robotics

MDPI

Article

Shelf Replenishment Based on Object Arrangement Detection and Collapse Prediction for Bimanual Manipulation

Tomohiro Motoda [1,*], Damien Petit [1], Takao Nishi [1], Kazuyuki Nagata [2], Weiwei Wan [1] and Kensuke Harada [1,2]

[1] Graduate School of Engineering Science, Osaka University, Osaka 560-8531, Japan
[2] Industrial CPS Research Center, National Institute of Advanced Industrial Science and Technology (AIST), Tokyo 135-0064, Japan
* Correspondence: motoda@hlab.sys.es.osaka-u.ac.jp

Abstract: Object manipulation automation in logistic warehouses has recently been actively researched. However, shelf replenishment is a challenge that requires the precise and careful handling of densely piled objects. The irregular arrangement of objects on a shelf makes this task particularly difficult. This paper presents an approach for generating a safe replenishment process from a single depth image, which is provided as an input to two networks to identify arrangement patterns and predict the occurrence of collapsing objects. The proposed inference-based strategy provides an appropriate decision and course of action on whether to create an insertion space while considering the safety of the shelf content. In particular, we exploit the bimanual dexterous manipulation capabilities of the associated robot to resolve the task safely, without re-organizing the entire shelf. Experiments with a real bimanual robot were performed in three typical scenarios: shelved, stacked, and random. The objects were randomly placed in each scenario. The experimental results verify the performance of our proposed method in randomized situations on a shelf with a real bimanual robot.

Keywords: shelf replenishment; bimanual manipulation; deep learning in grasping and manipulation

Citation: Motoda, T.; Petit, D.; Nishi, T.; Ngata, K.; Wan, W.; Harada, K. Shelf Replenishment Based on Object Arrangement Detection and Collapse Prediction for Bimanual Manipulation. *Robotics* **2022**, *11*, 104. https://doi.org/10.3390/robotics11050104

Academic Editor: Oscar Reinoso Garcia

Received: 17 August 2022
Accepted: 19 September 2022
Published: 22 September 2022

1. Introduction

Shelf replenishment in warehouses and retail stores is a particularly challenging example of dexterous robotic tasks. Recently, the use of robots in retail has rapidly increased. However, presently, most practical situations require humans to handle shelf-related tasks, owing to their flexibility and reliability, despite the recent progress in vision processing, manipulations, and the development of functional grippers [1–5].

The replenishment process is divided into two cases. In the first case, a space is found in which the object to be inserted fits, and the object is placed there. In the second case, no space is available, and objects already on the shelf must be moved to create space to place the new object. In the latter case, the manipulation of the objects on the shelf to create an insertion space must be performed carefully to avoid tipping over or damaging the objects already on the shelf. Appropriate manipulation strategies are required in both cases.

Shelf manipulation has recently received increased attention, owing to the use of various robots in the logistics and retail domains. For example, the Amazon Picking Challenge (APC) and Amazon Robotic Challenge (ARC) are competitions that encourage autonomous robotic manipulations in the cluttered environments of warehouses [6,7]. In practice, shelf scenes are densely concentrated and complex; therefore, all objects must be considered, including the target object. Recent learning-based research has contributed to the development of high-precision bin picking [8]. These methods compute the best grasp pose from an RGB-D image. Additionally, the picking systems adopted in [9,10] used a learning-based grasp detection and action decision model to handle the difficulty involved in picking a specific target from a complex scene. Recent studies involving

shelf replenishment tasks include Refs. [11,12]. The first proposed method for planning manipulation tasks is executed using reactive control. The second proposed knowledge-based autonomous object manipulation method uses implicit failure recovery. These approaches have improved dexterity but do not solve the problem of manipulating objects while avoiding neighboring objects from collapsing.

In randomized picking, densely stacked objects on a shelf are obstacles for successful grasping. Domae et al. [13] computed the grasp configurations of robotic grippers that did not collide with obstacles. Harada et al. [14] developed a method to prevent contact with adjacent objects, using machine learning while picking the target object. Dogar et al. [15] pushed obstacles to reach a target in cluttered environments. Lee et al. [16] and Nam et al. [17] relocated obstacles to retrieve a target object from clutter. Nagata et al. [18] defined the grasp patterns to be linked to the surfaces of target objects and proposed a dexterous strategy for sliding a top object, or tilting an aligned object from a complex environment to extract the target object. However, it is difficult to use these approaches to effectively avoid a collapse that would damage the objects, despite accurately picking the overlapping objects of the target.

Object detection is an integral component of shelf manipulation. Learning-based object detection has been widely studied in robotics [19], and its accuracy tends to be related to successful robotic manipulations. Goldman et al. [20] provided a network architecture for identifying each object in a dense display. Asaoka et al. [21] proposed a method that groups organized objects in an image and identifies the arrangement pattern of each group. However, these methods assume that the objects are properly stored on a shelf. Considering that objects on the shelf are cluttered, this study categorizes them into disorganized and organized objects.

Scene-understanding approaches have been proposed in robotics to understand qualitative structures and spatial relationships [22]. Panda et al. [23] estimated the support relationship using only simple interactions of stacked objects in tabletop scenarios. Mojtahedzadeh et al. [24] estimated the physical interactions between objects using 3D visual perception and machine learning. In picking approaches that consider support relations, Grotz et al. [25] extracted physically plausible support relations between objects from point clouds to predict the action effects. Nevertheless, these methods are needed to estimate the three-dimensional shape of objects in uncertain environments. Another similar approach is to use an RGB-based deep neural network to predict the stacking order of objects, as in [26]; however, it detected the stacking order of objects on a desk from the top view only. Therefore, we focus on the shelf containing the pile of the cluttered object in accordance with the situation, such as warehouses and retail stores.

Humans understand the reactions when an action is forced on an object and therefore manipulate objects based on these predictions. Similarly, previous approaches in robotics have evaluated each desired action with scene understanding to ensure safe and reliable results [27,28]. In learning-based predictions, CNN-based networks infer the next state in the targeted scene [29]. Magassouba et al. [30] predicted the risk of collision from an RGB-D image before the placement of an object. Janner et al. [31] presented a framework for learning object-oriented representations for physical scene understanding from image observations to predict the object state transition per time lapse. In our previous work [32], we proposed the learning-based evaluator to predict the risk of collapse of a shelf, based on both the desired object extraction and object evaluation supporting the successful extraction. The neural network explicitly learns the relationship between objects (extract/support) and evaluates whether a collapse would occur. The extracting action with the minimum risk of collapse was selected; however, manipulations necessary for shelf replenishment were not suggested. The present study automates replenishment by improving our previous collapse prediction network and proposes a new action plan while minimizing changes to the state of objects on the shelf. Moreover, our method supports the use of bimanual arms to create an insertion space while considering the safety of the shelf content.

In this study, a novel approach for automating the replenishment of disorganized shelves with a bimanual robot is presented (Figure 1). First, we classified the objects in organized/disorganized displays using a general object detection method. This allowed us to treat these categorized objects explicitly. Our deep neural network infers the neighboring object's behavior from a depth image when removing a specified manipulation target; that is, the network can predict which objects fall from a shelf. The deep neural network was trained on a dataset generated using a simulator. The proposed inference-based strategy provides an appropriate decision and course of action on whether to create an insertion space while considering the safety of the shelf content. Compared to our previous work, we improved our collapse prediction estimator to be applied to a shelf replenishment task, allowing the robot to estimate the risk of single-arm manipulation without supporting the other objects. We considered the replenishment task through single-arm/bimanual manipulation to cover various practical cases.

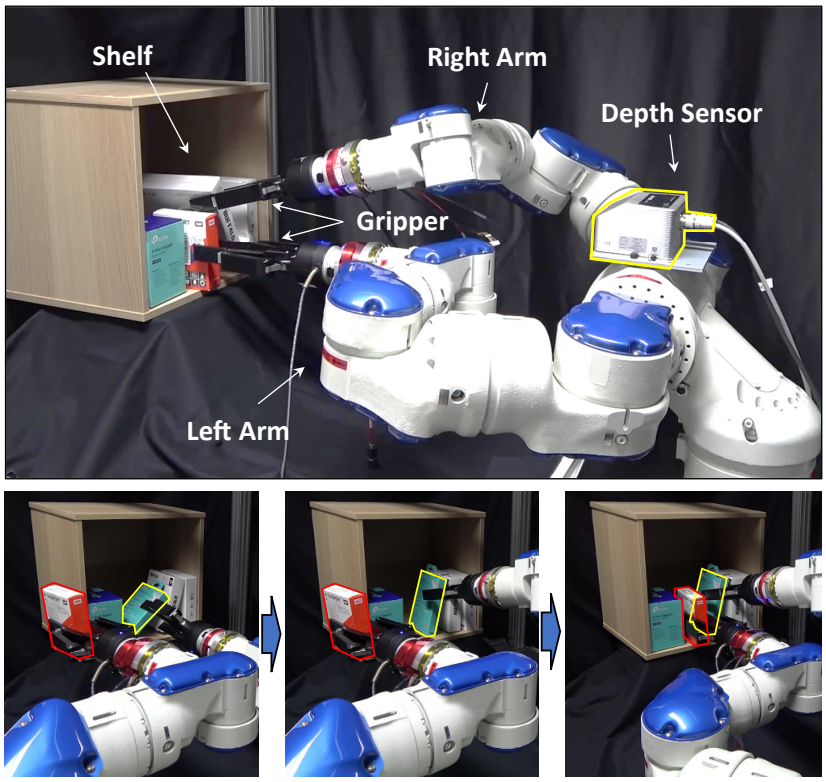

Figure 1. Bimanual robotic shelf replenishment. We present a robotic shelf replenisher with bimanual manipulation, which fills the shelf with an object. Our method allows for the slight rearrangement of the shelf to create space for replenishment without damaging the shelf or the other objects. Given the state of the shelf, bimanual or single-arm operation is appropriately selected to plan the action.

The main contributions of this study can be summarized as follows.

- We classify objects in organized/disorganized displays to understand the shelf display as a whole, which reduces the complexity of inter-object relationship analysis and allows the manipulation of a group of objects as a unit instead of single objects.
- Our method enables novel action planning with a bimanual robot for shelf replenishment by predicting the occurrence of an object collapsing via a neural network. In particular, our method can consider any state of the shelf, and select the best action for each state, including single-arm or bimanual manipulation.

The remainder of this paper is organized as follows. Section 2 explains the proposed shelf-replenishment algorithm. Section 3 describes the experiments and network benchmark used to evaluate our architecture. Section 4 provides a discussion. Finally, Section 5 concludes the paper.

2. Materials and Methods

Figure 2 illustrates the flow of our architecture. We present an approach for automating replenishment using vision-based detection and bimanual manipulation. First, the scene is analyzed to classify objects into object arrangement patterns, i.e., stacked, shelved, and disorganized. Second, a collapse prediction network is used to predict the safety of different actions. Third, the proposed strategy selects a bimanual action plan from a list of potential safe actions to organize the shelf, if necessary, and place the object on the shelf.

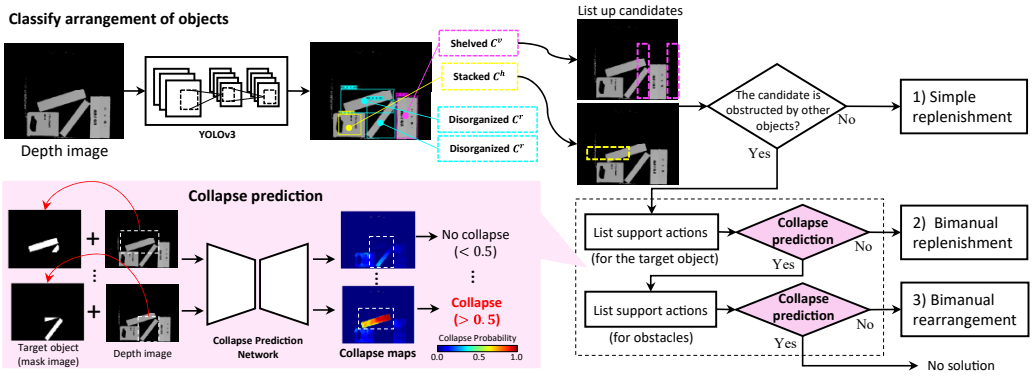

Figure 2. Overview of our bimanual robotic replenishment pipeline, which consists of (1) shelf scene classification into organized/disorganized arrangement using YOLOv3 [33], and (2) action planning based on a collapse prediction network that predicts the probabilities of collapse from a shelf in the form of a heatmap. The depth image is captured from the 3D vision sensor and then fed to YOLOv3 to classify the shelf scene into organized/disorganized arrangements (top left). The flowchart on the right shows the action planning to replenish an object based on the classification results. Each action is evaluated with the collapse prediction network (bottom-left) to avoid the objects from collapsing during bimanual manipulation.

2.1. Objects Arrangement Classification

The first step of our framework regards classifying the object arrangement. We used YOLOv3 (You Only Look Once, version 3) [33], a real-time object detection algorithm that identifies specific objects in a picture, to classify clusters of objects into object arrangement patterns.

As shown in Figure 3, the arrangements of objects are defined as one of the following four classes: stacked C^h, shelved C^v, disorganized right C^r, and disorganized left C^l. C^h and C^v denote horizontally and vertically arranged patterns, respectively, and C^r and C^l are disorganized patterns that lean to the right and left, respectively. In the case of a single object, it will be classified as C^v. YOLOv3 also generates a bounding box of the cluster. We define bounding box B_i $(i = 1, \ldots, N)$ as follows (N is the number of the generated B_i):

$$B_i = (x_i, y_i, w_i, h_i) \tag{1}$$

where (x_i, y_i) denotes a center position, and w_i, h_i denotes width and height, respectively. To apply YOLOv3 for our object arrangement pattern classification, we used the weighted model pretrained on ImageNet [34] and pretrained the model with real depth images. A depth sensor acquired the depth image with 256-step grayscale, which showed 5–10 rectangular objects on a shelf. Here, we do not use RGB images but depth images with the assumption that the object's textures are unnecessary for classifying the object arrangement patterns. To distinguish the disorganized pattern as either C^r or C^l, our training process does not use data augmentation by randomly flipping the images. We used 500 images to train the model, and annotations were performed manually. The confidence score was empirically set to 0.30, and the threshold of the intersection over union (IoU), which is the accuracy of the individual identification of the bounding box, was set to 0.45. The training

at 100 epochs took 1 h on a system running Ubuntu 16.04 with an Intel Core i7-9700F CPU clocked at 3.00 GHz and a single NVIDIA GeForce RTX 2060 SUPER graphics card with CUDA 10. The results are presented in Figure 3.

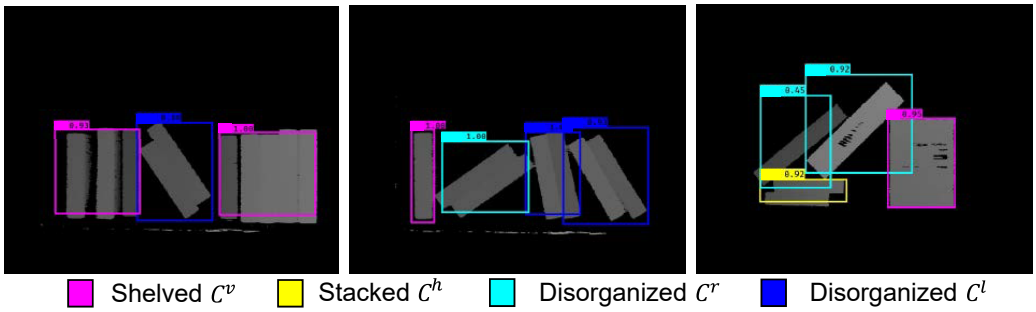

Figure 3. Detection results of object arrangement category using YOLOv3 classification. As shown in these images, each object arrangement is categorized into four classes: stacked C^h, shelved C^v, disorganized right C^r, and disorganized left C^l.

2.2. Collapse Map

We propose a collapse prediction network that can manipulate an object without the collapse of neighboring objects. The network outputs a heatmap that shows the pixel-wise collapse probabilities, that is, the collapse map. In this section, we first describe the architecture of our network model and then introduce the data collection and training settings applied to generate our model.

2.2.1. Network Architecture

In our previous study [32], we proposed an approach for shelf picking to assess whether extracting an object is possible based on a collapse prediction network. However, the use of the network is limited to a specific bimanual action to extract a target object while supporting an adjacent object; thus, we can only hold the adjacent object so as not to move based on the result of the collapse prediction. In the present study, we improve the collapse prediction network to directly determine the potential of an object falling from a shelf when removing the specified object with a single arm. This enables us to plan a sequential approach for replenishment based on the collapse probability.

The network is comprised of an encoder and decoder. The input data were a depth image of the shelf scene and a target mask image (binary image) of the specified object, in which the region representing the target object was set to 1 and the other regions were set to 0. The encoder network has two pipelines, as shown in Figure 4. One network extracts features from a depth image based on the convolutional layer of VGG-16 [35]. The other has five convolutional layers to compress the binary mask image. The outputs of the two pipelines are concatenated and fed into the decoder network. Finally, the computed collapse map is upsampled to match the size of the input depth image. The first branch has a skip architecture to improve the semantic segmentation performance. The input image was 256×256 grayscale and normalized in advance. Similarly, the mask image size was 256×256.

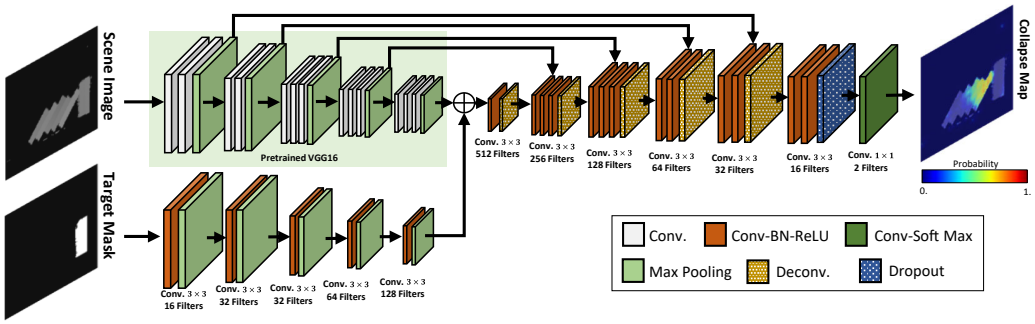

Figure 4. Network architecture. The collapse map network receives both a depth scene and a binary mask of a target object as the input. The output of the collapse map network is a heatmap, which shows the probability of position changes, e.g., an object turning over or falling down.

2.2.2. Dataset

For dataset generation, we used a maximum of 10 rectangular objects of various sizes in PhysX [36]. As shown in Figure 5, five to nine objects were first randomly sampled, which were initially positioned in an organized arrangement pattern as either C^h and C^v. Half of these objects were then assigned random poses to generate a disorganized arrangement. Subsequently, a target object was randomly selected and removed from the shelf. We then checked the positions of all objects, except for the target object, after the target object was removed from the shelf and the other objects reached a stable state. The objects that move during this operation constitute a collapse mask (binary image), in which the regions representing those objects were set to 1 and the other regions were set to 0 as shown in Figure 5. If the change of the objects' center position exceeds the threshold, we judge the objects to be moved. Note that we empirically set the threshold to 6.4 mm. To train our network on a pixel basis, we collected a depth image, target mask, and collapse mask, where the images were rendered from the recorded results. The depth image shows the initial arrangement before the selected target object is removed. The target mask shows the selected target object, and the collapse mask shows the objects that moved after the selected target object was removed. Finally, we empirically set the simulation parameters according to their actual movements as follows: we set the coefficient of static to 0.9, dynamic friction to 0.8, the coefficient of restitution to 0.1, and the density to 1.0 kg/m^3.

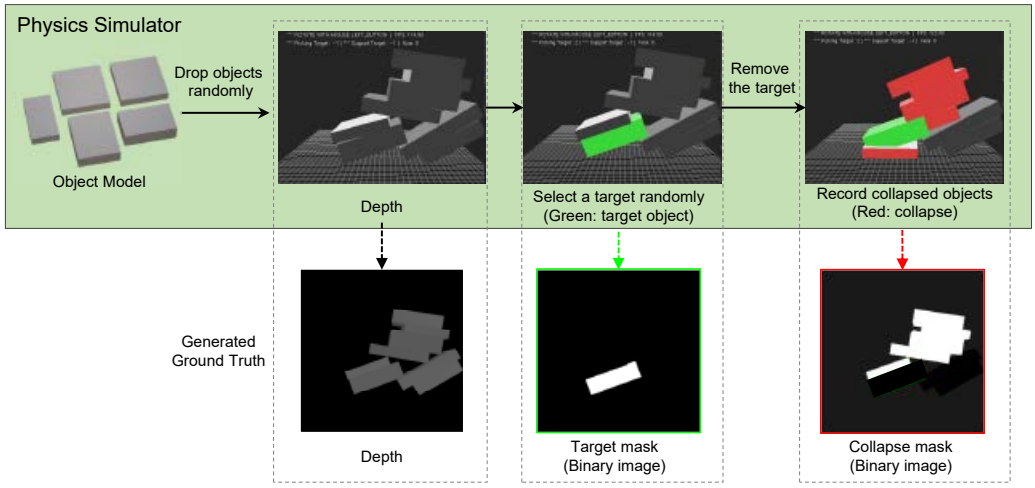

Figure 5. Dataset generation procedure. In our method, nine novel objects are used from five types of objects. First, the initial objects are stacked randomly in vertical or horizontal status. Next, the target object is removed, and the simulator monitors the movement of the other objects. Finally, the moving objects are marked as the collapse region, and one dataset is generated through a single simulation (bottom of the figure: depth scene image, target mask, and collapse mask).

2.2.3. Implementation Details

We built a dataset of 22,400 training images generated from the simulator described in Section 2.2.2 and trained the collapse map network. To eliminate the discrepancies between the real and synthetic depth images, noise was randomly added to the generated depth images.

We used a batch size of 32 (700 iterations) and the Adam optimizer [37] with a learning rate of 1.0×10^{-4}. The other Adam hyperparameters were set as the default values of $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The training at 50 epochs took 8 h on a system running Ubuntu 16.04 with an Intel Core i7-9700F CPU clocked at 3.00 GHz, and a single NVIDIA GeForce RTX 2060 SUPER graphics card with CUDA 10. The network achieved a processing time of 0.02 s or less to generate one collapse map.

2.3. Shelf Replenishment

Shelf replenishment requires action planning to place an object along the arrangement. However, moving an object in a densely stacked scene, i.e., a shelf, involves the risk of dropping the other object because safe manipulation on a shelf is complicated, especially considering the dynamics. To solve the problem, we formulate the robotic action as the manipulation within the limit of the bounding box based on the arrangement classification. Here, the collapse map can detect the risk of handling the object inside the bounding box. If there is no risk, we can provide the replenishment strategy without considering the strict dynamics.

As shown in Figure 3, the shelf scene is represented by bounding boxes B_i and classes. To find sufficient space to replenish an object, we define three placement candidates as rectangles on the top, left, and right of each target bounding box B_i in case the class is a stacked C^h or shelved C^v. Here, let B_i^{top}, B_i^{left}, and B_i^{right} denote the bounding box of the placement candidates to be placed on the shelf. B_i^{top} denotes the area where an object can be stacked on C^h, which we describe as

$$B_i^{top} = (x_i - \frac{w_i}{2} + \frac{w'}{2}, y^i - \frac{h_i}{2} - \frac{h'}{2} - g, w' + 2g, h') \tag{2}$$

B_i^{right} and $B_i{}^{left}$ denote the areas where an object can be placed on the right and left sides of C^v, respectively, which we describe as

$$B_i^{right} = (x_i - \frac{w_i}{2} - \frac{w'}{2} - g, y^i + \frac{h_i}{2} - \frac{h'}{2}, w' + 2g, h') \tag{3}$$

$$B_i^{left} = (x_i + \frac{w_i}{2} + \frac{w'}{2} + g, y^i + \frac{h_i}{2} - \frac{h'}{2}, w' + 2g, h') \tag{4}$$

where w' and h' are the height and width of the area to secure space, respectively, and g is the thickness of the fingers of the gripper. Each arrangement has the potential to place an object on B_i^{top}, B_i^{left}, and B_i^{right} unless the bounding box is outside the shelf. Note that the object is known, which fits in the secure area (the size of $w' \times h'$), and each candidate is excluded when it exceeds the limit of the working space. In the present study, the size of the inside of the working space (the shelf) is W330 × D280 × H330 mm.

As shown in Figure 2, based on the predicted collapse map for a target object and the prediction for each action, we assume three manipulations for replenishment.

2.3.1. Simple Replenishment

Firstly, we check that there is sufficient space in the candidate area (B_i^{top}, B_i^{left}, or B_i^{right}) to place an object on the shelf. Note that the size of the object placed on the shelf is known. When this condition is satisfied (i.e., there is no object inside the candidate area), the robot places the object at the center position of the candidate area.

2.3.2. Bimanual Replenishment

Secondly, when objects occupy all candidates, the objects must first be removed from the candidates. In the present study, we define this action as the supporting action in this study. Let B_s denote the arrangement overlapping the candidate (B_i^{top}, B_i^{left}, or B_i^{right}). On the collapse map targeted at the bounding box B_s, an area with a probability equal to or higher than the predefined probability threshold is defined as the collapse region R_s. If $\mathrm{IoU}(B_j, R_s) < th \quad (j \neq s, i)$, the objects in B_j should be stable after moving B_s; that is, B_s is movable. IoU$(\cdot)$ denotes a function that outputs the IoU, and th is a threshold value. The supporting action is defined as moving B_s horizontally to create space in a cluttered scene. The starting point p^{start} and goal point p^{goal} are defined as follows:

$$p^{start} = (x_s, y_s) \tag{5}$$

where (x_s, y_s) is the center position, and x_s and y_s denote the coordinates. Then, we then define the goal point subject to the target position as follows:

$$p^{goal} = \begin{cases} (x_s + \frac{w'}{2} + \frac{w_s}{2} + g, y_s) & \text{if } B_s \text{ on } B_i^{right} \\ (x_s - \frac{w'}{2} - \frac{w_s}{2} - g, y_s) & \text{if } B_s \text{ on } B_i^{left} \\ (x_s, y_s - \frac{h'}{2} - \frac{h_s}{2} - g) & \text{if } B_s \text{ on } B_i^{top} \end{cases} \tag{6}$$

where g denotes the margin of the gripper fingers in the experiment. Figure 6 shows the bimanual replenishment process. When the class of B_s is either C^r or C^l, the objects in B_s are rotated, aligned with the organized arrangement, and moved to the goal point with one robotic arm. While holding them for safety, the object is then placed in the placement candidate with the other robotic arm.

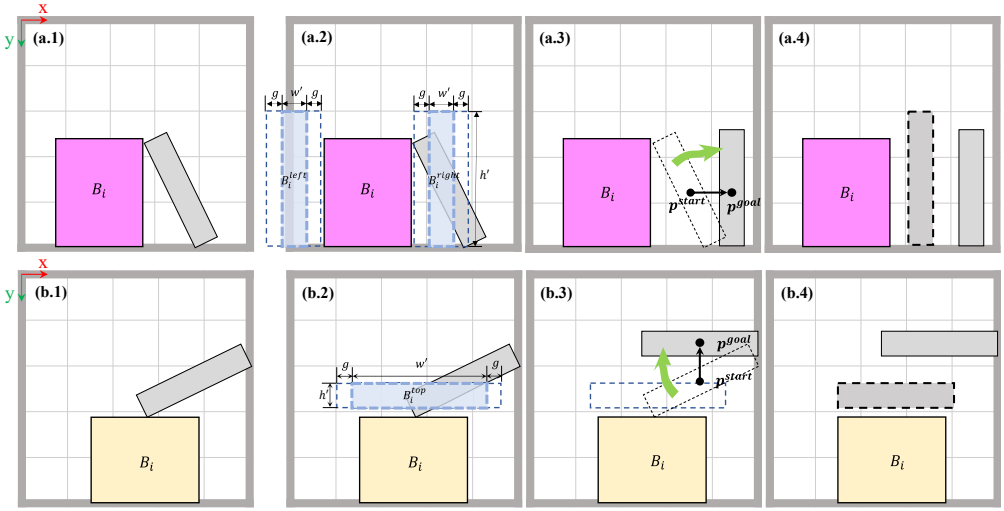

Figure 6. Example executions of bimanual replenishment: (**a.1**–**a.4**) Replenishment for a shelved scene. The robot moves the disorganized object to the right or left to place an object aligned with the shelved objects. (**b.1**–**b.4**) Replenishment for a stacked scene. The robot lifts the disorganized object to place an object on the stacked objects.

2.3.3. Bimanual Rearrangement

Finally, in case there is no candidate that satisfies the requirements, we repeatedly consider the rearrangement. If $\mathrm{IoU}(B_i, R_s) \geq th \quad (i \neq s)$ for all candidates, then supporting the objects in B_i increases the collapse risk. We select the arrangement $B_i\{i = 1, \ldots, N\})$ that has the highest overlapping rate to the collapse region R_s.

$$k = \underset{i \in 1,\ldots,N}{\arg\min} |\mathrm{IoU}(B_i, R_s)| \tag{7}$$

Here, we generate the collapse map targeted at the bounding box B_k and calculate the collapse region R_k. As mentioned above, If $\mathrm{IoU}(B_j, R_k) < th \quad (j \neq s, i, k)$, B_k is movable. The supporting action is defined as moving B_k horizontally to avoid object collapse. The starting point $p^{start,\dagger}$ and goal point $p^{goal,\dagger}$ are defined as follows:

$$p^{start,\dagger} = (x_k, y_k) \tag{8}$$

$$p^{goal,\dagger} = \begin{cases} (x_s - \frac{w_s}{2} - \frac{w_k}{2} - g, y_k) & \text{if } x_k < x_s \\ (x_s + \frac{w_s}{2} + \frac{w_k}{2} + g, y_k) & \text{if } x_k \geq x_s \end{cases} \tag{9}$$

When the obstacle for the supporting action is moved, it is possible to safely move B_s (Figure 7). If B_k is not movable, supporting it is also required. However, the bimanual robot cannot place the object while holding two or more objects. In the present study, we excluded such cases from consideration.

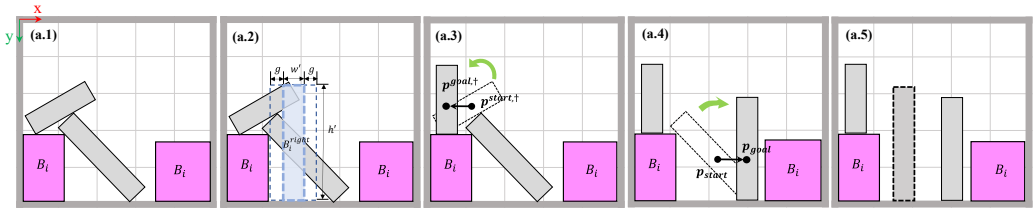

Figure 7. Bimanual arrangement example: (**a.1**–**a.5**) When more than two objects obstruct the replenishment, we need to move them with multi-step actions. The robot moves the objects one by one to make space to place an object.

3. Experiments and Results

In this section, we report the implementation details, experimental results, and the benchmark of the collapse network for evaluating the performance of our proposed method.

3.1. Predicting the Collapse Map

Figure 8 shows the collapse maps results with the validation data. We report the pixel accuracy to quantify the classifications and calculate these metrics as follows:

$$\text{Pixel Accuracy} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}, \tag{10}$$

$$\mathrm{IoU} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}. \tag{11}$$

where TP, TN, FP, and FN denote the numbers of true positives, true negatives, false positives, and false negatives, respectively, counted on a pixel basis. In our evaluations, we assume that a pixel is classified as a collapse region when the probability is higher than 0.5.

We validated our prediction model using 1000 simulated images created from the simulator. Compared to our previous baseline model (based on FCN-8s [38]), we achieved a pixel accuracy and IoU score of 0.982 and 0.668, respectively, as shown in Table 1. A comparison between the other parameters and these results shows that the batch size parameter was chosen appropriately (Table 1). Based on this result, we set the batch size to 32 and used the transfer learning of VGG-16, which was pretrained with ImageNet. We further note that our model infers that the object moves under physical dynamics; however, it achieves similar or better IoU scores than those of related studies [39,40].

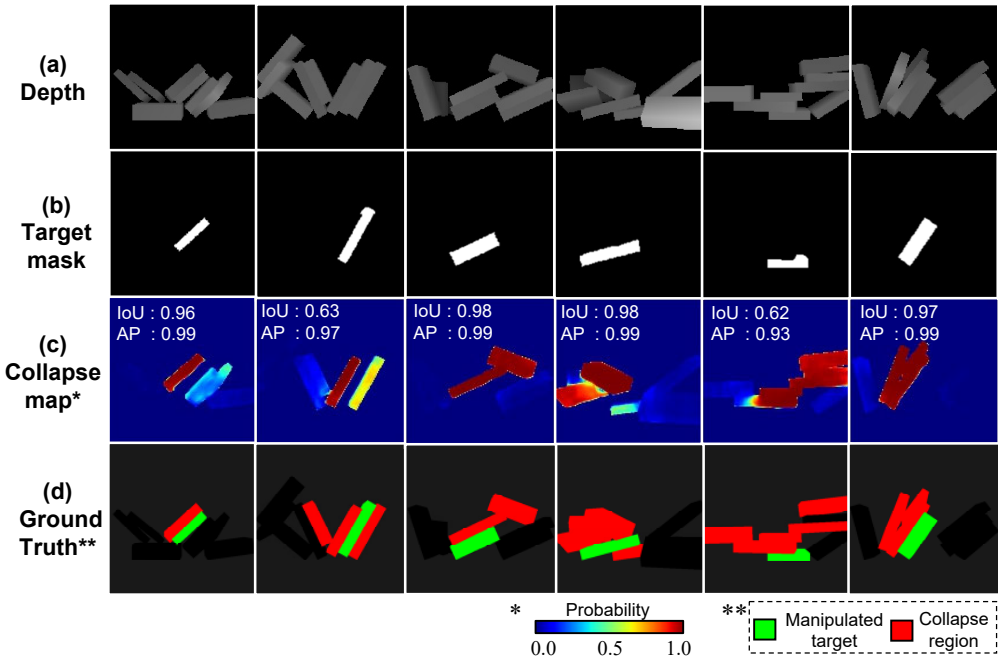

Figure 8. Collapse maps results: (**a**) Depth image showing the depth from a viewpoint in grayscale. (**b**) Target mask that shows a targeted object (white). (**c**) Collapse map generated from the collapse prediction network. (**d**) Ground truth, which consists of collapse regions (red) and target object (green).

Table 1. Performance comparison between collapse predictions for each setting.

Method	PA *	IoU **
FCN-8s-based	0.941	0.461
Ours (Batch size = 32)	**0.982**	**0.668**
Ours (Batch size = 16)	0.981	0.662
Ours (fine-tuned, Batch size = 16)	0.980	0.640
Ours (fine-tuned, Batch size = 32)	0.957	0.545

* Pixel Accuracy. ** Intersection Over Union.

3.2. Robotic Experiments

The efficiency of the proposed method was validated using real robotic experiments. We used MOTOMAN-SDA5F from Yaskawa Electric Corp. for our experiments [41]. The SDA5F has 15 degrees of freedom (DoFs): 7 DoFs per arm and one DoF for the waist. The robot was programmed using Choreonoid [42] and graspPlugin [43]. Two Robotiq gripper 2F-140 adaptive grippers [44] were used, which were installed at the arms of the SDA5F. The 2F-140 adaptive gripper is an underactuated parallel gripper. We used a YCAM3D-10L from YOODS Co. Ltd., Yamaguchi, Japan [45], which is a depth camera based on the phase shift method. We obtained a depth image from YCAM3D-10L. We used a median filter to smooth the image for noise removal from the real data. Each original image was resized to 256 × 256 pixels. We used 4–6 different rectangular objects. The objects were presented to the robot on the shelf, and a similar scene was maintained in the simulator.

We report the results of the experiments with a real robot in three typical scenarios: shelved, stacked, and random. The objects were randomly placed in each scenario. Success was defined as the case in which the replenishment of an object was completed. In a sequence of 100 experiments, 68 trials succeeded in obtaining the entire result (68.0%). From the viewpoint of each arrangement, the success rates were 57.5%, 84.0%, and 30.0% in the stacked, shelved, and random scenes, respectively. Moreover, we evaluated the performance of our collapse prediction. Our method without the collapse prediction showed

comparatively lower success rates. In particular, it performed poorly on rearrangements, which required moving objects inside the shelf, compared to the case when using the collapse prediction. In a sequence of 25 experiments, only 11 trials achieved the entire result (44.0%), and the success rates were 50.0%, 60.0%, and 0.0% in the stacked, shelved, and random scenes, respectively. Table 2 presents the corresponding statistics.

Table 2. Robotic experiment results.

	Stacked	**Shelved**	**Random**	**Total**
Success w/ Collapse Prediction	23/40 (57.5%)	42/50 (84.0%)	3/10 (30.0%)	68/100 (68.0%)
Success w/o Collapse Prediction	5/10 (50.0%)	6/10 (60.0%)	0/5 (0.0%)	11/25 (44.0%)

Figure 9 shows snapshots of the experiment, where the object was initially placed vertically on the shelf. Figure 9(a.1,b.1) show two scenes within the experiment. The depth images shown in Figure 9(a.2,b.2) were classified by our fine-tuned YOLOv3. The steps of these experiments are depicted in Figure 9(a.3–a.8,b.3–b.8), where the candidate placement can be derived by placing the objects on the left or right according to the display identified as shelved. An object leaning to the left (Figure 9a) or to the right (Figure 9b) is located at the planned placement point. The object was grasped by the right-hand gripper, rotated to align it, and moved to the right. We assumed the diagonal direction of the bounding box to be the angle of inclination of the object under the prior positional information (C^l / C^r). Additionally, snapshots of the experiment in which the object was placed horizontally on the shelf are shown in Figure 9c,d. Figure 9c shows how the obstructing object was grasped with one hand, lifted, and placed on top of the other hand. In Figure 9d, the object was placed on top of the objects on the shelf without the need to use the right hand, as no other object was detected.

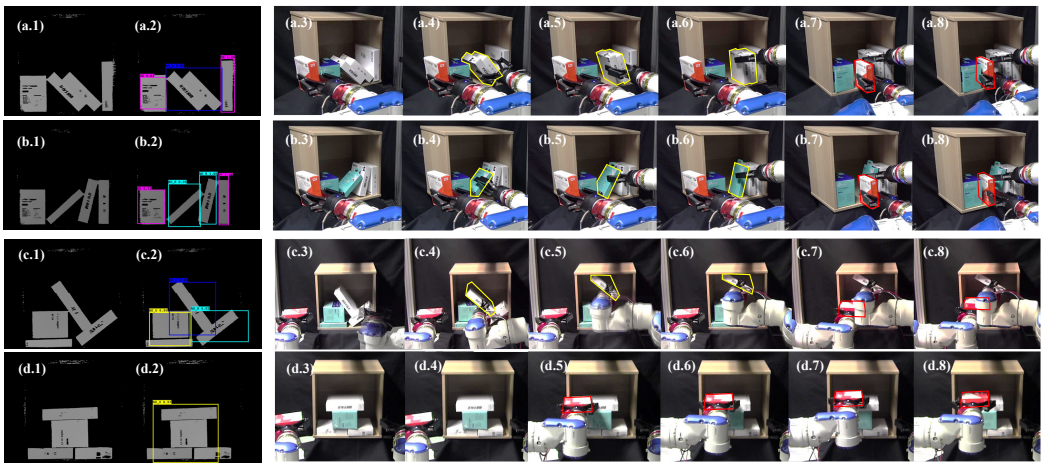

Figure 9. Snapshots of the experiments: To align an object with the vertical arrangement in (**a.1**,**a.2**,**b.1**,**b.2**), the robot horizontally moves the other objects that occupy the space of the shelf, using the other arm, see (**a.3**–**a.8**,**b.3**–**b.8**) . To align an object with the horizontal arrangement in (**c.1**,**c.2**,**d.1**,**d.2**), the robot lifts the other objects that occupy the space on the shelf using the other arm, see (**c.3**–**c.8**,**d.3**–**d.8**).

If the obstacle cannot be moved off the shelf with one hand, we can select the multi-step motions to organize the objects with dual arms, as shown in Figure 10(a.1–a.8, b.1–b.8). Using the collapse maps for each object, we selected the supported and moved objects that could be securely manipulated and well organized.

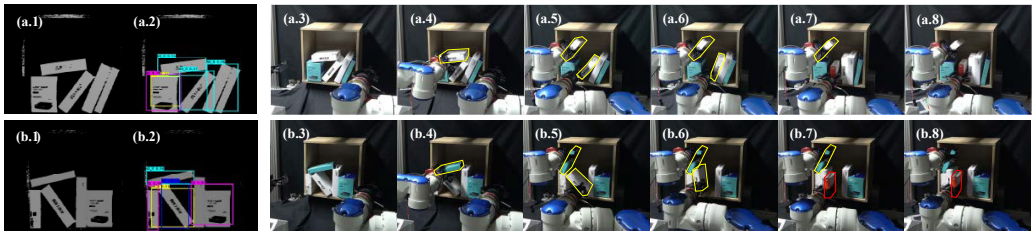

Figure 10. Snapshots of the specific scenarios in (**a.1**,**a.2**,**b.1**,**b.2**), which require multi-step actions to replace an object and to make space for replenishment in (**a.3**–**a.8**,**b.3**–**b.8**).

4. Discussion

This study analyzed an unknown shelf display to predict the risk of collapse during a replenishment operation. This enables the robot to replenish a shelf with an object by selecting a strategy based on the situation. Experiments on two typical and complex arrangements confirmed that the bimanual action plan replenished the object while dealing with disorganized arrangements.

Our proposed method has practical relevance when considering the difficulty of maintaining organized arrangements in retail and warehouses. When objects were not organized, Lee et al. [16] and Nam et al. [17] conducted rearrangement actions similar to ours and approached the target object, assuming that all objects were on the same plane. By contrast, our proposed method can move objects without collapse, even if the objects overlap with each other. In terms of safety, Zhang et al. [4] and Panda et al. [23] acquired knowledge about the geometrical structure of a scene to individually detect the support relation. However, they refer only to the safety of operations on an object with no support, that is, an object placed on the top. In contrast, our method quantitatively assesses all objects on the shelf.

It should be noted that the success rate of our method for random scenes is low (approximately 30%), as shown in Table 2. However, we conducted the experiments under strict conditions without the object collapsing, as opposed to [7,11,18,21]. The previous studies, in fact, required the system on a case-by-case basis for the object collapse. Compared with motion planning without collapse predictions, our proposed method can perform successfully on a complex scene using a bimanual robot. Despite its many advantages, there are some limitations associated with the present study. First, we only evaluated the risk of collapse in an instantaneous and static scene to determine the sequential action for replenishment. Therefore, because it cannot handle the collisions and the dynamics that may occur during object movement, the present study assumes that the target object for manipulation is limited to within the bounding box in order to avoid contact between the objects. In other words, this study had minimum space requirements, which makes it difficult to achieve the necessary conditions in narrow and dense shelf environments. In future, handling items requiring high dexterity will need the integration of reactive grasping control and motion planning to perform such tasks, even with grippers with limited dexterity, as shown in [11]. Second, our planner assumes that the robot has two arms and that when one arm moves an object, the other arm supports an obstacle. However, if there are too many disorganized objects, the support action with only one arm is insufficient, and collapse cannot be avoided. Our framework was limited to using only one support action. Accordingly, the mutual support relations among the objects should be analyzed, and an action planner developed based on a search algorithm to deal with many objects. Third, it is difficult to avoid interference between arms in a confined environment. Both arms tend to be close to each other, which makes the computation of inverse kinematics difficult. Particularly, in this method, we do not consider the dynamics and physical contact when considering the stability of learning the network to predict the collapse. Therefore, to increase the success rate, we should use a simulation to consider the robot arm and train the collapse prediction network by considering external interference and self-interference.

We assume that replenishment is to place an object on a shelf so that it is aligned with the typical arrangement in the warehouse or retail store. However, the display becomes disorganized as the objects move in or out of the shelf. The collapse prediction network makes it possible to evaluate the risk of any manipulation to replenish an object without collapse. Additionally, we solved the difficulty of organizing the shelf using a simple algorithm based on collapse predictions. However, because we must handle various objects in different environments, further verification of our proposed method is necessary. We should also examine whether it can be applied to other research fields. Thus, we intend to develop a sequential prediction network that considers the dynamical transition of objects in order to apply our approach to other tasks with different objects, for example, a policy to consider objects in an unstable pose or entangled objects. Similarly, considering the other damage source of items is also necessary for safe shelf manipulation, such as breaking an item with the robot hand's clamping force. Therefore, another interesting future study would be to use the property of the gripper to learn the collapse and the graspability of objects for further adaptability in realistic scenes [46–48].

5. Conclusions

We presented a shelf replenishment system that selects the safest action based on a collapse prediction estimator. Our collapse prediction network generates a probabilistic map from scene images and actions, making safe manipulation possible. In addition, our proposed method plans the best action based on single-arm or bimanual manipulation, making it possible to deal with complicated arrangements. In experiments using a real robot, we demonstrated the efficiency of our method for shelf replenishment.

In future work, we plan to extend our implementation of both the network and the data collection processes: (1) to further deal with any object shape and more disorganized arrangements, such as in retail stores and kitchens; (2) to use the robot properties in the simulations to estimate the physical contacts; and (3) to develop a prediction network to help analyze the state of stacked objects.

Author Contributions: Conceptualization, T.M.; Data curation, T.M.; Funding acquisition, K.H.; Investigation, T.M.; Methodology, T.M., T.N. and K.H.; Project administration, K.H.; Resources, K.H.; Software, T.M.; Supervision, D.P. and K.H.; Writing—original draft, T.M.; Writing—review and editing, T.M., D.P., T.N., K.N., W.W. and K.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Fujita, M.; Domae, Y.; Noda, A.; Garcia Ricardez, G.A.; Nagatani, T.; Zeng, A.; Song, S.; Rodriguez, A.; Causo, A.; Chen, I.M.; et al. What are the important technologies for bin picking? Technology analysis of robots in competitions based on a set of performance metrics. *Adv. Robot.* **2019**, 34, 560–574. [CrossRef]
2. Billard, A.; Kragic, D. Trends and challenges in robot manipulation. *Science* **2019**, 364, eaat8414. [CrossRef] [PubMed]
3. Mahler, J.; Liang, J.; Niyaz, S.; Laskey, M.; Doan, R.; Liu, X.; Ojea, J.A.; Goldberg, K. Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics. *arXiv* **2017**, arXiv:1703.09312.
4. Zhang, H.; Lan, X.; Zhou, X.; Tian, Z.; Zhang, Y.; Zheng, N. Visual Manipulation Relationship Network for Autonomous Robotics. In Proceedings of the 2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids), Beijing, China, 6–9 November 2018; pp. 118–125.
5. Li, J.K.; Hsu, D.; Lee, W.S. Act to see and see to act: POMDP planning for objects search in clutter. In Proceedings of 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea, 9–14 October 2016; pp. 5701–5707.
6. Eppner, C.; Höberfer, S.; Jonschkowski, R.; Martín-Martxixn, R.; Sieverling, A.; Wall, V.; Brock, O. Four aspects of building robotic systems: Lessons from the Amazon Picking Challenge 2015. *Auton. Robot.* **2018**, *42*, 1459–1475. [CrossRef]

7. Zhu, H.; Kok, Y.Y.; Causo, A.; Chee, K.J.; Zou, Y.; Al-Jufry, S.O.K.; Liang, C.; Chen, I.; Cheah, C.C.; Low, K.H. Strategy-based robotic item picking from shelves. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea, 9–14 October 2016; pp. 2263–2270.
8. Lenz, I.; Lee, H.; Saxena, A. Deep learning for detecting robotic grasps. *Int. J. Robot. Res.* **2015**, *34*, 705–724. [CrossRef]
9. Schwarz, M.; Lenz, C.; García, G.M.; Koo, S.; Periyasamy, A.S.; Schreiber, M.; Behnke, S. Fast Object Learning and Dual-arm Coordination for Cluttered Stowing, Picking, and Packing. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 3347–3354.
10. Zeng, A.; Song, S.; Yu, K.T.; Donlon, E.; Hogan, F.R.; Bauza, M.; Ma, D.; Taylor, O.; Liu, M.; Romo, E.; et al. Robotic Pick-and-Place of Novel Objects in Clutter with Multi-Affordance Grasping and Cross-Domain Image Matching. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 3750–3757.
11. Costanzo, M.; Stelter, S.; Natale C.; Pirozzi, S.; Bartels, G.; Maldonado, A.; Beetz, M. Manipulation Planning and Control for Shelf Replenishment. *IEEE Robot. Autom. Lett.* **2021**, *5*, 1595–1601. [CrossRef]
12. Winkler, J.; Balint-Benczedi, F.; Wiedemeyer, T.; Beetz, M.; Vaskevicius, N.; Mueller, C.A.; Fromm, T.; Birk, A. Knowledge-Enabled Robotic Agents for Shelf Replenishment in Cluttered Retail Environments. In Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems (AAMAS), Singapore, 9–13 May 2016; pp. 1421–1422.
13. Domae, Y.; Okuda, H.; Taguchi, Y.; Sumi, K.; Hirai, T. Fast graspability evaluation on single depth maps for bin picking with general grippers. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 1997–2004.
14. Harada, K.; Wan, W.; Tsuji, T.; Kikuchi, K.; Nagata, K.; Onda, H. Initial experiments on learning-based randomized bin-picking allowing finger contact with neighboring objects. In Proceedings of the 2016 IEEE International Conference on Automation Science and Engineering (CASE), Fort Worth, TX, USA, 21–25 August 2016; pp. 1196–1202.
15. Dogar, M.R.; Hsiao, K.; Ciocarlie, M.T.; Srinivasa, S.S. *Physics-Based Grasp Planning through Clutter*; MIT Press: Cambridge, MA, USA, 2012.
16. Lee, J.; Cho, Y.; Nam, C.; Park, J.; Kim, C. Efficient Obstacle Rearrangement for Object Manipulation Tasks in Cluttered Environments. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 183–189.
17. Nam, C.; Lee, J.; Cheong, S.H.; Cho, B.Y.; Kim, C. Fast and resilient manipulation planning for target retrieval in clutter. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 3777–3783.
18. Nagata K.; Nishi, T. Modeling object arrangement patterns and picking arranged objects. *Adv. Robot.* **2021**, *35*, 981–994. [CrossRef]
19. Grauman, K.; Leibe, B. *Visual object recognition, Synthesis Lectures on Artificial Intelligence and Machine Learning*; Morgan & Claypool Publishers: San Rafael, CA, USA, 2011.
20. Goldman, E.; Herzig, R.; Eisenschtat, A.; Goldberger, J.; Hassner, T. Precise detection in densely packed scenes. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 5227–5236.
21. Asaoka, T.; Nagata, K; Nishi, T.; Mizuuchi, I. Detection of object arrangement patterns using images for robot picking. *Robomech J.* **2018**, *5*, 23. [CrossRef]
22. Rosman, B.; Ramamoorthy, S. Learning spatial relationships between objects. *Int. J. Robot. Res.* **2011**, *30*, 1328–1342. [CrossRef]
23. Panda, S.; Hafez, A.H.A.; Jawahar, C.V. Learning support order for manipulation in clutter. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 809–815.
24. Mojtahedzadeh, R.; Bouguerra, A.; Schaffernicht, E.; Lilienthal, A.J. Support relation analysis and decision making for safe robotic manipulation tasks. *Robot. Auton. Syst.* **2015**, *71*, 99–117. [CrossRef]
25. Grotz, M.; Sippel, D.; Asfour, T. Active vision for extraction of physically plausible support relations. In Proceedings of the 2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids), Toronto, ON, Canada, 15–17 October 2019; pp. 439–445.
26. Zhang, H.; Lan, X.; Bai, S.; Wan, L.; Yang, C.; Zheng, N. A Multi-task Convolutional Neural Network for Autonomous Robotic Grasping in Object Stacking Scenes. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 6435–6442.
27. Temstsin, S.; Degami, A. Decision-making algorithms for safe robotic disassembling of randomly piled objects. *Adv. Robot.* **2017**, *31*, 1281–1295. [CrossRef]
28. Ornan, O.; Degani, A. Toward autonomous disassembling of randomly piled objects with minimal perturbation. In Proceedings of 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 4983–4989.
29. Xu, Z.; He, Z.; Wu, J.; Song, S. Learning 3D Dynamic Scene Representations for Robot Manipulation. In Proceedings of the 4th Conference on Robot Learning (CoRL), Cambridge, MA, USA, 16–18 November 2020; pp. 1–17.
30. Magassouba, A.; Sugiura, K.; Nakayama, A.; Hirakawa, T.; Yamashita, T.; Fujiyoshi, H.; Kawai, H. Predicting and attending to damaging collisions for placing everyday objects in photo-realistic simulations. *Adv. Robot.* **2021**, *35*, 787–799. [CrossRef]
31. Janner, M.; Levine, S.; Freeman, W.T.; Tenenbaum, J.B.; Finn, C.; Wu, J. Reasoning about physical interactions with object-oriented prediction and planning. In Proceedings of the 7th International Conference on Learning Representations (ICLR), New Orleans, LA, USA, 6–9 May 2019; pp. 1–12.

32. Motoda, T.; Petit, D.; Wan, W.; Harada, K. Bimanual Shelf Picking Planner Based on Collapse Prediction. In Proceedings of the 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE), Lyon, France, 23–27 August 2021; pp. 510–515.
33. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
34. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 240–255.
35. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015; pp. 1–14.
36. NVIDIA DEVELOPER. Available online: https://developer.nvidia.com/physx-sdk (accessed on 7 September 2022).
37. Kingma, D.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
38. Shelhamer, E.; Long, J.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 640–651. [CrossRef] [PubMed]
39. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
40. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef] [PubMed]
41. Industrial Robots & Robot Automation Tech | Yaskawa Motoman. Available online: https://www.motoman.com/en-us/products/robots/industrial/assembly-handling/sda-series/sda5f (accessed on 7 September 2022).
42. Choreonoid Official Site. Available online: https://choreonoid.org/en/ (accessed on 7 September 2022).
43. graspPlugin for Choreonoid. Available online: http://www.hlab.sys.es.osaka-u.ac.jp/grasp/en/ (accessed on 7 September 2022).
44. Robotiq: Start Production Faster. Available online: https://robotiq.com (accessed on 7 September 2022).
45. YOODS Co. Ltd. Available online: https://www.yoods.co.jp/products/ycam.html (accessed on 7 September 2022).
46. 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] [PubMed]
47. Avigal, Y.; Berscheid, L.; Asfour, T.; Kräger, T.; Goldberg, K. SpeedFolding: Learning Efficient Bimanual Folding of Garments. *arXiv* **2022**, arXiv:2208.10552.
48. Kartmann, R.; Paus, F.; Grotz, M.; Asfour, T. Extraction of Physically Plausible Support Relations to Predict and Validate Manipulation Action Effects. *IEEE Robot. Autom. Lett.* **2018**, *3*, 3991–3998. [CrossRef]