Learning a Sequential Policy of Efficient Actions for Tangled-Prone Parts in Robotic Bin Picking

Xinyi Zhang\textsuperscript{1*}, Yukiyasu Domae\textsuperscript{2}, Weiwei Wan\textsuperscript{1} and Kensuke Harada\textsuperscript{1,2}

Abstract—This paper introduces an autonomous bin picking system for cable harnesses - an extremely challenging object in bin picking task. Currently cable harnesses are unsuitable to be imported to automated production due to their length and elusive structures. Considering the task of robotic bin picking where the harnesses are heavily entangled, it is challenging for a robot to pick harnesses up one by one using conventional bin picking methods. In this paper, we present an efficient approach to overcoming the difficulties when dealing with entangled-prone parts. We develop several motion schemes for the robot to pick up a single harness avoiding any entanglement. Moreover, we proposed a learning-based bin picking policy to select both grasps and designed motion schemes in a reasonable sequence. Our method is unique due to the novelty for sufficiently solving the entanglement problem in picking cluttered cable harnesses. We demonstrate our approach on a set of real-world experiments, during which the proposed method is capable to perform the sequential bin picking task with both effectiveness and accuracy under a variety of cluttered scenarios.

I. INTRODUCTION

Robotic bin picking is automated in manufacturing in order to pick up and arrange necessary parts from a disordered clutter for further processing. It requires an autonomous and accurate system that can pick up the industrial parts from a bin full of disorganized parts one by one. This paper considers cable harnesses as the target objects, which is a bundle of multiple wires with ties, tubes, adhesive tapes and multi-conductor connectors. Although this product can significantly simplify the assembly process of a machine, it is extremely challenging to be automated on a variety of industrial applications. Given examples of cable harnesses as Fig. 1(a-b) shows, the robot consistently fails the task by picking up entangled and twisted objects following the conventional bin picking motions. This task is well performed under the following conditions - Only one harness is grasped and singulated; The rest of harness can not be dragged outside of the parts bin.

Human can complete this task using some specific strategies. We run a few tests to let human pick up cable harnesses one by one from the box in order to explore the most suitable solution for robotic manipulation. Eventually, we observe that: 1) a motion scheme is highly accurate and time-saving, i.e., a circular trajectory all around the robot workspace so that the grasped cable can be dragged or pulled out of the part bin. Fig. 1(c.1-3) shows the key frames when the grasped harness is separated from others using the action duplicated from human’s demonstration. 2) Human always pick sequentially. Those object that are sigulated or located on the top of the heap have the greatest chance to be picked first.

Therefore, we propose a novel bin picking solution for tangled-prone cable harnesses by imitating human motions and strategies. We design several motion schemes following the circular trajectory to cope with different levels of entanglement. Moreover, an action success prediction network is trained using active learning strategy for predicting the task success given depth images, grasps, and proposed human-imitated actions. Finally, we propose a policy to determine the optimal grasp and action under the current observation to perform the efficient and sequential bin picking task. Our work presents a unique bin picking system equipped with the capabilities to solve the entanglement problem and ensure both accuracy and speed for the task. The proposed method is claimed to be promisingly practical to realize the autonomous manipulation for cable harnesses in production. Experimental results show that our bin picking policy significantly improves the average success rate from 49.2% to 84.6% compared with conventional methods.

\textsuperscript{*}Correspond to: chou@hlab.sys.es.osaka-u.ac.jp
\textsuperscript{1}Graduate School of Engineering Science, Osaka University, Japan
\textsuperscript{2}Industrial Cyber Physical Systems Research Center, National Institute of Advanced Industrial Science and Technology (AIST), Japan
II. RELATED WORK

Robotic bin picking refers to picking up the parts one by one from a container where the parts are randomly placed. A classical bin picking solution is to match the cluttered scene with the known object model or shape to confirm the locations or poses of each object, and plan grasps considering some constraints such as collision and occlusion [1], [2], [3], [4]. Another solution assumes that the information of the target object is unknown and directly explores the grasps on a depth map or point cloud, which leads to more speedy and generalized bin picking [5], [6], [7], [8]. However, these learning-free approaches sometimes suffer from sensory noise, uncertain environments, or complex-shaped parts. Recently, learning-based approaches are widely used over accurate object detection, pose estimation or grasp planning in bin picking to achieve better robustness and generalization [9], [10], [11], [11], [12], [13].

However, there still remain challenges in bin picking. One of the most critical problems is picking up objects with complex shapes. More manipulation techniques are applied to bin picking to handle the objects with unique shapes. Tong et al. present a picking strategy called dig-grasping to simultaneously singulate and pick objects with large width-to-thickness ratios from clutters [14]. They also presents a technique called tilt-and-pivot manipulation [15] and scooping manipulation [16] for picking up thin objects. None of the above approaches consider the case where the complex-shaped objects are entangled together in the container. Let us assume the target objects are easily tangled, such as S-shaped or C-shaped, even if deformable objects. It is impossible for a robot to pick up only one object at a time without considering the entanglement situation. Matsumura et al. tackle the challenge of manipulating potentially-tangled objects in bin picking for the first time [17]. They propose a learning-based approach to predict non-tangle grasping pose among several pre-computed grasp candidates. Zhang et al. propose an analytical method to detect tangle-free grasps using evaluated metrics for entanglement on a single depth image [18]. Moosmann et al. also present a learning-based method to directly predict which objects are entangled and which are not [19]. However, these approaches only contribute to avoiding tangled objects so that they all suffer from the same problem where all objects in the bin are heavily entangled together. Hence, it is necessary to not only avoid the entanglement but also plan practical motions to separate already entangled objects.

III. BIN PICKING FOR ENTANGLED CABLE HARNESS

We are facing a robotic bin picking challenge that even humans have difficulties in completing. Thus, picking up the cable harnesses one by one without entangling with others in the bin and transport them to another container. This section presents our bin picking system setup, grasp detection method, and the implementation of several efficient actions imitated from human demonstrations. Fig. 2 shows the experimental setup for this work. We fix a 3D depth camera at the top of the robot workspace and mount a force/torque sensor on the wrist of the left manipulator. The robot will pick up objects from the bin in the front and transport them to another container located on the left side. We select a parallel jaw gripper to execute the pickings.

A. Grasp Detection

Since the objects are randomly placed in the bin, we must ensure that the gripper can not collide with the other objects while grasping. On the other hand, our target objects - cable harnesses, are deformable and complex-structured so that it is difficult to obtain 3D models. Hence, we select a model-free method in [5] to obtain collision-free grasps on the input depth map. Graspability is an index for detecting a grasping position by convoluting a template of contact areas and collision areas for a robot hand. It is based on the idea that the object should be in the trajectory of the hand closing, and there should be no object in the position to lower the robot hand. We use a parallel jaw gripper and rotate the gripper template along for orientations. The output is a pixel location on the depth map, which refers to the best grasp pose and a rotation angle indicating the orientation of the parallel jaw gripper.

B. Efficient Picking Strategy

Firstly, we investigate how humans perform when separate or discard the entangled cable harnesses. There are various strategies where they execute some extra motions while lifting the grasped objects including shaking, rotating or dual-arm collaboration. From our perspectives, the most suitable motion scheme must meet the needs so it can separate the objects as much as possible and has limited time costs. We find that it is the most efficient that the robot performs a circular motion while lifting the target rather than directly lifting. Additionally, a spinning motion of the robot end-effector can also contribute to the entanglement solution. Designed motion can be considered useful according to the
following reasons: 1) It creates enough space to pull out the long objects; 2) it separates the grasped objects with others in the bin gently from a side angle to ensure that other objects remains in the bin; 3) A spinning motion can solve the situation where the connectors of each cable harness slightly hang on each other. In a word, it is the most suitable motion scheme to pick up entangled cable harnesses among all human solutions we have tested.

Meanwhile, we divide the proposed circular motion into several sub-motions corresponding to the entanglement level of the clutter. If the grasped objects are slightly entangled with others, a motion following a half-circle trajectory will be enough to singulate it. If the grasped objects are heavily entangled in the box, the robot must execute a full circle or -even two-circle trajectory to separate. Under this consideration, we proposed seven circular actions executed by one robot arm corresponding to different entanglement levels. $a_{dl}$ refers to directly lifting the grasped object as the conventional bin picking motion; $a_h$ refers to a half circular motion; $a_{hs}$ adds a spinning motion of the wrist in the end of $a_h$; $a_f$ refers to a full circular motion; $a_{fs}$ is basically $a_f$ plus a spinning motion; $a_{tf}$ denotes a motion following two full circles; $a_{tfs}$ is $a_{tf}$ with a spinning motion. An ordered collection of these seven actions is so-called an action space formulated as $\mathcal{A} = \{a_{dl}, a_h, a_{hs}, a_f, a_{fs}, a_{tf}, a_{tfs}\}$. Specifically, the action space $\mathcal{A}$ also includes a rank of action complexities.

We use $f_{\mathcal{A}(a_i)}$ to describe the action complexity of the action $a_i$. Fig. 3 illustrates all actions by a increased action complexity order.

We implement all actions in bin picking experiments. The centroid and size of the circle motion are pre-defined according to the robot workspace. We sample a set of key points in the trajectory and then plan the motion along with the pre-defined moving speed of the robot arm. To confirm the efficiency of our proposed actions, we prepare four different clutter patterns to execute each action respectively for 20 times. The results in Table I show the success rate and robot execution time. It is confirmed that all actions show a significant improvement in success rate expect $a_{dl}$. We also can observe that as the actions complexity increases, so does the success rate and the execution time. However, even if the action $a_{tfs}$ achieves the highest success rate, it is still time-consuming to implement these actions to all pickings. Alternatively, we have to let the robot learn to pick up the objects with a proper sequence to avoid the entanglement as much as possible.

### IV. Problem Statement

Our goal is to learn a policy on a robot that takes as input a depth image and a collision-free grasp and outputs an action scheme that the robot can pick up only one object. Moreover, we design our bin picking policy to let the robot select
the optimal action every time to perform a sequential bin picking task with limited time and relatively fast speed. We assume a parallel-jaw gripper, entangled harnesses randomly in a container, and a depth image captured from the top viewpoint.

Observation. Let \( x \in \mathbb{R}^{H \times W} \) be a grayscale image with height \( H \) and width \( W \) taken by a camera with a calibrated extrinsic matrix with the robot coordinate. Let \( g \in \mathbb{R}^2 \) denotes a parallel-jaw grasp in the depth image specified by \( g = (u, v) \) relative to a pixel position on the image.

Action. Let \( \mathcal{A} \) be the action space specified by a collection of our predefined motion trajectories. Let the action space \( \mathcal{A} = \{a_1, a_2, \ldots, a_m\} \) also denote to a collection of proposed actions sorted with their complexities, where the complexity can be represented by success rates shown in the Table I.

Success Metrics. Let \( S(x, g, a) \in \{0, 1\} \) be a binary-valued motion success metric representing whether the given action can pick up a single object or not.

Objective. Our goal is firstly learning an action success prediction model \( Q_\theta(x, g, a) \in \{0, 1\} \) parameterized by \( \theta \), which is a function that can discriminate whether if the action \( a \) can solve the entanglement under the current observation \( x \) and \( g \). Next, our bin picking policy \( \pi_\theta(x, g, a) \) is to select the optimal action \( a \in \mathcal{A} \) minimizing the action complexity over an input depth image and a set of grasp candidates.

V. LEARNING SEQUENTIAL BIN PICKING

As shown in Fig. 4, the overview of our proposed bin picking method mainly includes an action success prediction module and sequential bin picking policy module. We first sample several collision-free grasp point candidates using Graspability measure, and then categorize all action candidates from action space. For each grasp-action pair, we use our predictor to estimate if the robot will succeed using the corresponding grasp and action. Finally, after evaluating all grasp-action pairs, we choose the one with the lowest action complexity and more possibility of a successful picking. The robot will execute this picking.

Algorithm 1: Sequential Bin Picking Policy

input: Depth image \( x \), A set of grasp candidates \( G \) computed by Graspability, Action collection \( A \), Prediction Model \( Q_\theta \)

output: Optimal grasp \( g^* \), Optimal action \( a^* \)

1 // Initialize
2 \( \mathcal{P} \leftarrow \text{Empty List} \);
3 for \( g \in G, a \in A \) do
4 \( a \leftarrow \text{categorical vector of action } a \);
5 \( q \leftarrow Q_\theta(x, g, a) \);
6 Append \( \{g, a, q\} \) to \( \mathcal{P} \);
7 Sort \( \mathcal{P} \) in the descending order of \( q \);
8 if \( \hat{p} < 0.5 \) then
9 \( g^* \leftarrow \hat{g}, a^* \leftarrow a_{\text{fs}} \)
10 else
11 Sort \( \mathcal{P} \) in the ascending order of \( f_A(a) \);
12 \( a^* \leftarrow \text{top action of } \mathcal{P} \);
13 \( \{\hat{g}, a^*, \hat{q}\} \leftarrow \text{top element of } \mathcal{P} \);
14 if \( \mathcal{P}.\text{count}(a^*) = 1 \) then \( g^* \leftarrow \hat{g} \);
15 else \( g^* \leftarrow \text{grasp with the highest Graspability score among grasps with the same } a^* \);
16 return \( g^*, a^* \)

A. Sequential Bin Picking Policy

A policy \( \pi_\theta(x, g, a) \) takes predicted action success possibilities of each grasp and action pair and outputs the optimal grasp and action at the same time. We first describe our sequential bin picking policy to foreshadow the training process of the action success prediction network. Algorithm 2 describes our policy. Given the depth map \( x \), we detect several grasp candidates \( G \) using the graspability measure. Other inputs are all actions from the action space \( \mathcal{A} \) and
our trained model $Q_\theta$. Firstly, we predict the action success possibilities for all grasp and action pairs. Action with smaller complexity will be selected among all the successful pairs, as well as the grasp in the same pair. If our model predicts all pairs will fail the task, we select the most complex action $a_{fs}$ to execute. If multiple pairs with the same action complexity are the final candidates, we select the pair with the top graspsability index. The output is an optimal grasp and an optimal action. we select the pair with the top graspsability index. The output is an optimal grasp and an optimal action.

### B. Action Success Prediction via Active Learning

#### Data Collection. As described in Section III, we test our proposed actions on real-world experiments. We use these experimental results combined with extra experiments as our dataset $\mathcal{D}$, where $\{x, g, a, S\} \in \mathcal{D}$. To collect the dataset more efficiently without human labeling, we designed a system to automatically label the action success using a force/torque sensor mounted on the left wrist of the robot. Every time the robot arm is moved to the top of the placing container, the force/torque sensor takes the readings to discriminate whether the gripper is holding a single cable or not. Finally, we collected 710 samples as our dataset. We did not augment our dataset with additional operations. The first reason is that our action-conditional samples are unique so that the basic operations of data augmentation, such as image rotation, are not suitable for our data. We will elaborate on another reason later in this subsection.

#### Network Architecture. We encode the depth image by a convolutional neural network similar to 50-layer ResNet architecture [20]. We apply a single fully-connected layer with 256 units for the grasp point, and a fully-connected layer with 16 units for the categorical action. Then we concatenate them and pass them to a fully-connected layer with 256 units and output the action success possibility. Our model is shown in Fig. 5.

#### Training via Active Learning. For our current dataset, it is observed that the labeled action for each picking may not be optimal. If the picking was successful, the labeled action could surely solve the entanglement, but those actions with lower complexities than labeled action may also succeed. As for the negative samples, the labeled action and those with lower complexities all fail the task. If we train the network directly using the original dataset, even if the trained model can map the labeled data precisely, this model may not achieve optimal performance. If we use this idea to augment our dataset by augment labels for each image, the dataset will get redundant and the training results will disappoint our expectation even more. Therefore, instead of data augmentation, it is necessary to pre-process the dataset to dig the potential prediction ability of our model. We propose active learning to handle this issue. Algorithm 2 shows our active learning method while Fig. 5 also illustrates this process. Firstly, we manually select an initial training data $X$ from the data pool $D$ following the

---

**Algorithm 2: Active Learning Algorithm**

**input:** Data Pool $\mathcal{D}$, Sampling Ratio $\gamma$

**output:** Final model $Q^*_\theta$

1. Select training dataset $\mathcal{X}$ from data pool $\mathcal{D}$;
2. Remove $\mathcal{X}$ from $\mathcal{D}$;
3. Train the network $Q_\theta$ using dataset $\mathcal{X}$;
4. while $\text{len}(\mathcal{D}) > 0$
   5. $q \leftarrow 0$;
   6. $N \leftarrow \gamma \times \text{len}(\mathcal{D})$;
   7. while $q \leq N$
      8. Select example $u = \{x, g, a, s\}$ from $\mathcal{D}$;
      9. $\hat{a} \leftarrow \text{BinPickingPolicy}(Q_\theta, x, g)$;
     10. if $s = 1$
         11. // For positive examples
         12. if $f_A(a) \leq f_A(\hat{a})$ then
             13. Move $u$ from $\mathcal{D}$ to $\mathcal{X}$;
             14. $q = q + 1$;
         15. else // For negative examples
             16. if $f_A(a) > f_A(\hat{a})$ then
                 17. Move $u$ from $\mathcal{D}$ to $\mathcal{X}$;
                 18. $q = q + 1$;
     19. Fine-tuning $Q_\theta$ using the updated $\mathcal{X}$;
20. $Q^*_\theta = Q_\theta$
fact that complex clutter patterns may need more complex actions. We train the network using the initial dataset. Next, we will evaluate the labels for the rest of the data in $D$. We predict their action success possibilities using the trained model under this sample’s observation. The policy mentioned earlier (Algorithm I) is used to output the current optimal action $\hat{a}$. Finally, samples that are logically predicted will be added to the training data $X$. Here is how we define if a prediction is logical or not. For the positive examples, the prediction is logical as the complexity of predicted action $\hat{a}$ is larger than or equal to the labeled action complexity. For the negative examples, we define the logical prediction as to the predicted action $\hat{a}$ is smaller than the labeled action complexity. This label evaluation stage will transfer a certain number of data from data pool $X$ to training data pool $\mathcal{X}$. Our action success prediction network is fine-tuned using the dataset $\mathcal{X}$ repeatedly until data pool $D$ is empty. Finally, we get the optimal model $Q^*_\theta$ as well as our sequential bin picking policy.

VI. EXPERIMENTAL RESULTS

We prepare several real-world experiments to answer the following question: 1) Does the proposed active learning and network architecture learn a better predictor? 2) Does our proposed bin picking method perform more efficient and accurate than baselines? 3) How does our method perform bin picking task both sequentially and efficiently?

A. Experiment Setup

We design for baselines as follows while Final Model is the most optimal model of our proposed method.

- **Graspability**: Use Graspability measure to detect the grasp point, and execute the grasp with the highest graspability score by directly lifting ($a_{d\ell}$).
- **Random Action**: After grasp detection, the action is randomly selected from action space.
- **All Complex Action**: After grasp detection, the robot always takes the most complex action ($a_{tf\ell}$).
- **First Model**: Very first model from our proposed active learning approach.

We prepared two bin picking experiments in real-world. The first one is *consecutive picking* and the goal is to empty the bin filled with respectively 5, 10, or 15 objects. The robot will pick up the object one by one until the bin is empty. Then we will fill the bin again for another round of picking. The second one is *randomized picking* which refers to pick up objects from the bin filled with 18-20, 20-22 and 22-25 objects. After the robot executes one picking trial, we will supply the bin with the exact number of objects to ensure that the number of objects in the bin remains the same during the picking task.

We leverage two metrics to evaluate the bin picking performance. The first one is *success rate*, which is the number of successful pickings from the total number of picking trials. Here, we define successful picking as the robot must perform the task without following situations: 1) The robot grasps no objects; 2) The robot grasps multiple objects inside the gripper; 3) The robot grasps one object and transports more than one object in the place bin; 4) The robot grasps and transport one object while other objects are dragged out of the bin. In this case, if the large part of the object is outside of the bin or contacts with the table, this trial fails. If a small part of the object is slightly outside the bin and does not contact the table, it can also be discriminated as a success. Another metric is **PPH (Pickings Per Hour)**, which is the times of the picking the robot can perform for one hour, multiply the success rate of the corresponding task.

B. Model Performance

First, we validate our active learning model as Fig. 6 shows. We fine-tuned our model three times and we compared the training accuracy and loss of each model. As our active learning loop runs, the model is more accurate on our dataset. The green line indicates the final model, which is the one that we use in the following experiments.

C. Bin Picking Performance

We illustrate our bin picking strategy in Fig. 7 with the depth images, corresponding grasps and actions during a consecutive picking task. Our policy can select relatively simple actions under the current observations and ensure a safe picking without entanglement.

Table II compares the performance of the proposed method and other four approaches in success rate and PPH. Our model achieves the best in both success rate and PPH for both consecutive picking and randomized picking. As the number of objects increases, the bin picking task gets more challenging. However, our final model achieves similar success rates regardless of the number of objects in consecutive picking since the proposed action can solve the entanglement problems in such clutter. The PPH may decrease a little bit since a complex clutter may need more actions with high complexity to ensure a specific success rate. We also can observe that although the model of all complex action reaches the highest success rates, this action is extremely time-consuming so that the PPH is lowest than the final model. It is proved that our active learning model is necessary to improve the efficiency for the bin picking task. As for randomized picking, tasks with more than 20 objects are clearly more complex than consecutive
### TABLE II

**PERFORMANCE OF BIN PICKING EXPERIMENTS.**

| Consecutive Picking | 5 objects | 10 objects | 15 objects |
|---------------------|----------|-----------|-----------|
| Graspability        | 32/50    | 25/50     | 28/50     |
| PPH                 | 128      | 92        | 108       |
| Random Action       | 44/50    | 46/50     | 45/50     |
| PPH                 | 115      | 117       | 124       |
| All Complex Action  | 48/50    | 46/50     | 43/50     |
| PPH                 | 133      | 127       | 143       |
| First Model         | 42/50    | 38/50     | 37/50     |
| PPH                 | 131      | 117       | 111       |
| Final Model         | 44/50    | 44/50     | 43/50     |
| PPH                 | 156      | 140       | 143       |

| Randomized Picking  | 18-20 objects | 20-22 objects | 22-25 objects |
|---------------------|---------------|---------------|---------------|
| Graspability        | 14/30         | 12/30         | 7/30          |
| PPH                 | 93            | 80            | 47            |
| Final Model         | 26/30         | 24/30         | 22/30         |
| PPH                 | 113           | 112           | 103           |

Fig. 7. Picking sequence using our proposed policy. Grasp poses are drawn on each depth image and the red one denotes the executed grasp.

All models perform not good as the consecutive picking since the objects are extremely entangled when the number of the objects increases. But our model remains the most outstanding performance among another baseline, which proves that even in the heavy clutter, our model is also promisingly possible to solve the entanglement problem. Fig. 8 illustrates the full process of the robot executing one of the proposed action $a_{tf}$.

#### D. Generalization

Instead of the objects used in the mentioned experiment, we also generalize our final model to those objects that are not in our training dataset. Fig. 9 shows the bin picking sequence and the corresponding actions. We can observe that this object is a little short than the one in the datasets, but they share a similar structure. In this case, our model can predict the correct action to complete bin picking task. In this case, our model does not predict actions with too large complexities. Instead, it learns to output the optimal actions that exactly pull out the objects at the right timing, which proves that our model can adapt to the cable harnesses that are shorter and share similar structures.

#### E. Discussion

Our circular motion schemes provide a significant advantage in sigulating and picking complex-shaped parts. However, as the number of objects increases, the single robot arm may not be enough to handle a more complex and elusive cluttered scene. In this case, dual-arm manipulation may solve the entanglement instead of fast single-arm manipulation. Meanwhile, our learning-based methods can be generalized on objects shorter than those in the dataset. The proposed actions can not achieve great performance for those long cable harnesses. In the future, we will continue to solve the entanglement problem using more difficult industrial parts by considering more dexterous manipulation schemes using multiple sensory inputs.

### VII. CONCLUSIONS

This work proposes an efficient bin picking approach for the robot to select an efficient action to complete bin picking with extremely entangled objects. We designed a set of circling motions for a robot to drag the target object out of the clutter with a significantly high success rate. Our work is the first to propose a practical bin picking system cope with the entanglement solution for extremely challenging industrial objects. Experimental results show our method exceeds previous approaches in both success rate and PPH. Our method provides a promising solution for entangle-prone parts like cable harnesses to be automated in manufacturing.

### REFERENCES

[1] B. Drost, M. Ulrich, N. Navab, and S. Ilic, “Model globally, match locally: Efficient and robust 3d object recognition,” in 2010 IEEE computer society conference on computer vision and pattern recognition, pp. 998–1005, Ieee, 2010.

[2] C. Choi, Y. Taguchi, O. Tuzel, M.-Y. Liu, and S. Ramalingam, “Voting-based pose estimation for robotic assembly using a 3d sensor,” in 2012 IEEE International Conference on Robotics and Automation, pp. 1724–1731, IEEE, 2012.
Fig. 8. Robot executes the action $a_{tf}$, where the robot arm run two full circular trajectory around the robot workspace.

Fig. 9. Picking sequence using unknown harnesses using our proposed policy.

[3] D. Chetverikov, D. Svirko, D. Stepanov, and P. Krsek, “The trimmed iterative closest point algorithm,” in Object recognition supported by user interaction for service robots, vol. 3, pp. 545–548, IEEE, 2002.

[4] M.-Y. Liu, O. Tuzel, A. Veeraraghavan, Y. Taguchi, T. K. Marks, and R. Chellappa, “Fast object localization and pose estimation in heavy clutter for robotic bin picking,” The International Journal of Robotics Research, vol. 31, no. 8, pp. 951–973, 2012.

[5] Y. Domae, H. Okuda, Y. Taguchi, K. Sumi, and T. Hirai, “Fast graspsibility evaluation on single depth maps for bin picking with general grippers,” in 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 1997–2004, IEEE, 2014.

[6] O. Ghita and P. F. Whelan, “A bin picking system based on depth from defocus,” Machine Vision and Applications, vol. 13, no. 4, pp. 234–244, 2003.

[7] D. C. Dupuis, S. Léonard, M. A. Baumann, E. A. Croft, and J. J. Little, “Two-fingered grasp planning for randomized bin-picking,” in Proc. of the Robotics: Science and Systems 2008 Manipulation Workshop-Intelligence in Human Environments, 2008.

[8] M. Guattieri, A. Ten Pas, K. Saenko, and R. Platt, “High precision grasp pose detection in dense clutter,” in 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 598–605, IEEE, 2016.

[9] D. Fischinger, M. Vincze, and Y. Jiang, “Learning grasps for unknown objects in cluttered scenes,” in 2013 IEEE international conference on robotics and automation, pp. 609–616, IEEE, 2013.

[10] K. Harada, W. Wan, T. Tsuji, K. Kikuchi, K. Nagata, and H. Onda, “Experiments on learning based industrial bin-picking with iterative visual recognition,” arXiv preprint arXiv:1805.08449, 2018.

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

[12] R. Matsumura, K. Harada, Y. Domae, and W. Wan, “Learning based industrial bin-picking trained with approximate physics simulator,” in International Conference on Intelligent Autonomous Systems, pp. 786–798, Springer, 2018.

[13] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, “Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection,” The International Journal of Robotics Research, vol. 37, no. 4-5, pp. 421–436, 2018.

[14] Z. Tong, Y. H. Ng, C. H. Kim, T. He, and J. Seo, “Dig-grasping via direct quasistatic interaction using asymmetric fingers: An approach to effective bin picking,” IEEE Robotics and Automation Letters, vol. 6, no. 2, pp. 3033–3040, 2021.

[15] Z. Tong, T. He, C. H. Kim, Y. H. Ng, Q. Xu, and J. Seo, “Picking thin objects by tilt-and-pivot manipulation and its application to bin picking,” in 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 9932–9938, IEEE, 2020.

[16] T. He, S. Aslam, Z. Tong, and J. Seo, “Scoping manipulation via motion control with a two-fingered gripper and its application to bin picking,” IEEE Robotics and Automation Letters, vol. 6, no. 4, pp. 6394–6401, 2021.

[17] R. Matsumura, Y. Domae, W. Wan, and K. Harada, “Learning based robotic bin-picking for potentially tangled objects,” in 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7990–7997, IEEE, 2019.

[18] X. Zhang, K. Koyama, Y. Domae, W. Wan, and K. Harada, “A topological solution of entanglement for complex-shaped parts in robotic bin-picking,” arXiv preprint arXiv:2106.00943, 2021.

[19] M. Moosmann, F. Spennrath, K. Kleeberger, M. U. Khalid, M. Mönig, J. Rosport, and R. Bormann, “Increasing the robustness of random bin picking by avoiding grasps of entangled workpieces,” Procedia CIRP, vol. 93, pp. 1212–1217, 2020.

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