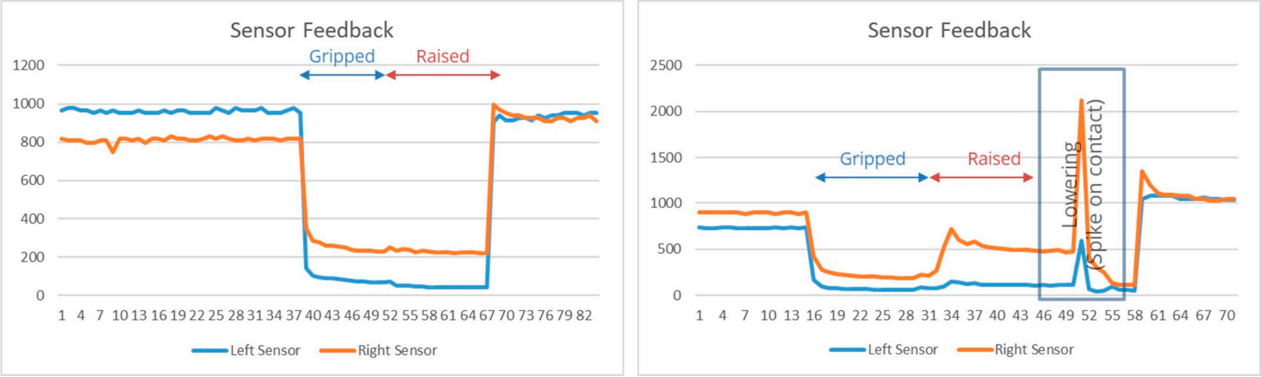


Figure 16. Force sensor reading from object grasping Actions. gripped → raised → dropped (**left**), gripped → raised → placed (**right**).

The voltage drop is measured from the force sensors as contact with the grasped objects occurs. Seldom occurrence of slippage and misalignment of the force sensor contact point with the grasp objects is still present. Tabulating the results across the trials, the current grasp detection is determined to be positive as long as there is a voltage drop of 20% present in either of the mounted tactile sensors.

3.4. Robotic Grasping Based on PointNetGPD

To improve the grasping performance, we use a grasping algorithm with sensor feedback that is capable of estimating the optimal grasping pose straight from the raw point cloud. An unprocessed raw RGB-D point cloud is used as input. The grasp sampling evaluation network is capable of analyzing the complex mathematical representation of the point of contact between the grasping object and the EOAT, and a sparse point cloud is present in particular. Low-cost RGB-D cameras can be selected to provide the inputs such as the Realsense D435i camera or Zivid camera.

Usually, in grasp detection, we have to find out the object's grasping position and orientation. Given a specified object, g = (p,r) ∈R^6 means 3D grasping configuration, where r = (rx, ry, rz) and p = (x, y, z) denote the orientation and position of the gripping tool, respectively. There are several grasping candidates generated from the deep learning network; in this case, we need to evaluate which candidate is of higher quality [4]. PointNetGPD provides a network architecture and grasp representation. In this network, PointNet takes the unified grasp represented by the RGB-D point cloud after filtering out the point cloud that falls outside the gripper enclosure. Potential local sub-optimal or optimal grasping candidates will be selected based on the highest scores from the evaluation metrics.

Machine learning algorithms can be used to rank new grasping poses by training a model on a set of labeled examples of successful and unsuccessful grasps, along with the corresponding features such as object shape, size, material properties, and the position and orientation of the gripper. Once the model is trained, it can be used to rank new grasping poses by analyzing the features of the grasp and the object and predicting the likelihood of a successful grip with higher scores indicating higher confidence in the grip.

There are several machine learning algorithms that can be used for grasping pose ranking, including supervised learning algorithms such as decision trees, random forests, and support vector machines, as well as unsupervised learning algorithms such as k-means clustering and principal component analysis. In this paper, rather than using a machine learning approach to evaluate the quality of an attempted grasp, a physics-based method is here preferred based on the force closure as a result of a large amount of labeled data and careful feature selection required for the machine-learning-based approaches.

The optimal grasping pose selection metric is tagged on the force enclosure metrics. Equations (3) and (4) describe the calculation of the force and torque of the end effector of a parallel gripper mounted on UR10e, respectively.

$$f_{ee} = J'_{robot} \cdot f_{gripper} \tag{3}$$

$$T_{ee} = J'_{robot} \cdot cross\Big(f_{gripper},\ p_{gripper} + [-;0;0] - f_{fk}(robot,\ q)\Big); \tag{4}$$

where J_{robot} represents the end-effector Jacobian of the robot, and f_{fk} is the forward kinematics of the robot, while s means the gripper opening distance. f_{ee} and T_{ee} stand for end effector force and torque, respectively, and $f_{gripper}$ denotes the gripping force.

The final force enclosure is calculated in Equation (5).

$$force\_encl - 1 - (norm(-f_{ee})\hat{}2 + norm(-T_{ee})\hat{}2)/(2 \cdot m \cdot g \cdot \theta); \tag{5}$$

where m represents grasping object mass, and θ represents gripper orientation.

This approach is mainly to eliminate the ambiguity caused by the different experiment (especially camera) settings; specifically, for the gripper approaching, the parallel and orthogonal directions of the gripper as the XYZ axes, respectively, while the origin is located at the bottom center of the gripper. After obtaining the smaller and segmented point cloud, an N point is passed through the proposed evaluation networks. This method is lightweight with approximately 1.6 million parameters that outperformed other CNN-based grasp quality evaluation networks [51].

The PointNetGPD network is trained using the YCB dataset, which contains common objects that can be found in our day-to-day activities. In our algorithm evaluation phase, our robotic setup consists of an Intel RealSense D435 camera and a UR10 robot with a soft gripper. We have tested the grasping algorithm performance using cluster scenes. Figure 17a shows the results of the grasping pose generation from PointNetGPD. It generates several grasping candidates, as indicated by the green gripper poses. Lastly in Figure 17b, the selected red grasping pose possesses the highest grasping quality among all the grasping candidates.

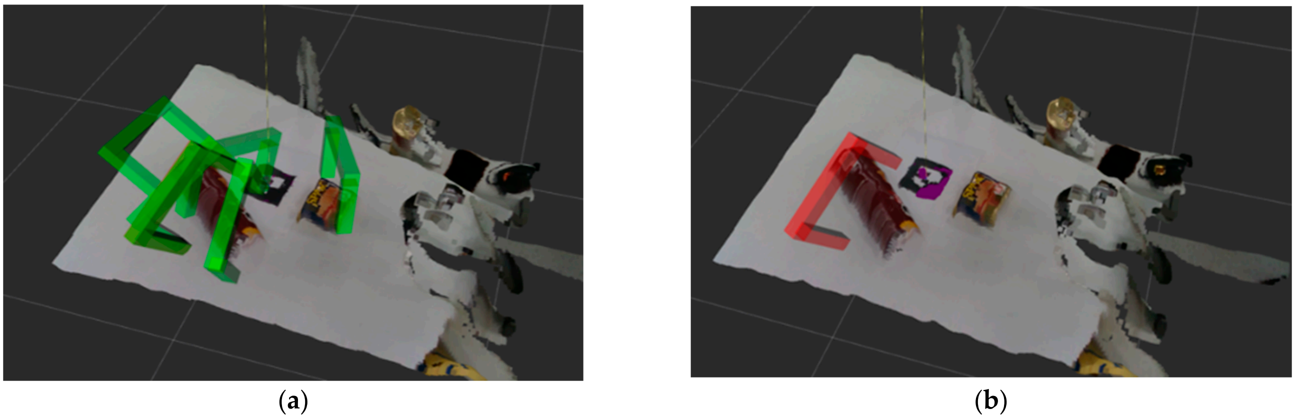


Figure 17. Simulation results in ROS1 environment: (**a**) grasping pose candidates generated from PointNetGPD; (**b**) grasping pose selected with the best grasping quality.

3.5. Adaptive Anti-Slip Force Control

In order to improve grasping stability and accuracy, the grasping force can be adjusted to prevent the dropping of grasping objects and maintain stability even in challenging scenarios. Moreover, the grasping force can be controlled by the input pneumatic pressure, as mentioned above. The material information of FMCG products can be predicted via Mask R-CNN. After that, the initial force is pre-set by material type, and the estimated weight profile can be looked up in the database. For instance, if the object is detected to be Campbell's soup can, it will show the material as aluminum, and its weight is estimated at around 300 g in our online database. If the object is made of paper and is lightweight, the initial grasping force will be reduced to prevent damage to the packaging.

Due to the sensor only giving 1D force information, in this phase, the slip detection is based on heuristic considerations. Once the slippage is detected, the grasping force will increase by 30% to prevent slippage from happening. By adopting this method, we reduce grasping slipping by 43%; however, due to the limitation of a 1D sensor, such as being unable to detect contact area, point of contact, and 2D strain, this is the best we can get from a 1D force sensor.

For slip detection, the inputs are force sensor measurements on the robot's gripper, and the output is a decision about whether or not the object is slipping. The benefit of the heuristic method is its capability to handle uncertain and non-linear relationships between inputs and outputs even when the data are noisy and sparse. The crisp outputs for heuristic considerations include slipping, stable grasping, and non-contact. Anti-slip extra force is only activated when slipping is detected. The graph for slip detection based on the heuristic method is shown in Figure 18. It shows 6 grasping sequences for the same grasping object. During the grasping motion, the force drops for two of the grasping scenarios, which activates the slip detection module, and an extra 30% of the force will be applied to the pneumatic grippers to counter the slipping. The other three scenarios in the middle depict the failed grasping scenes where the force is not enough to lift up the object, thus the anti-slip module is not activated during these three failed scenarios.

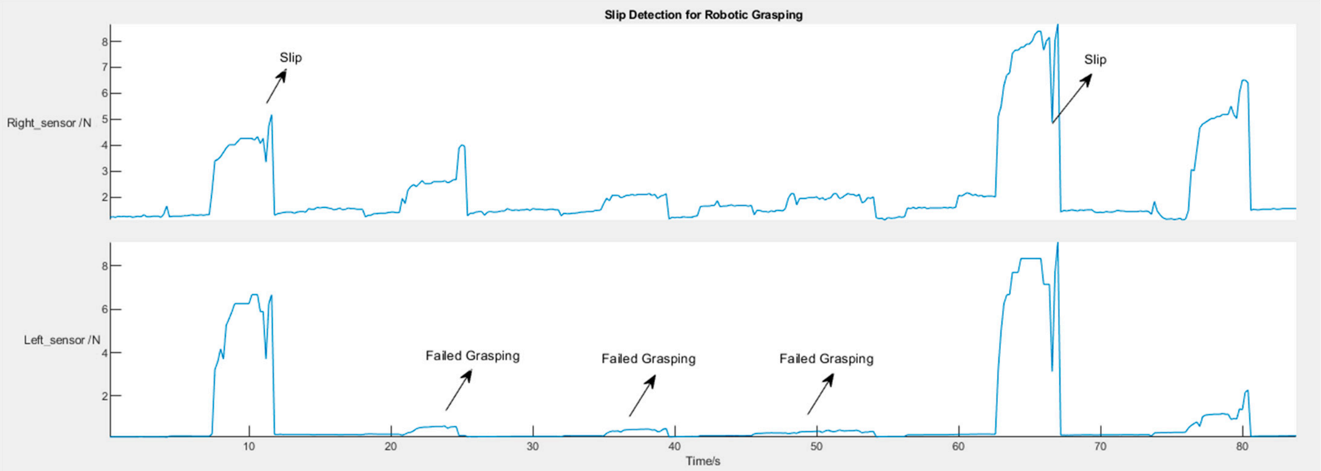
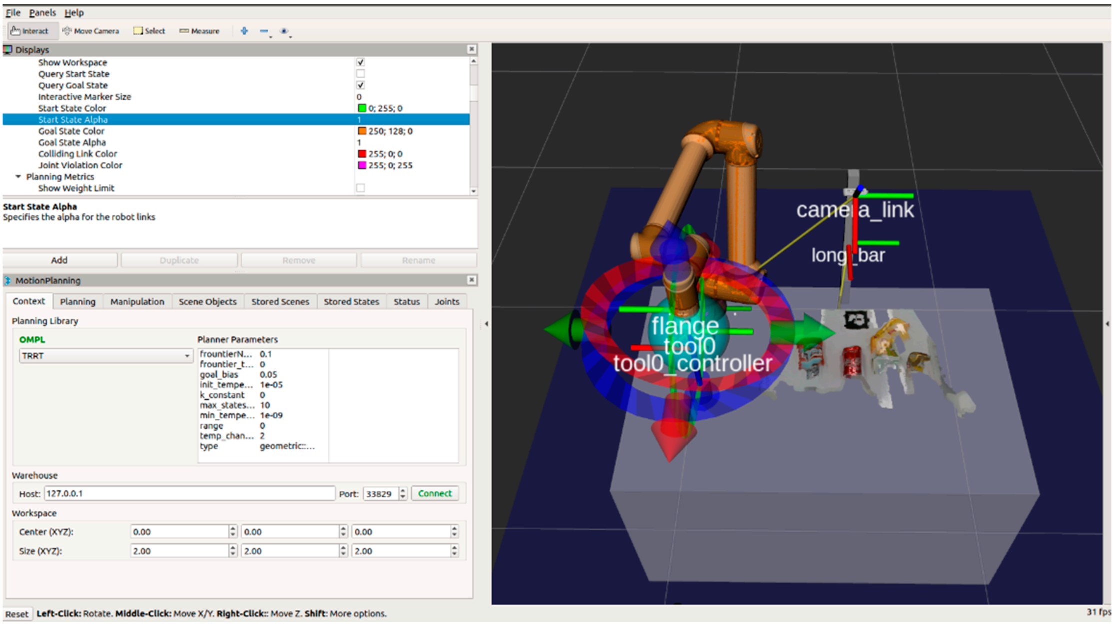


Figure 18. Force sensor reading from object grasping actions.

3.6. *Trajectory Planning*

The destination EOAT pose is selected by the DL-based sampling algorithm. After that, the inverse kinematics (Ik) solver is called to calculate the 6 joint values in the joint space. In this research, we use IKFast in ROS MoveIt. As demonstrated in Figure 19, the Open Motion Planning Library (OMPL) [52] planner is a sampling-based motion planning library that contains a lot of state-of-the-art motion planners such as LazyRRT [53], probabilistic roadmaps (PRM *) [54], single-query bi-directional lazy collision checking planner (SBL) [55], kinematic planning by interior–exterior cell exploration (KPIECE) [56], etc. Moreover, it also offers ROS plugins. Transition-based rapidly exploring random tree (TRRT) is deployed to calculate the optimal collision-free trajectory with the shortest traveling distance. The global bias is set at 0.04.



Show Workspace	☑
Query Start State	□
Query Goal State	☑
Interactive Marker Size	0
Start State Color	🟩 0; 255; 0
Start State Alpha	1
Goal State Color	🟧 250; 128; 0
Goal State Alpha	1
Colliding Link Color	🟥 255; 0; 0
Joint Violation Color	🟪 255; 0; 255
Planning Metrics	
Show Weight Limit	□

Start State Alpha
Specifies the alpha for the robot links

Planning Library

OMPL: TRRT

Planner Parameters

frountierN...	0.1
frountier_t...	0
goal_bias	0.05
init_tempe...	1e-05
k_constant	0
max_states...	10
min_tempe...	1e-09
range	0
temp_chan...	2
type	geometric::...

Warehouse

Host: 127.0.0.1 Port: 33829

Workspace

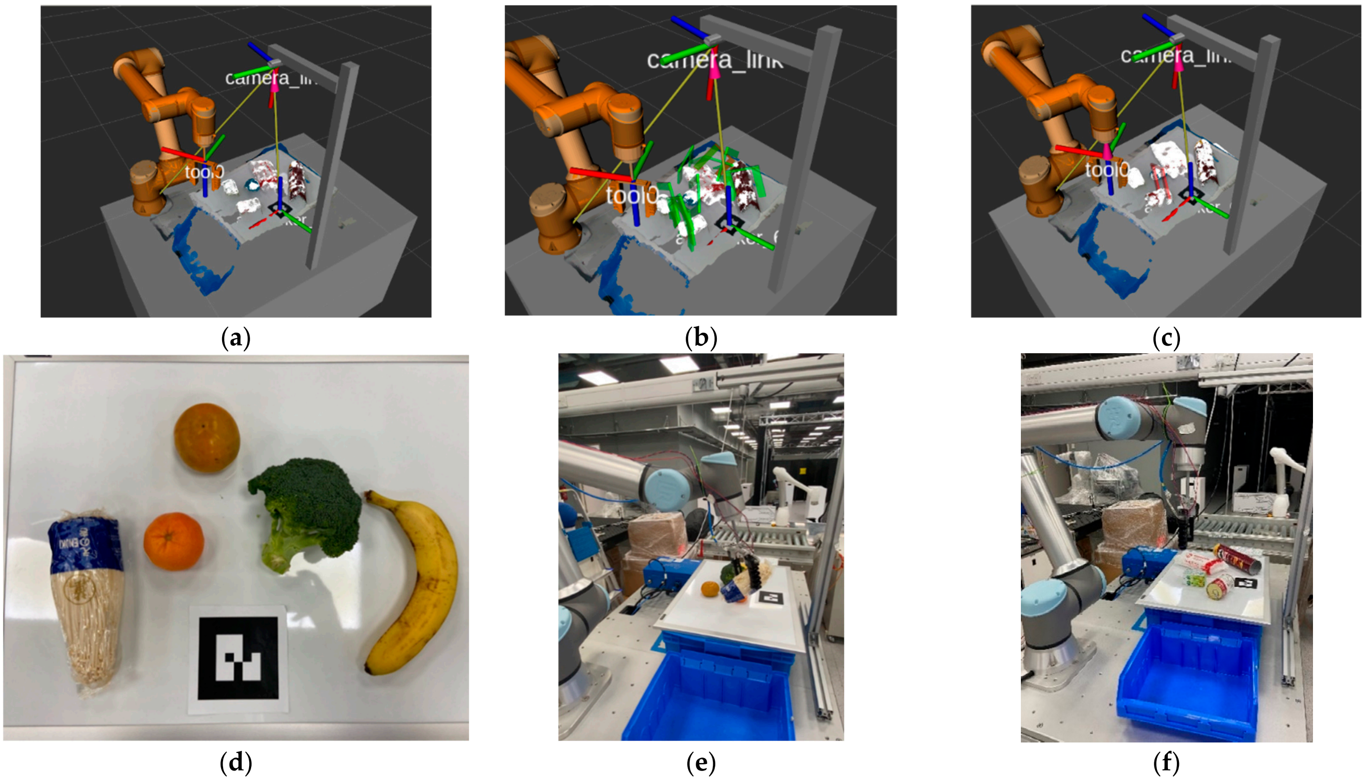
Center (XYZ):	0.00	0.00	0.00
Size (XYZ):	2.00	2.00	2.00

Left-Click: Rotate. **Middle-Click:** Move X/Y. **Right-Click::** Move Z. **Shift:** More options.

Figure 19. OMPL path planner in ROS.

4. Result Analysis

Figure 20a shows our experimental setup. An Intel RealSense D435i camera that is mounted on a camera fixture produces a point cloud representation of the scene. The UR10e robot and Rochu two-finger soft gripper realize object manipulation using robotic grasping. A Linux workstation acts as the centerpiece of communication driving the UR10e robot through Ethernet connection and ROS middleware, Realsense Camera via USB port, and controlling the I/O of the soft gripper via RTDE socket. A point cloud noise removal filter was added to reduce the computation load, as shown in Figure 20a. All the grasping poses were simulated and validated in ROS MoveIt before sending to the real robot for execution, as shown in Figure 20b. Retraining of the grasping algorithm based on the selected soft gripper configuration is carried out on the Linux computer. In our experiments, the UR robot will pick the objects from the table and place them into the blue bin [6]. The objects are placed on a whiteboard to imitate conveyor picking due to lab space constraints. In the upcoming phase, an actual conveyor belt-picking demonstration will be presented.



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 20. Experiment results for daily used products: (**a**) ROS simulation environment; (**b**) grasping pose candidates by sampling; (**c**) final quality grasping pose with the best scores; (**d**) grasping trials on fragile fruits and vegetables objects; (**e**) Enoki grasping; (**f**) grasping on FMCG-selected objects.

Figure 20c shows the grasping pose generation and decision-making process based on the metrics we have pre-defined in the core algorithm. Using the filtered object point cloud, PointNet GPD randomly samples grasping poses for different objects and determines the best grasping pose after evaluating the combination of grasping quality metrics, including force closure and grasping wrench space (GWS). With our current system, the grasping pose detection and robot execution take approximately 10–15 s and 5 s, respectively. In future development, we plan to accelerate our entire grasping pose detection process through the adoption of multiple computation units. Figure 20f shows one of the instances of our robotic grasping in action.

The soft gripper is capable of handling fragile and deformable products such as fruit, vegetables, and FMCG products in general. Since these grippers are deformable, they can handle the products such as fruits and vegetables, which have inconsistent dimensions quite well. Using a rigid gripper such as the Robotiq gripper may not be practical, and it can cause surface damage or induce irreversible deformation on fragile products. Figure 20d,e shows two scenarios in which non-textured and textured objects are randomly placed on the surface. Subsequently, the robotic system performs the grasping action based on the checkpoint that we obtained from the training. We conducted around 120 grasping trials each involving novel and unknown objects using selected FMCG products and achieved approximately 81% grasping success rate using the Rochu soft gripper, as shown in Figure 21 and Table 3. The grasping success rate for unfamiliar objects is around 10% lower than for grasping familiar products. Moreover, grasping with stability evaluation and anti-slip control increase the grasping success rate significantly. Part of the grasping failures can be attributed to the limited payload of the Rochu gripper. Generally, the DL-based grasping pose detection is stable and accurate, with the grasping pose detection rate reaching above 90% for PointNetGPD. This is a significant improvement over the previous DRL methods, which report a 65% grasping success rate. In general, deep-learning-based systems are computationally intensive and require high-performance hardware to run efficiently. PointNetGPD is no exception, as it involves processing large amounts of 3D point cloud data, which can be computationally expensive. However, recent advancements in hardware, such as GPUs and TPUs, have made it possible to run deep-learning-based systems faster and more efficiently. Moreover, computation time also relates to input point cloud size. In conclusion, the speed and computation time of PointNetGPD and visual push grasping are dependent on various factors and may vary depending on the hardware configuration and the size of the input point cloud.

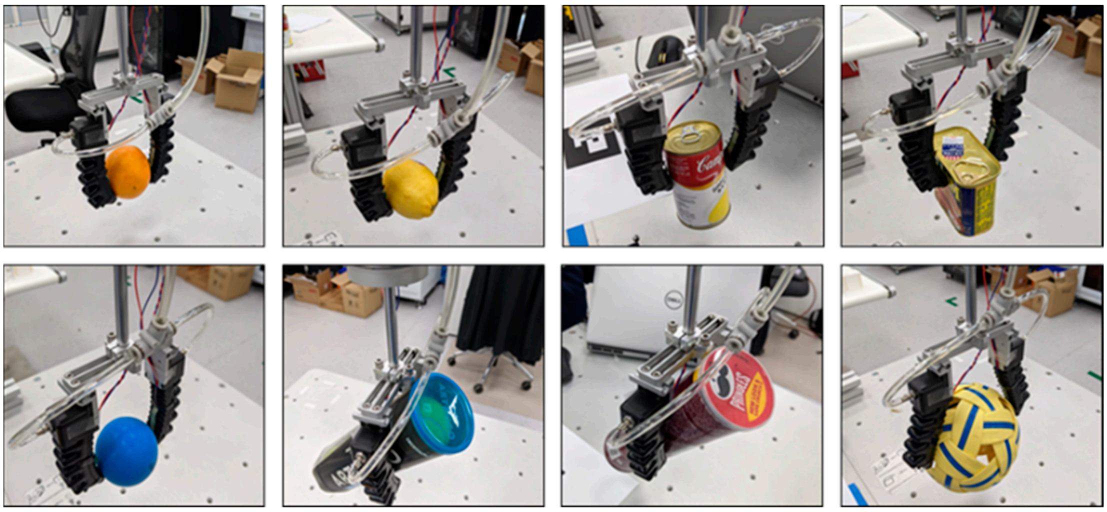


Figure 21. Grasping experiments for FMCG products.

Table 3. Results from grasping experiments.

Grasping Methods	Total Attempts	Failure	2nd Attempts	Success Rate	Computation Time
Soft grasping on familiar products with anti-slip control (FMCG)	120	38	15	80.8%	11.5 s
Soft grasping on familiar products without anti-slip control (FMCG)	120	54	12	65%	11.7 s
Soft grasping on unfamiliar products with anti-slip control (veggies and fruit)	120	51	17	70.8%	13.2s
Visual pushing grasping (grasping)	120	62	19	64.2%	17.8 s

All the objects here are within the gripper grasping size limitation and weight limitation. We include both familiar and non-familiar objects for the success rate study. The primary failure causes during this grasping experiment include, firstly, inaccurate grasping pose estimation that either causes robot collision or not enough contact to lift the object. Secondly, the object slips from the gripper due to weak friction, especially if the object weight is close to the weight capacity limit. Thirdly, the cluttered grasping scene can obstruct the smooth grasping process as the camera is in a fixed position, thus causing the grasping failure. In summary, the top three main failure causes are learning-based perception errors, grasping scenes, and physics.

5. Future Works

One-axis FSR resistive sensor can only provide force information on the vertical plane, thus information is lost on the tangent plane. As a result, the adoption of a one-axis force sensor will cause certain limitations when deciding the grasping stability. In the next phase, one of our key focuses is to upgrade the force sensor to a two-axis or more tactile sensor, which can provide not only two-axis deformation information on the contact plane but also the point of contact. One of the options is DIGIT tactile sensors developed by Facebook AI and Gelsight [17], as can be seen in Figure 22. Due to the grasping nature, time-series [57] neural networks can be adopted to analyze the grasping stability and provide inputs for the grasping sampling. After that, more data can be collected and used for training an advanced adaptive force control algorithm that can detect slip during the grasping motion in order to increase the grasping stability.

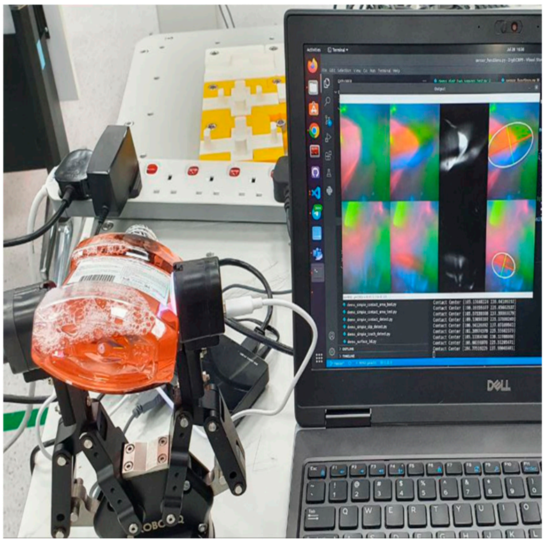


Figure 22. DIGIT sensor with Robotiq 2f-85.

Moreover, the current grasping selection is based on the initial scene state where the desired placing pose is not taken into consideration. This is only applicable in scenarios where the products have a high tolerance for collision. This will, however, further pose limitations if the placing pose is important in scenarios such as fragile product handling and collision-prone packaging. In the next phase, placing pose [58] will be considered as a constraint in the evaluation metrics when selecting the best grasping pose. Furthermore, we are also looking to further modularize and simplify the grasping module in the next step [59]. Moreover, cloud deployment will also be studied in our next phase, along with continuity of accuracy and speed enhancement.

6. Conclusions

In this study, we first conducted a review of the soft gripper, tactile sensing technology, and data-driven and deep-learning-based grasping pose detection methods. After that, we proposed end-to-end HPL grasping solutions for high-mix high-volume scenarios. To be more specific, we proceed with a data-driven approach together with the Rochu soft gripper and resistor-based force sensors mounted on the fingertips. The PointNet GPD algorithm is chosen as the core algorithm due to its advantages in raw point cloud processing and large numbers of grasping databases. In addition, we retrain the PointNet GPD with our own gripper configuration and anti-slip control algorithm. The results show that the PointNet GPD performs quite well in terms of grasping pose generation for familiar and unknown objects. In terms of grasping rate, we achieve around 81% for chosen FMCG products and 71% for unfamiliar fruit and vegetables, even when faced with the limitation of the payload of the soft gripper.

Future works involve software optimization for ease of deployment on cloud-based services and upgrade from a one-axis force sensor to Facebook DIGIT two-axis tactile sensing. Moreover, a Recurrent neural network (RNN) based time-series grasping evaluation network is also developed in our next phase. Working in parallel, further improvement in terms of grasping speed and accuracy will be addressed as well.

Author Contributions: Conceptualization, Z.X., J.Y.S.C. and G.W.L.; methodology, Z.X., J.Y.S.C. and G.W.L.; software, Z.X., F.B., G.W.L. and J.Y.S.C.; validation, Z.X., J.Y.S.C. and G.W.L.; formal analysis, Z.X.; investigation, Z.X., F.B., G.W.L. and J.Y.S.C.; resources, Z.X., J.Y.S.C. and G.W.L.; writing—original draft preparation, Z.X.; writing—review and editing, Z.X., G.W.L. and J.Y.S.C.; supervision, Z.X.; project administration, Z.X.; funding acquisition, Z.X. and F.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Advanced Remanufacturing and Technology Center under Project No. C21-11-ARTC for AI-Enabled Versatile Grasping, grant number SC29/22-3005CR-ACRP.

Data Availability Statement: Data are contained within the article.

Acknowledgments: This research was supported by the Advanced Remanufacturing and Technology Center under Project No. C21-11-ARTC for AI-Enabled Versatile Grasping. We thank our colleagues and ARTC industry members who provided insight and expertise that greatly assisted the research.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bormann, R.; de Brito, B.F.; Lindermayr, J.; Omainska, M.; Patel, M. Towards automated order picking robots for warehouses and retail. In Proceedings of the Computer Vision Systems: 12th International Conference, ICVS 2019, Thessaloniki, Greece, 23–25 September 2019; pp. 185–198.
2. Xie, Z.; Somani, N.; Tan, Y.J.S.; Seng, J.C.Y. Automatic Toolpath Pattern Recommendation for Various Industrial Applications based on Deep Learning. In Proceedings of the 2021 IEEE/SICE International Symposium on System Integration (SII), Narvik, Norway, 8–12 January 2012; pp. 60–65.
3. Zhen, X.; Seng, J.C.Y.; Somani, N. Adaptive Automatic Robot Tool Path Generation Based on Point Cloud Projection Algorithm. In Proceedings of the 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Zaragoza, Spain, 10–13 September 2019; pp. 341–347.
4. Liang, H.; Ma, X.; Li, S.; Görner, M.; Tang, S.; Fang, B.; Sun, F.; Zhang, J. Pointnetgpd: Detecting grasp configurations from point sets. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 3629–3635.
5. Zeng, A.; Song, S.; Welker, S.; Lee, J.; Rodriguez, A.; Funkhouser, T. Learning synergies between pushing and grasping with self-supervised deep reinforcement learning. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 4238–4245.
6. Xie, Z.; Seng, J.C.Y.; Lim, G. AI-Enabled Soft Versatile Grasping for High-Mixed-Low-Volume Applications with Tactile Feedbacks. In Proceedings of the 2022 27th International Conference on Automation and Computing (ICAC), Bristol, UK, 1–3 September 2022; pp. 1–6.
7. Xie, Z.; Liang, X.; Roberto, C. Learning-based Robotic Grasping: A Review. *Front. Robot. AI* **2023**, *10*, 1038658. [CrossRef] [PubMed]
8. Cao, H.; Fang, H.-S.; Liu, W.; Lu, C.J.I.R.; Letters, A. Suctionnet-1billion: A large-scale benchmark for suction grasping. *IEEE Robot. Autom. Lett.* **2021**, *6*, 8718–8725. [CrossRef]

9. Jiang, Y.; Moseson, S.; Saxena, A. Efficient grasping from rgbd images: Learning using a new rectangle representation. In Proceedings of the 2011 IEEE International conference on robotics and automation, Shanghai, China, 9–13 May 2011; pp. 3304–3311.
10. Goldfeder, C.; Ciocarlie, M.; Dang, H.; Allen, P.K. The columbia grasp database. In Proceedings of the 2009 IEEE International Conference on Robotics and Automation, Kobe, Japan, 12–17 May 2009; pp. 1710–1716.
11. Chao, Y.-W.; Yang, W.; Xiang, Y.; Molchanov, P.; Handa, A.; Tremblay, J.; Narang, Y.S.; Van Wyk, K.; Iqbal, U.; Birchfield, S. DexYCB: A benchmark for capturing hand grasping of objects. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9044–9053.
12. Mahler, J.; Matl, M.; Liu, X.; Li, A.; Gealy, D.; Goldberg, K. Dex-net 3.0: Computing robust vacuum suction grasp targets in point clouds using a new analytic model and deep learning. In Proceedings of the 2018 IEEE International Conference on robotics and automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 5620–5627.
13. Mousavian, A.; Eppner, C.; Fox, D. 6-dof graspnet: Variational grasp generation for object manipulation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 2901–2910.
14. Fang, H.; Fang, H.-S.; Xu, S.; Lu, C.J.a.p.a. TransCG: A Large-Scale Real-World Dataset for Transparent Object Depth Completion and Grasping. *IEEE Robot. Autom. Lett.* **2022**, *7*, 7383–7390. [CrossRef]
15. Shintake, J.; Cacucciolo, V.; Floreano, D.; Shea, H.J.A.m. Soft robotic grippers. *Adv. Mater.* **2018**, *30*, 1707035. [CrossRef] [PubMed]
16. Goncalves, A.; Kuppuswamy, N.; Beaulieu, A.; Uttamchandani, A.; Tsui, K.M.; Alspach, A. Punyo-1: Soft tactile-sensing upper-body robot for large object manipulation and physical human interaction. In Proceedings of the 2022 IEEE 5th International Conference on Soft Robotics (RoboSoft), Edinburgh, UK, 4–8 April 2022; pp. 844–851.
17. 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 Robot. Autom. Lett.* **2020**, *5*, 3838–3845. [CrossRef]
18. Fishel, J.A.; Loeb, G.E. Sensing tactile microvibrations with the BioTac—Comparison with human sensitivity. In Proceedings of the 2012 4th IEEE RAS & EMBS international conference on biomedical robotics and biomechatronics (BioRob), Rome, Italy, 24–27 June 2012; pp. 1122–1127.
19. Mittendorfer, P.; Cheng, G.J.I.T.o.r. Humanoid multimodal tactile-sensing modules. *IEEE Trans. Robot.* **2011**, *27*, 401–410. [CrossRef]
20. Cavallo, A.; Costanzo, M.; De Maria, G.; Natale, C. Modeling and slipping control of a planar slider. *Automatica* **2020**, *115*, 108875. [CrossRef]
21. Costanzo, M.; Maria, G.D.; Natale, C. Slipping Control Algorithms for Object Manipulation with Sensorized Parallel Grippers. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 7455–7461.
22. Huang, S.-J.; Chang, W.-H.; Su, J.-Y. Intelligent robotic gripper with adaptive grasping force. *Int. J. Control. Autom. Syst.* **2017**, *15*, 2272–2282. [CrossRef]
23. Deng, Z.; Jonetzko, Y.; Zhang, L.; Zhang, J. Grasping Force Control of Multi-Fingered Robotic Hands through Tactile Sensing for Object Stabilization. *Sensors* **2020**, *20*, 1050. [CrossRef]
24. Fischinger, D.; Vincze, M.; Jiang, Y. Learning grasps for unknown objects in cluttered scenes. In Proceedings of the 2013 IEEE international conference on robotics and automation, Karlsruhe, Germany, 6–10 May 2013; pp. 609–616.
25. Schmidt, P.; Vahrenkamp, N.; Wächter, M.; Asfour, T. Grasping of unknown objects using deep convolutional neural networks based on depth images. In Proceedings of the 2018 IEEE international conference on robotics and automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 6831–6838.
26. Levine, S.; Pastor, P.; Krizhevsky, A.; Ibarz, J.; Quillen, D. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int. J. Robot. Res.* **2017**, *37*, 421–436. [CrossRef]
27. Varley, J.; Weisz, J.; Weiss, J.; Allen, P. Generating multi-fingered robotic grasps via deep learning. In Proceedings of the 2015 IEEE/RSJ international conference on intelligent robots and systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 4415–4420.
28. Ma, B.; Li, X.; Xia, Y.; Zhang, Y. Autonomous deep learning: A genetic DCNN designer for image classification. *Neurocomputing* **2020**, *379*, 152–161. [CrossRef]
29. Xie, Z.; Zhong, Z.W. Unmanned Vehicle Path Optimization Based on Markov Chain Monte Carlo Methods. *Appl. Mech. Mater.* **2016**, *829*, 133–136. [CrossRef]
30. Bohg, J.; Morales, A.; Asfour, T.; Kragic, D. Data-Driven Grasp Synthesis—A Survey. *IEEE Trans. Robot.* **2014**, *30*, 289–309. [CrossRef]
31. Fang, H.-S.; Wang, C.; Gou, M.; Lu, C. Graspnet-1billion: A large-scale benchmark for general object grasping. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11444–11453.
32. Rajeswaran, A.; Kumar, V.; Gupta, A.; Vezzani, G.; Schulman, J.; Todorov, E.; Levine, S.J.a.p.a. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv* **2017**, arXiv:1709.10087.
33. Wu, B.; Akinola, I.; Allen, P.K. Pixel-attentive policy gradient for multi-fingered grasping in cluttered scenes. In Proceedings of the 2019 IEEE/RSJ international conference on intelligent robots and systems (IROS), Macau, China, 3–8 November 2019; pp. 1789–1796.

34. Liu, R.; Nageotte, F.; Zanne, P.; de Mathelin, M.; Dresp-Langley, B. Deep Reinforcement Learning for the Control of Robotic Manipulation: A Focussed Mini-Review. *Robotics* **2021**, *10*, 22. [CrossRef]
35. Kleeberger, K.; Bormann, R.; Kraus, W.; Huber, M.F. A survey on learning-based robotic grasping. *Curr. Robot. Rep.* **2020**, *1*, 239–249. [CrossRef]
36. Kopicki, M.; Detry, R.; Schmidt, F.; Borst, C.; Stolkin, R.; Wyatt, J.L. Learning dexterous grasps that generalise to novel objects by combining hand and contact models. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–07 June 2014; pp. 5358–5365.
37. Bhagat, S.; Banerjee, H.; Ho Tse, Z.T.; Ren, H. Deep reinforcement learning for soft, flexible robots: Brief review with impending challenges. *Robotics* **2019**, *8*, 4. [CrossRef]
38. Wu, B.; Akinola, I.; Varley, J.; Allen, P. Mat: Multi-fingered adaptive tactile grasping via deep reinforcement learning. *arXiv* **2019**, arXiv:1909.04787.
39. Mohammed, M.Q.; Chung, K.L.; Chyi, C.S. Review of Deep Reinforcement Learning-Based Object Grasping: Techniques, Open Challenges, and Recommendations. *IEEE Access* **2020**, *8*, 178450–178481. [CrossRef]
40. Nian, R.; Liu, J.; Huang, B. A review On reinforcement learning: Introduction and applications in industrial process control. *Comput. Chem. Eng.* **2020**, *139*, 106886. [CrossRef]
41. Patnaik, S. *New Paradigm of Industry 4.0*; Springer: Manhattan, NY, USA, 2020.
42. François-Lavet, V.; Henderson, P.; Islam, R.; Bellemare, M.G.; Pineau, J. An introduction to deep reinforcement learning. *arXiv* **2018**, arXiv:181112560.
43. Hays, T.; Keskinocak, P.; De López, V.M. Strategies and challenges of internet grocery retailing logistics. *Appl. Supply Chain. Manag. E-Commer. Res.* **2005**, *92*, 217–252.
44. Cano, J.A.; Correa-Espinal, A.A.; Gómez-Montoya, R.A.J. Management. An evaluation of picking routing policies to improve warehouse efficiency. *Int. J. Ind. Eng. Manag.* **2017**, *8*, 229.
45. Wadhwa, R.S. Flexibility in manufacturing automation: A living lab case study of Norwegian metalcasting SMEs. *J. Manuf. Syst.* **2012**, *31*, 444–454. [CrossRef]
46. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE international conference on computer vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
47. Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. *Microsoft Coco: Common Objects in Context*; Lecture Notes in Computer Science; Springer: Manhattan, NY, USA, 2014; pp. 740–755.
48. Xie, Z.; Zhong, Z. Visual Conspicuity Measurements of Approaching and Departing Civil Airplanes using Lab Simulations. *Int. J. Simul. Syst. Sci. Technol.* **2016**, *17*, 81–86.
49. Xie, Z.; Zhong, Z. Simulated civil airplane visual conspicuity experiments during approaching and departure in the airport vicinity. In Proceedings of the 2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS), Bangkok, Thailand, 25–27 January 2016; pp. 279–282.
50. Schaller, R. Moore's law: Past, present and future. *IEEE Spectr.* **1997**, *34*, 52–59. [CrossRef]
51. Mahler, J.; Matl, M.; Satish, V.; Danielczuk, M.; DeRose, B.; McKinley, S.; Goldberg, K. Learning ambidextrous robot grasping policies. *Sci. Robot.* **2019**, *4*, eaau4984. [CrossRef]
52. Sucan, I.A.; Moll, M.; Kavraki, L.E. The Open Motion Planning Library. *IEEE Robot. Autom. Mag.* **2012**, *19*, 72–82. [CrossRef]
53. Bohlin, R.; Kavraki, L.E. Path planning using lazy PRM. In Proceedings of the 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065), Francisco, CA, USA, 24–28 April 2000; pp. 521–528.
54. Kavraki, L.; Svestka, P.; Latombe, J.-C.; Overmars, M. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robot. Autom.* **1996**, *12*, 566–580. [CrossRef]
55. Sánchez, G.; Latombe, J.-C. A Single-Query Bi-Directional Probabilistic Roadmap Planner with Lazy Collision Checking. In Proceedings of the International Symposium of Robotic Research, Lorne, Victoria, Australia, 9–12 November 2001; pp. 403–417.
56. Sucan, I.A.; Kavraki, L.E. Kinodynamic motion planning by interior-exterior cell exploration. *Algorithmic Found. Robot. VIII* **2009**, *57*, 449–464.
57. Xie, Z.; Zhong, Z.W. Changi Airport Passenger Volume Forecasting Based On An Artificial Neural Network. *Far East J. Electron. Commun.* **2016**, *2*, 163–170. [CrossRef]
58. Bormann, R.; Wang, X.; Völk, M.; Kleeberger, K.; Lindermayr, J. Real-time Instance Detection with Fast Incremental Learning. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 13056–13063.
59. Liang, R.; Wang, Y.; Xie, Z.; Musaoglu, S. A Vacuum-Powered Soft Robotic Gripper for Circular Objects Picking. In Proceedings of the 2023 6th IEEE-RAS International Conference on Soft Robotics (RoboSoft 2023), Singapore, 3–7 April 2023.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.