To appear in *Advanced Robotics*
Vol. 00, No. 00, January 2019, 1–19

FULL PAPER

What are the Important Technologies for Bin Picking? Technology Analysis of Robots in Competitions based on a Set of Performance Metrics

M. Fujita[a†], Y. Domae[b*†], A. Noda[c†], G. A. Garcia Ricardez[d†], T. Nagatani[a], A. Zeng[e], S. Song[e], A. Rodriguez[f], A. Causo[g], I.M. Chen[g], and T. Ogasawara[d]

[a]*Corporate Research and Development, Mitsubishi Electric Corporation, Tokyo, Japan*;
[b]*Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology (AIST), Tokyo, Japan*;
[c]*Faculty of Robotics and Design, Osaka Institute of Technology, Osaka, Japan*;
[d]*Division of Information Science, Nara Institute of Science and Technology, Nara, Japan*;
[e]*Department of Computer Science, Princeton University, Princeton, USA*;
[f]*Mechanical Engineering Department, Massachusetts Institute of Technology, Boston, USA*;
[g]*School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore*

(*v3.0 released November, 2019*)

Bin picking is still a challenge in robotics, as patent in recent robot competitions. These competitions are an excellent platform for technology comparisons since some participants may use state-of-the-art technologies, while others may use conventional ones. Nevertheless, even though points are awarded or subtracted based on the performance in the frame of the competition rules, the final score does not directly reflect the suitability of the technology. Therefore, it is difficult to understand which technologies and their combination are optimal for various real-world problems. In this paper, we propose a set of performance metrics selected in terms of actual field use as a solution to clarify the important technologies in bin picking. Moreover, we use the selected metrics to compare our four original robot systems, which achieved the best performance in the Stow task of the Amazon Robotics Challenge 2017. Based on this comparison, we discuss which technologies are ideal for practical use in bin-picking robots in the fields of factory and warehouse automation.

Keywords: quality metrics; bin picking; manipulation strategy; general gripper; object recognition

1. Introduction

Bin picking is still an important problem in robotics. Its difficulty is described in [1], for example. Picking an item from a cluttered scene is applied to various fields: parts supply in Factory Automation (FA), pick-and-place in Warehouse Automation (WA), cleaning up using household robots and so on. But it was difficult to apply existing methods, if target items have various shape, and materials. These few years, robotic competitions which aim to solve a domain challenges are often held as a platform to accelerate technology development. In the FA field, for example, the National Institute of Standards and Technology of USA is carrying Agile Robotics for Industrial Automation Competition [2] since 2017. It is a simulation-based competition focused on agility. Ministry of Economy, Trade and Industry of Japan is carrying out a Assembly Challenge [3, 4]

*Corresponding author. Email: domae.yukiyasu@aist.go.jp
†Authors contributed equally to this manuscript.

in their World Robot Summit[1] since 2018. It is a real robot's competition focus on factors during setup changes and those of during operation. The former factors are agility and leanness. The latter ones are operation rate improvements. In WA field, Amazon Robotics, Inc. held a competition regarding to a warehouse task automation by robotics in 2017. In large warehouses of e-commerce corporations, a mixture of daily items are manually picked and placed. The automation of the manual work is an important problem in robotic bin-picking. In particular, technical problems lie in picking items with various shapes and in identifying their texture, shape, and material. Various methods have been proposed to solve the problems in the competition in which the ability of the robot system was tested in a competition setting. In this paper, we show a system comparison of the four unique teams which ranked first to fourth places in the Stow task, Amazon Robotics Challenge 2017. In the competition, the systems were ranked according to the organizer's rules. But we have difficulty in analysis of a system performance in terms of more practical use. The reason is as follows. A single metric like the competition's score is a kind of principal component analysis [5], thus dimensionality reduction always lose some information about original data that reflects their performance. While the needs of each industry are different, competitions reflecting the needs of each industry are taking place. In this paper, we propose a set of metrics in order to reflects a detail behavior of a system to our system performance analysis as a solution. Then we describe an example analysis show the issues that arise when robots are applied to actual factories or warehouses by using the comparison results. Also, items in each system to be improved to get more performance will become clear.

The main contributions of this paper are the following:

- We show the details of four unique robot systems and compare the system configurations of each team.
- We proposed a system performance evaluation based on plural metrics introduced from reliability engineering [6, 7]. By this analysis, each team strategy was revealed.
- We discuss on pros and cons of the systems and technologies including the details of the systems, and also discuss on what we have learned and on future system design.

The rest of this paper is organized as follows. Section 2 presents the related works. Section 3 introduces the Stow task. Section 4 describes four systems developed for this task. Section 5 presents our proposed system performance evaluation. Section 6 analyzes our findings and presents lessons learned. Finally, Section 7 concludes this paper.

2. Related works

The related works are presented in five groups. The first group includes systems developed for the bin-picking tasks. The second group summarizes different design approaches for grippers. The third group is on grasp planning. The fourth group is on item recognition. Finally, the fifth group is about comparison analysis of competition systems.

2.1 *Bin-picking systems related to competitions*

Bin-picking is a classical but still state-of-the-art challenge in robotics. Many proposals were made on the automation of pick-and-place tasks in warehouses in the Amazon Picking Challenge (later known as Amazon Robotics Challenge), held from 2015 to 2017. In particular, there were many proposals and findings on gripper design to solve problems when picking various items as described in later sections. In the first competition held in 2015, actual shelves used in Amazon warehouses were also used in the pick-and-place of daily items from a bin.

[1] https://worldrobotsummit.org/en/

As mentioned in the summary article of the 2015 competition [8], it was proven that a suction gripper is able to pick many kinds of items. Further in the 2016 competition, the winner Delft [9] and many other teams succeeded in picking hard-to-pick items such as a mesh cup, which a suction gripper was unable to handle, by combining suction and two-finger grasping or similar pinching mechanisms.

2.2 *Gripper design*

For grasping, the combination of suction and two-finger or suction only became the common configuration. In the 2017 competition, almost all the teams used either of the two-gripper configurations mentioned above. The overall winner of 2017 [10] as well as team NAIST-Panasonic (Garcia *et al.* [11]), who configured the system by analyzing the past competitions and came fourth at their first attempt, adopted the gripper configuration of suction and two-finger combination. Team MIT-Princeton [12], the winner of Stow task in 2017, where only bin-picking ability was tested, made a system which enabled several motions such as suction down, suction side, grasp down, flush grasp in one gripper system that combined suction and two-finger grippers. Only the runner-up in Stow Task, Team Nanyang [13], used a configuration with two suction grippers and without a two-finger gripper. They achieved a high score by focusing on bin rather than gripper design. To explain in detail, they added a mechanism which expanded the bin space thereby modifying the problem from a hard bin-picking to a simpler picking like a pick-and-place problem from a wide open flat space. The team was successful in picking various items, and their strategy was necessary for items in a hard-to-pick pose or occluded. Team MC^2 proposed a two-stage strategy to use three types of gripper properly [14]. As aforementioned, many types of gripper designs were proposed in the competitions.

In comparison, jamming gripper [15] is highly versatile in picking various kinds of items. Nevertheless, there are some difficulties in applying jamming grippers to bin-picking because when a jamming gripper tends to pick several small items in a tightly packed bin simultaneously. Also, in principle, as it needs to come into contact with the item before it starts grasping, it tends to fail in picking soft items which may change shape easily. Thus, nobody used the jamming gripper in the competitions.

2.3 *Grasp planning and grasp point detection*

Grasp point detection takes place to determine the gripper pose to pick detected items. A method [16] which convolutes a binary image model of the gripper to the depth image and does not require pre-information of the object is already used in factory automation. Many methods for grasp point detection using machine learning from RGB images and depth images have been proposed. Jiang *et al.* proposed a method [17] which searches in an RGB image for a pose that is easy for a two-finger gripper to grasp and they were the first to make it practical [18] with deep learning. Pinto *et al.* proposed a grasp point detection method [19] from an RGB image based on 50,000 trials on actual robots. Moreover, Levine and others achieved a method [20] where hand-eye coordination detects a grasp point from RGB data. The common among all these methods is that they are able to determine the grasp point using only images. Unfortunately, the physical correspondence between gripper and item in grasping [21] cannot be understood just from appearances in an image. Mahler and others have defined a matrix which determines grasp points for a number of objects in advance from the relationship between the 3D object model and the 3D gripper model and propose a method [22] which assumes a physical grasp point of unknown objects by learning from a vast amount of data. They achieved this with deep learning [23] and used it with vacuum and suction[1] type of grippers [24]. In this method,

[1] In this paper, we refer to the blower-based suction as *suction* and vacuum-pump-based suction as *vacuum*.

bin picking is available based on the learned results when the learning becomes precise enough. Furthermore, Matsumura and others succeeded in bin picking with real robots exclusively by learning from simulation data [25]. Team MIT-Princeton [12] and others fitted for both suction and two-finger grippers by detecting grasp points with Fully Convolutional Network (FCN) on the base [12]. Whether to use a learning method by providing data beforehand or to use a non-learning method which is more adaptive to unknown objects and environmental changes depends on the preconditions of the problem.

2.4 *Sensors and algorithms for item recognition*

In the competition, item detection based on images obtained from RGB-D sensors is often used. The winning team in 2015 probabilistically classified multi-class items with a method which describes the type of item in each pixel using image features obtained from RGB and depth images [26]. Then, items are segmented by integrating the result. In the 2015 competition, many teams used algorithms based on image features. From 2016 onward, many adopted Convolutional Neural Networks (CNN) and showed good performance. Faster network variants such as Faster R-CNN [27] which performs bounding box detection and multi-class classification in order, and high speed YOLO [28] and SSD [29], which perform the detection and classification in parallel, were used for the recognition. There were teams [10] who used semantic segmentation methods as a base and all performed well in detecting (classifying) items in the bins.

Object pose detection is used to determine grasp point on items after they are detected. In general, data obtained from an RGB-D sensor and object model are matched together using methods such as Iterative Closest Point [30], which minimizes the point cloud position errors between data and model, and Directional Chamfer Matching [31], which presumes object pose by featuring image edges and matching them for every view direction. Such methods are used for bin-picking in factory automation as they are robust against illumination changes, among other things. The methods [32, 33], which estimate object pose by voting after extracting features and matching pairs of vectors obtained from edges and planes of object, are good in speed and accuracy balance. A method [34], which assumes the position and approximate pose of an object by adding multi-view data to a CNN, is also proposed. Nevertheless, to use these methods, a 3D model of the object is required. If a 3D model is not available, a method by the Team MIT-Princeton [35], which assumes the object pose by fitting primitive shapes like spheres and cubes directly to the data, is also proposed. The method to use depends on the precondition of the problems.

2.5 *Comparison analysis of competition systems*

In the Amazon Robotics Challenge, points are awarded or subtracted based on the performance in the frame of the rules. But the final score does not seem to directly reflect the technology suitability. Therefore, organizers and teams published papers about system analysis [8]. Results of the analysis are not based on the scores in the competition, but statistical data by a questionnaire survey about used technologies, team configurations, and so on. From the results, we can understand which technologies were well used in the competition. But we have difficulty in analysis of a system performance in terms of more practical use.

Successful picking rate as shown in [14, 16] or well-known cycle time[1] is an important metric to evaluate the system performance in practical use. Mean Picks Per Hour (MPPH) [22] is also a well-used metric [12, 24] which is related to both picking rate and picking time. But such a single metric is a kind of principal component analysis [5]. Thus dimensionality reduction always lose some information of original performance data. Therefore, in this paper, we propose

[1] Originally, Adept company had proposed this metrics. It is a round trip time or cycle numbers per a minute of a robot TCP trajectory which goes 25 [mm] up, 305 [mm] horizontal, 25 [mm] down.

Figure 1. Example of items in a tote for the Stow task.

a set of metrics in order to analyze a system performance that reflects different aspects of a system behavior. We calculate metrics as below. **Average time per pick** and **Number of trials per hour** for comparison of a bare picking action speed, **Mean Time Between Failure (MTBF)** which is expected to compare a validity of a control algorithm that can keep up with the phenomenon, **Mean Time to Repair (MTTR)** expected to reflects skills of recovery operation when work error occurs, and **Availability** that can expect a comparison of duty rate of normal operation and recovery operation, in addition to MPPH, for each team. These introduced metrics [6] are originally used in a field of reliability engineering (for ex. [7]). The details of the metrics will be shown in Chapter 5. The purpose of introducing a set of these metrics is primarily a comparison of the detailed functionality and performance of each system as described above. The true goals are an uncovering of system design strategies, a discovery of measures to obtain better functionality and performance for them, and a discovery of legitimate technologies for the purpose.

3. Problem setting of the Stow task

We show an outline of the Stow task rules in the Amazon Robotics Challenge 2017.

- Robots must pick various daily items automatically from a tote and place them into a storage.
- Items are mixed in a tote as shown in Fig. 1. Robots must identify the items and record which items are placed into a bin in the storage.
- If a robot can pick and classify an item, then place the item into a bin in the storage and record its location successfully, points are awarded.
- If items are dropped or damaged while picking and placing, points are subtracted.
- 20 items are in a tote. Half of the items are distributed to teams in advance. The other half of the items are distributed just before the starting the competition. The recognition dataset must be updated in a few minutes.
- Robots must finish the task in 15 minutes. If robots finish the task before the 15-minute period, teams get additional points.
- Storage can be designed by each team with some limitations: size, number of bins and so on.

We also describe the details of the scoring system in the rules. Points are awarded as follows:

- 5 points for each item stowed into the storage system, plus 5 additional points if the item is a new (unknown) item.
- 1 point for every 5 seconds or fraction thereof that remain on the clock when the task is

complete and all items have been successfully stowed in the storage system, so long as at least 15 of the locations of the items are correctly registered in the item location file.

Points are subtracted as follows:

- -15 points for each item that is not in the storage system, stow tote, or amnesty tote at the end of the task, except for items grasped by the robot under normal operation when time runs out.
- -5 points for each item in the storage system or stow tote with an incorrect final location in the item location file.
- -5 points for any item that is dropped into the storage system from a height of more than 15 cm.
- -5 points for each item that is protruding more than 2 cm out of the storage system.
- -5 points for minor damage to an item, such as bends and dents.
- -20 points for major damage to an item, such as large rips, holes, or crushing.

More details, please check the official site[1].

4. Four proposed systems for the Stow task

4.1 *Team MIT-Princeton (1st place)*

The MIT-Princeton system [12] consists of a 6DOF ABB IRB 1600id robot arm next to four picking work-cells (see Fig. 2a). The robot arm uses a multi-functional gripper with two fingers (built on top of a Weiss WSG 50 gripper) for parallel-jaw grasps and a custom retractable suction cup. The gripper is designed to function in cluttered environments: finger and suction cup length are specifically chosen such that the bulk of the gripper body does not need to enter the cluttered space. One gripper fingertip is equipped with a GelSight tactile sensor, while the other fingertip uses an actuated fingernail for scooping along the sides of storage bins. Each work-cell consists of a storage bin, as well as four fixed-mounted RealSense SR300 RGB-D cameras: two cameras overlooking the storage bins (positioned on opposite sides) are used to infer grasp points, while the other two pointing towards the robot gripper (also positioned on opposite sides) are used to recognize objects in the gripper. Each work-cell also includes a force sensor underneath for 1) checking the weight of picked objects, and 2) detecting collisions.

The system is built around a grasp-first-then-recognize pipeline. For each pick-and-place operation, it uses fully convolutional networks (FCNs) to take as input RGB-D images of the work-cell, and output pixel-wise confidence scores (i.e., affordances) of four different motion primitives for picking (see Fig. 2b): top-down suction, side suction, top-down grasp, side-flush grasp. Each pixel of the output represents a suction or parallel-jaw grasp centered at the 3D location of that pixels corresponding surface in view (Fig. 2c). The FCNs are trained using a dataset of 1,837 RGB-D images of cluttered work-cells, with good/bad grasp locations manually annotated by human experts. During inference, the system selects and executes the motion primitive with the highest predicted confidence score, picks up one object, isolates it from the clutter, holds it up in front of cameras, recognizes its category, and places it into the appropriate bin. The recognition algorithm uses a two-stream network to learn a common feature embedding space between 1) observed images of held objects, and 2) product images – where images of the same object match to more similar output features. Since this network architecture does not rely on knowing the number of object categories beforehand, it is capable of recognizing images of novel objects unseen during training by matching them to corresponding product images

[1] https://www.amazonrobotics.com/site/binaries/content/assets/amazonrobotics/arc/2017-amazon-robotics-challenge-rules-v3.pdf

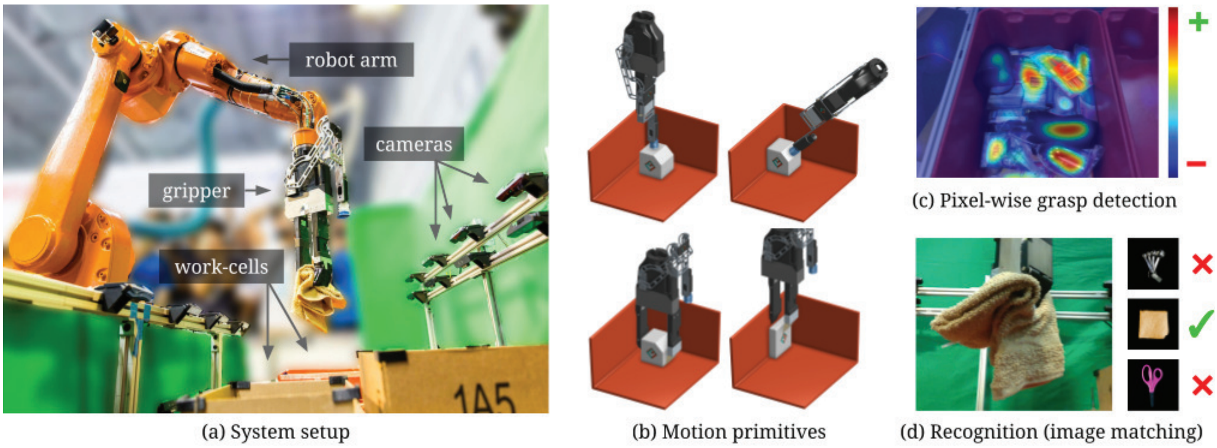

Figure 2. The MIT-Princeton system setup (a) consists of a 6DoF robot arm next to four picking work-cells. The system uses (c) FCNs to predict pixel-wise grasping confidences scores (*i.e.*, affordances) of (b) four motion primitives using suction and parallel-jaw grasping. After executing the motion primitive at the 3D location of the pixel with the highest confidence score, the system picks up an object and uses (d) a two-stream network to match images of the held object to the most similar product image for recognition.

that are provided at test time (Fig. 2d). Prior to the competition, the network is trained over observed-image-to-product-image pairs of known objects.

This system design has several advantages. First, the FCN-based grasping algorithm is model-free and agnostic to object identities. It detects grasps by using local geometric and texture features on objects, allowing it to learn biases that can generalize to novel objects without retraining (e.g. flat surfaces are good for suction, porous surfaces are bad for suction, etc.). Second, the object recognition algorithm works without task-specific data collection or retraining for novel objects, which makes it scalable for applications in warehouse automation and service robots where the range of observed object categories is large and dynamic. Third, our grasping framework supports multiple grasping modes with a multi-functional gripper (suction and grasping) and thus handles a wide variety of objects. Finally, the entire processing pipeline including grasp detection and recognition requires only a few forward passes through deep networks and thus executes quickly (a few hundred milliseconds in total per pick-and-place).

4.2 *Team Nanyang (2nd place)*

The team formed by members of the Nanyang Technological University (Singapore) developed a dual-arm robot equipped with suction-based grippers and a top-open drawer-like storage system.

The robot system features two identical manipulators (Universal Robots UR5), three stereo cameras (Stereolabs ZED) and two custom-built grippers. The built system is shown in Fig. 3(a) together with its system architecture shown in Fig. 3(b).

We divide the workspace into two individual and one shared work cell to optimize the manipulation performance and decrease the risk of collision between the manipulators.

Our shelf has two bins which temporarily extend sideways in order to disperse the cluttered pile of items. This allows the system to have easier access to the items and to facilitate the object detection by decreasing occlusion.

For object detection, we use the results of either one of two classifiers, one based on engineered features and the other based on learned features, whichever has the highest confidence. This is because we expect higher confidence for unknown items from the former, and higher confidence for known items from the latter. As engineered features, we use Grid-based Motion Statistics (GMS) [36], which is a feature detection algorithm similar in principle to SIFT but superior in performance. The learned-features are extracted using CNNs.

The grippers are suction-based since over 98% of the training items were successfully grasped using our modified suction cups. Our grasping strategy consists mainly on approaching the

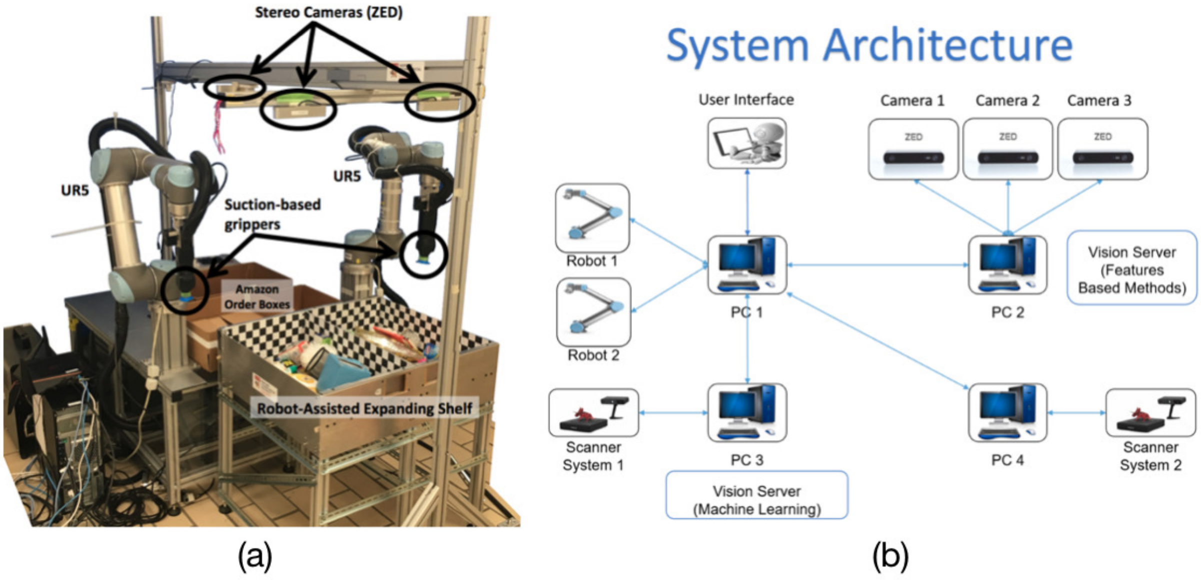

Figure 3. Team Nanyang's system (a) and its system architecture (b).

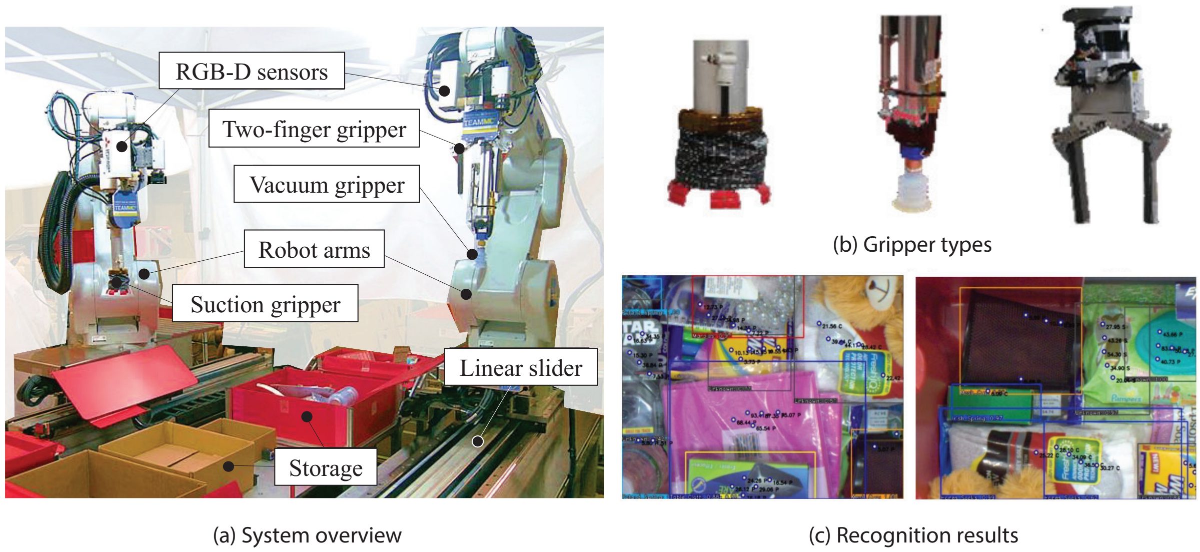

Figure 4. Robot system by team MC^2. (a) the system overview. (b) The system has three different types of gripper: suction, vacuum and two-finger. (c) SSD-based object detection and 3D pose estimation algorithms can detect the graspable items and the grasping points from cluttered scenes.

objects straight down from the top, which is effective for almost 98% of the items.

4.3 *Team MC^2 (3rd place)*

The team MC^2 (Mitsubishi electric corporation, Chubu university, Chukyo university) is shown in Fig. 4 (a). Two robot arms are mounted on linear sliders, facing each other with the item bins in the center in between them. Each robot arm is able to operate individually and has an RGB-D sensor and a force sensor. The RGB-D sensor is used for item and picking position detection. The recognition algorithms are based on SSD, graspability, and primitive shape matching for item detection and classification, gripper pose detection, and item pose estimation separately as shown in Fig. 4 (c). The force sensor is used for force control when the robot picks and places items. The proposed robot system has three different types of gripper: suction, vacuum and two-

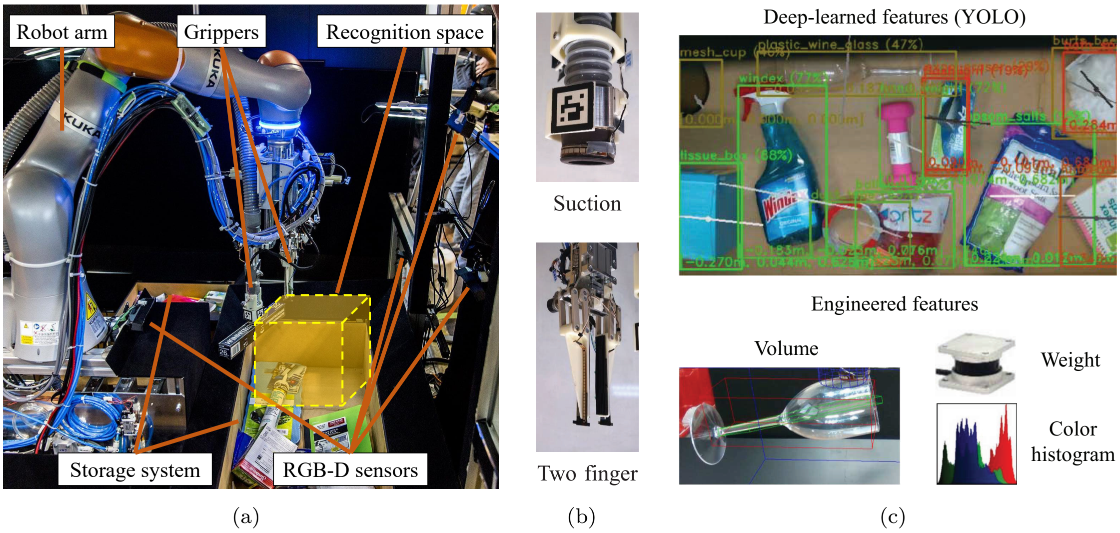

Figure 5. Bin-picking system proposed by team NAIST-Panasonic. (a) shows the system overview, (b) shows the suction and two-finger grippers, and (c) presents the learned and engineered features used for item recognition.

finger as shown in Fig. 4 (b). The suction gripper is mounted on the left-side robot, as shown in Fig. 4 (a). The vacuum and two-finger grippers are mounted on the right-side robot, as shown in Fig. 4 (a). The two-finger gripper is used after removing the vacuum gripper by using a tool changer mechanism.

We devised a strategy in which the gripper combination changes accordingly. As bins are crowded and items are on top of each other, vacuum gripper, which picks items in smaller a surface area for picking, is preferred. As items are both large and small, the suction gripper, which is able to pick large items once it recognizes the surface, is also suitable. In contrast, collision due to item crowding inside the bin must be considered for the two-finger gripper and it is hard to obtain a pose for grasp positioning in a crowded bin. Therefore, the combination of vacuum and suction gripper is chosen for the beginning and middle part of the picking process.

When items are isolated, two-finger gripper can reach a grasp pose more easily. Besides, the value of two-finger gripper rises because the remaining items are hard to pick with vacuum and suction grippers used at the beginning. What is more, the more sparse the items get, the risk of picking several items also decreases, and approaching items in hard-to-pick poses becomes easier. Thus, the suction gripper is also adopted. To achieve the strategy explained so far, we configured a robot system in which one robot arm has a suction gripper and the other has vacuum and two-finger grippers, as described in [14]. The grippers are switched with a tool changer.

4.4 *Team NAIST-Panasonic (4th place)*

Team NAIST-Panasonic is formed by the Nara Institute of Science and Technology (NAIST) and Panasonic Corporation and include members with experience in robotics competitions [37].

The proposed solution consists of a 7-DOF robot arm (KUKA LBR iiwa 14 R820) with a custom-made end effector, a controlled space (recognition space) with four RGB-D cameras, and a shelf (storage system) with weight sensors underneath [38]. The setup of the proposed bin-picking solution is shown in Fig. 5(a).

The end effector has a suction gripper and a two-finger gripper, shown in Fig. 5(b), mounted on two separate linear actuators, and an RGB-D camera to recognize items and estimate the grasping points. The suction gripper consists of a compliant vacuum cleaner hose which is partially constrained to reduce swinging when transporting items. The two-finger gripper has high-friction rubber on its parallel fingers and is used as a secondary grasping tool. Both grippers include force-sensitive resistors to detect collisions with items and force control to avoid damaging items. The smart design of the end effector provided a reliable and consistent performance. The high flow and compliance of the suction gripper reduced the negative effects of vision and motion planning errors, making the system able to pick and transport the items safely.

The recognition space consists of four RGB-D cameras (Intel Realsense SR300) pointing at a space over the storage system, where eight LEDs control the illumination and the background of the cameras' views is controlled using non-reflective black plates. We combine learned and engineered features, shown in Fig. 5(c), to achieve a robust object recognition for both known and unknown items. This was particularly useful in the case of the combination of bounding box volume and weight for clamshell-type and deformable items.

The strategy to recognize an item is: 1) point the end effector's camera to the target container, perform object detection using YOLO v2 [28] and grasping point estimation using RGB images, and pick the item with the highest recognition confidence; and 2) move this item into the recognition space to confirm or reject the initial belief using SVMs for single or combined engineered features (color histogram, bounding box volume, and weight) trained with data collected at approximately 90 seconds per item. A weight is assigned to each learned or engineered feature to adapt object recognition to the task requirements, physical characteristics of target items, and so on, resulting in a voting system that determines the final item class.

We designed the system to overcome failures by quickly detecting the most common errors and by preparing recovery behaviors in advance. This allowed us to retry failed grasping attempts in a short time. Furthermore, the *recognize-while-holding* concept of the recognition space increased the robustness of the system to accidentally-dropped and unrecoverable items which could critically compromise the object recognition capabilities.

4.5 *Comparison of system configurations*

We show the system configurations of each system in Table 1. The main similarities which can be understood from this table are:

- All systems are based on industrial robots because accuracy and speed are important factors to complete the task. Industrial robot's high accuracy may be excessive but some collaborative robots are difficult to use for the Stow task because of low accuracy.
- Almost all teams based their grippers on suction (or vacuum) and two fingers. Suction-based grippers can pick many items including deformable objects but they are difficult to apply to mesh items (*i.e.*, air-permeable items). On the other hand, two-finger grippers can pick mesh items. Therefore, the combination performs well.
- The item recognition of the systems is mainly based on RGB-D sensors and CNN-based algorithms. Open source computer vision implementations are easy to use for researchers of the robotics field and perform well enough.

The main differences are:

- The number of robots and their degrees of freedom are different. All teams basically pick items from above the storage system with 4 DoF. Therefore, 6 DoF should be enough. Robot systems can have many robots but systems with too many robots are hard to implement and are more prone to collision problems. Therefore, many teams use systems with fewer robots.
- Some teams used force sensors, weight sensors, visual-tactile sensors (GelSight), and so on. These sensors seem to be useful for the Stow task but the implementation may be difficult, mainly because there are very few useful open source projects to help with the implementation.

Table 1. System configuration of each team.

	MIT-Princeton	Nanyang	MC^2	NAIST-Panasonic
Robots	One 6-DoF robot arm	Two 6-DoF robot arms	Two 6-DoF robot arms on 1-DoF linear sliders	One 7-DoF robot arm
Sensors	Sixteen fixed-mount RGB-D sensors, four force sensors (below bins), one tactile sensor (GelSight on gripper), and one air pressure sensor	Three RGB-D sensors	Two RGB-D sensors and two force sensors on robot arms, and two weight sensors	Five RGB-D sensors, two weight sensors, two FSR-based contact sensors, and one air pressure sensor
Grippers	Multi-functional gripper with two fingers for parallel-jaw grasps and a retractable suction cup	Two suction-based grippers	One large suction gripper, one small vacuum gripper, and one two-finger gripper	One suction gripper and one two-finger gripper
Recognition algorithm	Two FCNs to infer grasping points for both suction and parallel-jaw grasping, and a two-stream network to match real images of objects to product images for classification	Mixed-mode classifier using feature extraction (GMS) and CNN	SSD-based item detector and classifier from a RGB image, gripper pose detector from a single depth map, and 3D pose estimator from a point cloud data	Multi-modal weighted voting classifier using learned and engineered features (YOLO from RGB, volume from depth, weight, and color histogram)
Unique features	Learning visual affordances for multi-functional gripping (grasping and suction)	Top-open extendable shelf design	Using three types of grippers and its combination strategy	High-flow suction gripper and fast failure recovery
# of robot arms	1	2	2	1
# of sensors	22	3	6	10
# of grippers	1	2	3	2

5. Performance evaluation

5.1 *Metrics*

We have previously compared two systems (team MIT-Princeton and team MC^2 at the Amazon Robotics Challenge 2017) using Mean Picks Per Hour (MPPH) [22] and our original successful picking rate as a robot performance index in [14]. MPPH is currently widely-used as a bin-picking system performance metric [22, 39]. Since a high correlation to the score of the competition is seen with MPPH, it is suitable as a performance comparison index of several systems developed for the Amazon Robotics Challenge 2017.

Comparing system performance using a single number is similar to the results of the Principal Component Analysis [5], which is a method used for dimensional reduction of multidimensional data. The principle is to reduce the original amount of information by choosing the component with the largest possible variance. In that sense, many performance indicators provide the first principal component innately. However, if the first principal component is chosen for each measurement data, there is the problem that it can only be used for a relative comparison. A solution to this problem is to evaluate the system performance in a comprehensive way by calculating several absolute indices for each system.

In this paper, we calculate **Average time per pick**, **Number of trials per hour**, **Mean Time Between Failure (MTBF)**, **Mean Time to Repair (MTTR)**, and **Availability**, in addition to MPPH, for each team.

By evaluating each system individually and comprehensively, we analyze their system design policy and system performance in a multifaceted manner. In each system, their subsystems work differently as they are designed differently, then multiple metrics will reflect the difference between systems.

MPPH is calculated by multiplying the **Number of trials per hour** and the **Average probability of success**.

Average time per pick is measured from the competition video of each team. In this metric, *pick* refers to the motion from capturing object data by the sensors to pick an item and place it. Then, **Number of trials per hour** can be calculated by Average time per pick.

The **Average probability of success** is also measured from the competition video of each team.

MTBF stands for Mean Time Between Failures and is calculated by dividing the operating time by the failure count of the system.

$$\mathrm{MTBF} = \frac{T_{\mathrm{up}}}{N_{\mathrm{down}}}\,, \tag{1}$$

where T_{up} is the duration of the system running well, and N_{down} is the number of times the system fails and takes some recovery actions until it succeeds. These were obtained from the videos recorded during the Amazon Robotics Challenge 2017.

MTTR stands for Mean Time to Repair and is obtained as follows:

$$\mathrm{MTTR} = \frac{T_{\mathrm{down}}}{N_{\mathrm{down}}}\,, \tag{2}$$

where T_{down} is the duration from the beginning of the failure to the end of the recovery actions.

Availability is obtained as follows:

$$\mathrm{Availability} = \frac{\mathrm{MTBF}}{(\mathrm{MTBF} + \mathrm{MTTR})}\,. \tag{3}$$

Availability is a dimensionless quantity, which represents the ratio between the time that the system is running well and the total time the system is operating.

Table 2. Results of metrics calculation of each team.

	MIT-Princeton	Nanyang	MC^2	NAIST-Panasonic
Score based on ARC* rules	160	125	120	110
Number of trials	32	37	28	50
Number of successes**	19	16	18	17
Number of errors	13	21	10	33
Successful picking rate	0.594	0.432	0.643	0.340
Average time per pick [sec]	23.1	24.3	32.1	18.0
Number of trials per hour	156	148	112	200
Average probability of success	0.594	0.432	0.643	0.340
MPPH	92.6	64.0	72.0	68.0
Sum of up time [sec]	535	504	488	437
Sum of down time [sec]	204	396	412	463
Total time [sec]	739	900	900	900
MTBF [sec]	28.2	31.5	27.1	25.7
MTTR [sec]	15.7	18.9	41.2	14.0
Availability	0.642	0.626	0.397	0.647

* Amazon Robotics Challenge 2017.
** Successful sequences of pick, move, and place.

Table 3. Normalized results of selected metrics based on highest score team (team MIT-Princeton).

	MIT-Princeton	Nanyang	MC^2	NAIST-Panasonic
Score based on ARC* rules	**1.00**	0.78	0.75	0.69
Average time per pick	1.00	1.05	**1.39**	0.78
Number of trials per hour	1.00	0.95	0.72	**1.28**
Average probability of success	1.00	0.73	**1.08**	0.57
MPPH	**1.00**	0.69	0.78	0.73
MTBF	1.00	**1.12**	0.96	0.91
MTTR	1.00	1.20	**2.63**	0.89
Availability	1.00	0.97	0.62	**1.01**

* Amazon Robotics Challenge 2017.

5.2 *Performance evaluation using the proposed metrics*

We calculated the proposed set of metrics and relative values, as shown in Table 2. Then, we normalized the most significant metrics based on the winning team (team MIT-Princeton), as shown in Table 3. We also show these normalized results in Fig. 6.

From Fig. 6, we observe that MPPH and the scores are highly correlated, which makes it suitable as a comprehensive performance index in that sense. We consider that the slight deviation is caused by a difference in scoring when including a bonus point. However, the other metrics are considerably fluctuating.

6. Discussion

6.1 *Analysis of the performance comparison*

In this section, we analyze the results shown in Figure 6. We consider changes of each metric in comparison to the actual system implementation, and explore the system design concept.

MPPH is a good indicator that represents system performance, as evident in the fact that **MPPH** and the score based on the rules of the Amazon Robotics Challenge are similar. Hereafter, we examine the factors constituting the **MPPH**, namely, the **Number of trials per**

Table 4. Number of picked items and the successful picking rate of each gripper. We refer to the blower-based suction as *suction* and vacuum-pump-based suction as *vacuum*. In this paper, dropped items during pick-and-place do not count as successful picking.

MIT-Princeton				
	Suction	Vacuum	Two-finger	Total
Number of picked items	13	-	6	19
Successful picking rate [%]	54.2	-	75	59.4
Nanyang				
	Suction	Vacuum	Two-finger	Total
Number of picked items	16	-	-	16
Successful picking rate [%]	43.2	-	-	43.2
MC^2				
	Suction	Vacuum	Two-finger	Total
Number of picked items	9	6	3	18
Successful picking rate [%]	64.3	100	37.5	64.3
NAIST-Panasonic				
	Suction	Vacuum	Two-finger	Total
Number of picked items	17	-	0	17
Successful picking rate [%]	34	-	0	34.0

hour and **Average probability of success**. First, **Number of trials per hour** is very low for MC^2, while it is high for team NAIST-Panasonic. The other two teams are in the middle. When we look closely at the system design of each team, MC^2, for example, has a hand-eye system with a vision sensor attached to a wrist of their robot, and it is configured to perform the vision sensing operation and the other operation sequentially. In other words, the recognition operation is performed after the completion of the stow operation, which is one cycle before, then, the picking operation starts. Therefore, one cycle takes an amount of time while the other teams can perform the previous stow operation and the recognition operation concurrently and shorten their cycle times.

For the posture variation of the grasping object, the team NAIST-Panasonic uses only one hand with a vertical downward trajectory, and the grasping operation starts immediately after recognition is completed. The operation itself is simple, which leads to an increase in the number of attempts per unit time. The other two teams, especially the team MIT-Princeton, have more than one type of gripping trajectory, and it takes longer than the team NAIST-Panasonic to plan and operate. The difference of these system designs is reflected in the number of trials per hour.

In terms of the **Average probability of success**, MC^2 and MIT-Princeton are comparable, while NAIST-Panasonic and Nanyang have lower values. As a system design concept, the difference lies in whether it is thought that every single operation is important, and the system attempts to score by the number of retries even if it fails somehow.

In terms of the **MTBF**, Nanyang team is somewhat larger, while the other teams are almost equal. In other words, since the success possibility is slightly higher than the other teams, Nanyang team can keep their normal operation for longer.

Considering **MTTR**, MC^2 team has a large value. The reason for this is that it is disadvantageous in recovering, it takes long time in one operation, and fails many times at the same object. This can lead to imagine that the recognition method and the grasping method are too naive to succeed (*i.e.*, the system makes the same mistakes). The team NAIST-Panasonic is expected to have a strategy switch that succeeds in recovering at high speed. Teams Nanyang and MIT-Princeton are moderate, but MIT-Princeton is a little dominant. MIT-Princeton tries different ways for every fail, each seems reasonable. This enables quick recovery. Nanyang team did grasping point change for each error. This worked well most of the times. Team NAIST-Panasonic

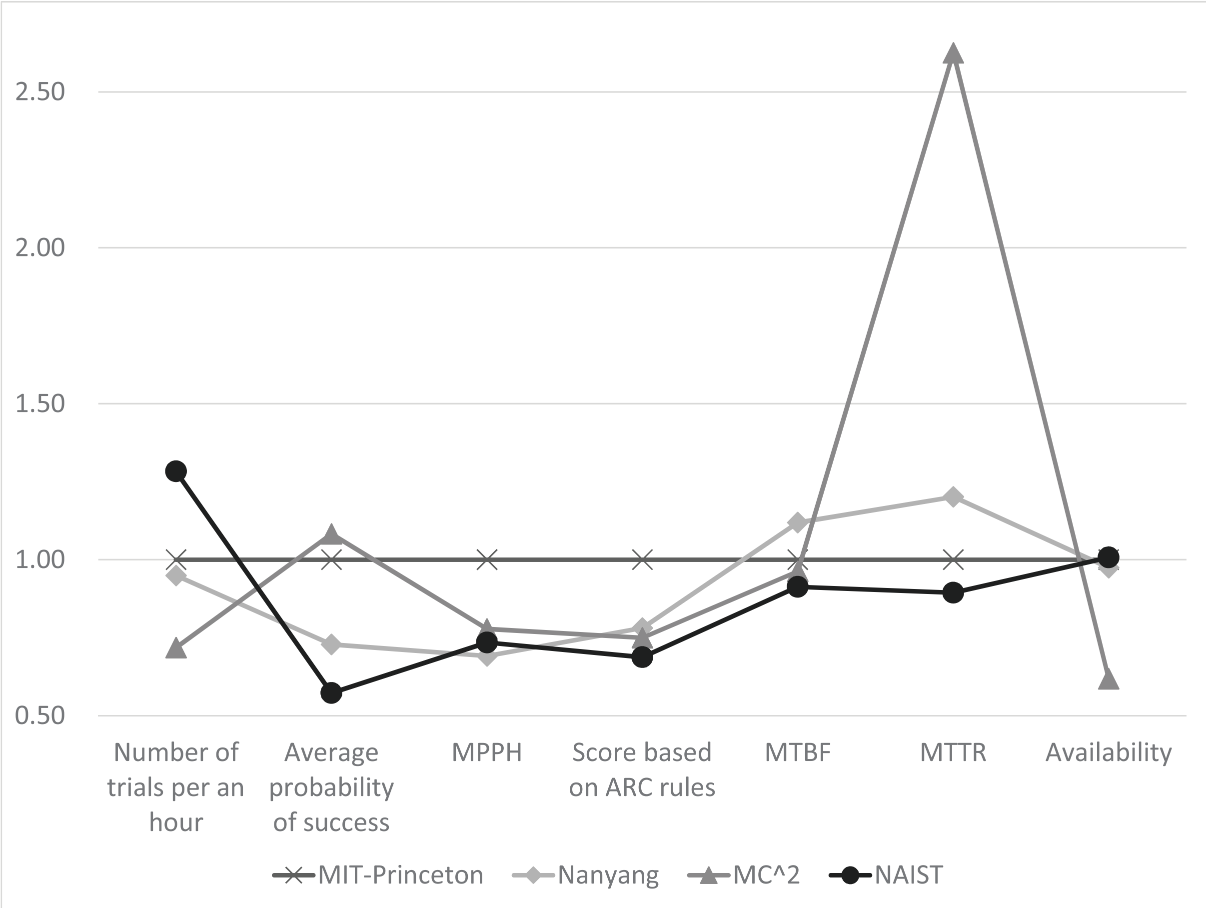

Figure 6. System performance comparison based on the selected metrics.

tries aggressively, thus items flipped when grasping failed, and, consequently, the state of the bin was changed. Moreover, they sometimes changed the grasping point or the target object, so the success probability increases. MC^2 did gently and naive repetition so that the state of the bin did not change. Then, the same unsuccessful picking motion is repeated.

In terms of **Availability**, the MC^2 team's value is very low, while the other teams present high, comparable values. When combined with other indicators, MC^2 has a high probability of success but it takes long time to recover from failure. The other teams seem to be balanced on the speed of operation, the probability of success, and the time of recovery from failure.

6.2 *Lessons learned and future practical system design*

In the previous section, we estimated the design policy differences and their performance with newly introduced metrics, which could not be understood from the analysis based on a single metric.

Next, we discuss an ideal system according to the arguments we have made so far. We found that it is especially advantageous to have short operation time, if we try to estimate the ideal system design. **MPPH**, which is highly correlated with the competition score, is likely to become significantly larger as the operation time is shortened. Although high probability of success is preferable, its effect on **MPPH** is only linear. These facts indicate that in a system hardware configuration that best shortens cycle times, a hand eye system is disadvantageous, so vision sensors should be placed separately. Nevertheless, the freedom of the field of view is restricted with a hand eye configuration. If the system continues to manipulate the same objects, this restriction will not be a problem. However, this can be disadvantageous if the target objects constantly change, which will require a workspace reconfiguration.

In addition, recovery strategies for grasping failures are important and repetition which does not rely on probabilistic phenomena is required, for example, by changing parameters of the recognition algorithm and grasping method, and possibly changing postures of the object by such as flipping them, shaking the bins, and so on. Here too, it is important to improve the repetition rate.

Now, we can discuss a theoretical practical system design considering a combination of the four teams' systems. It is particularly true that short operation time is a target factor. **MPPH**, which is correlated with the score, is likely to take a significantly larger value as the operation time is shortened. **Average probability of success** is not so important, as the effects on **MPPH** are only linear. These facts give suggestions for tactical strategies to adjust the system. If there is a choice between success probability and operation time, one should choose to improve operation time.

In a word on a hardware configuration, hand eye system is disadvantageous, and vision sensors should be put separately. However, one must be careful for the model switching or system cost in actual business use. On the other hand, the recovery strategy at the time of grasping failure is also important. It is worthwhile to focus on the development of error recovery methods. In any case, speed is desired. If such a team comes to the competition, they will also earn bonus points and must be the first prize.

Finally, we discuss the identification of the optimal system configuration and technology for practical use. Even though, an analysis based on the data from the Amazon Robotics Challenge alone is difficult to generalize, what we have found at this time is that attaching or detaching a vision sensor to a robot is about changing the advantage or disadvantage depending on the presence or absence of production model switching.

Then, the need for the next-generation production system is said to be speeding up for production changeover or switching, re-usability of production system bodies, autonomous improvement of production speed and quality, and unmanned operation time by autonomous error recovery and operation learning [2, 3]. Though these factors are not included in the Amazon Robotics Challenge, it is expected that competition rules are formulated and implemented to compare these factors in future competitions.

7. Conclusions

In this paper, we analyzed four robot systems developed for the Amazon Robotics Challenge using a set of performance metrics that clarify the hidden features behind the competition scoring. Based on the competition results, we could show the difference between these systems and which technologies are important for the competition and future practical use according to the proposed metrics. The technologies relevant to picking robots are improved through the competition but further technology improvements are needed for practical use. We expect this analysis to be a good reference for advanced future technologies along with novel needs of the industry.

Though the evaluation criteria reflecting the needs of the industry were reflected in the scoring of competitions, it is also true that high-level systems have been proposed with implicit specifications which are difficult to evaluate through competition scores. In fact, working speed and probability of item damage are important as needs in the industry, and they can become problems in advanced system design and adjustment. However, in this paper, the ultimate solution has not been found because no records of item damage exist. It is presumed that all teams are afraid of penalty points so that they added a wide-enough margin to their system operation speed to prevent items from being damaged. In the future, a system configuration which can reduce such margin will be important, as it will also improve the operating speed.

Acknowledgement(s)

We gratefully thank our colleagues and team members whose support drove the success of our teams.

Team MIT-Princeton: Kuan-Ting Yu, Elliott Donlon, Francois R. Hogan, Maria Bauza, Daolin Ma, Orion Taylor, Melody Liu, Eudald Romo, Nima Fazeli, Ferran Alet, Nikhil Chavan Dafle, Rachel Holladay, Isabella Morona, Prem Qu Nair, Druck Green, Ian Taylor, Weber Liu, and Thomas Funkhouser.

Team Nanyang: Reeve Zheng Hao Chong, Kok Yuan Yik, Ramamoorthy Luxman, Pang Wee Ching, Ren Meixuan, Teoh Yee Seng, Hendra Suratno Tju, Wu Jing, Zhang Dongsheng, and Zhao Yi.

Team MC^2: Ryosuke Kawanishi, Kenta Kato, Koji Shiratsuchi, Rintaro Haraguchi, Ryosuke Araki, Hironobu Fujiyoshi, Shuichi Akizuki, Manabu Hashimoto, Hiroyuki Tsutada, Makito Seki, Haruhisa Okuda, Kiyoshi Maekawa, Masahiro Shikai, Masaki Iizuka, Takuro Torii, Takahiro Hasegawa, Takashi Nanmoto, Katsuhoshi Maekawa, Katsuhisa Yamada, Akira Okihiro, Takeshi Katsura, and Norio Matsui.

Team NAIST-Panasonic: Lotfi El Hafi, Felix von Drigalski, Wataru Yamazaki, Viktor Hoerig, Arnaud Delmotte, Akishige Yuguchi, Marcus Gall, Chika Shiogama, Kenta Toyoshima, Pedro Uriguen, Rodrigo Elizalde, Masaki Yamamoto, Yasunao Okazaki, Kazuo Inoue, Katsuhiko Asai, Ryutaro Futakuchi, Seigo Okada, Yusuke Kato, and Pin-Chu Yang.

References

[1] Marvel JA, Saidi K, Eastman R, Hong T, Cheok G, Messina E. Technology readiness levels for randomized bin picking. In: Proceedings of the Workshop on Performance Metrics for Intelligent Systems. PerMIS '12. College Park, Maryland. New York, NY, USA: ACM. 2012. p. 109–113.

[2] NIST. Agile Robotics for Industrial Automation Competition. Online. 2019. Available from: `https://www.nist.gov/el/intelligent-systems-division-73500/agile-robotics-industrial-automation-competition` [cited 2019-06-21].

[3] Yokokohji Y, Kawai Y, Shibata M, Aiyama Y, Kotosaka S, Uemura W, Noda A, Dobashi H, Sakaguchi T, Yokoi K. Assembly challenge: a robot competition of the Industrial Robotics category, World Robot Summit - summary of the pre-competition in 2018. Advanced Robotics. 2019;33(17):876–899.

[4] Mizuho S, Hiroki D, Wataru U, Shinya K, Yasumichi A, Takeshi S, Yoshihiro K, Akio N, Kazuhito Y, Yasuyoshi Y. Task-board task for assembling a belt drive unit. Advanced Robotics. 2019; 33(17):TADR–2019–0208.

[5] Jolliffe IT. Principal component analysis. 2nd ed.. 2nd ed. New York, NY: Springer. 2002.

[6] Reliability and availability basics. Online. 2019. Available from: `http://www.eventhelix.com/RealtimeMantra/FaultHandling/reliability_availability_basics.htm` [cited 2019-05-22].

[7] Reliability engineering. Online. 2019. Available from: `https://en.wikipedia.org/wiki/Reliability_engineering` [cited 2019-05-22].

[8] Correll N, Bekris KE, Berenson D, Brock O, Causo A, Hauser K, Okada K, Rodriguez A, Romano JM, Wurman PR. Analysis and observations from the first Amazon Picking Challenge. IEEE Transactions on Automation Science and Engineering. 2018;15(1):172–188.

[9] Hernandez C, Bharatheesha M, Ko W, Gaiser H, Tan J, van Deurzen K, de Vries M, Van Mil B, van Egmond J, Burger R, et al.. Team Delft's robot winner of the Amazon Picking Challenge 2016. In: Robot World Cup. Springer. 2016. p. 613–624.

[10] Morrison D, Tow AW, McTaggart M, Smith R, Kelly-Boxall N, Wade-McCue S, Erskine J, Grinover R, Gurman A, Hunn T, et al.. Cartman: The low-cost Cartesian manipulator that won the Amazon Robotics Challenge. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE. 2018. p. 7757–7764.

[11] Garcia Ricardez GA, El Hafi L, von Drigalski F. Standing on giant's shoulders: Newcomer's experience from the Amazon Robotics Challenge 2017. In: Robotic Item Picking - Applications in Warehouse & E-Commerce. Springer. 2019 (in press).

[12] Zeng A, Song S, Yu KT, Donlon E, Hogan FR, 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: 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE. 2018. p. 1–8.

[13] Causo A, Chong ZH, Luxman R, Kok YY, Yi Z, Pang WC, Meixuan R, Teoh YS, Jing W, Tju HS, et al.. A robust robot design for item picking. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE. 2018. p. 7421–7426.

[14] Fujita M, Domae Y, Kawanishi R, Kato K, Shiratsuchi K, Haraguchi R, Araki R, Fujiyoshi H, Akizuki S, Hashimoto M, Garcia Ricardez GA, Causo A, Noda A, , Okuda H, Ogasawara T. Bin-picking robot using a multi-gripper switching strategy based on object sparseness. In: 2019 IEEE International Conference on Automation Science and Engineering. IEEE. 2019. p. 1540–1547.

[15] John A, Eric B, Rodenberg N, Jaeger H, Lipson H. A positive pressure universal gripper based on the jamming of granular material. IEEE Transactions on Robotics. 2012;28(2):341–350.

[16] Domae Y, Okuda H, Taguchi Y, Sumi K, Hirai T. Fast graspability evaluation on single depth maps for bin picking with general grippers. In: 2014 IEEE International Conference on Robotics and Automation (ICRA). IEEE. 2014. p. 1997–2004.

[17] Jiang Y, Moseson S, Saxena A. Efficient grasping from RGBD images: Learning using a new rectangle representation. In: 2011 IEEE International Conference on Robotics and Automation. IEEE. 2011. p. 3304–3311.

[18] Lenz I, Lee H, Saxena A. Deep learning for detecting robotic grasps. International Journal of Robotics Research. 2015;34(4-5):705–724.

[19] Pinto L, Gupta A. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE. 2016. p. 3406–3413.

[20] 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. International Journal of Robotics Research. 2018;37(4-5):421–436.

[21] Rodriguez A, Mason MT, Ferry S. From caging to grasping. International Journal of Robotics Research. 2012;31(7):886–900.

[22] Mahler J, Pokorny FT, Hou B, Roderick M, Laskey M, Aubry M, Kohlhoff K, Kröger T, Kuffner J, Goldberg K. Dex-Net 1.0: A cloud-based network of 3D objects for robust grasp planning using a multi-armed bandit model with correlated rewards. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE. 2016. p. 1957–1964.

[23] Mahler J, Liang J, Niyaz S, Laskey M, Doan R, Liu X, Ojea JA, Goldberg K. Dex-Net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. arXiv preprint arXiv:170309312. 2017;.

[24] 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: 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE. 2018. p. 1–8.

[25] Matsumura R, Harada K, Domae Y, Wan W. Learning based industrial bin-picking trained with approximate physics simulator. In: International Conference on Intelligent Autonomous Systems. Springer. 2018. p. 786–798.

[26] Jonschkowski R, Eppner C, Höfer S, Martín-Martín R, Brock O. Probabilistic multi-class segmentation for the Amazon Picking Challenge. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE. 2016. p. 1–7.

[27] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems. 2015. p. 91–99.

[28] Redmon J, Divvala S, Girshick R, Farhadi A. You Only Look Once: Unified, real-time object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016 June.

[29] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC. SSD: Single shot multibox detector. In: Leibe B, Matas J, Sebe N, Welling M, editors. Computer Vision – ECCV 2016. Cham: Springer International Publishing. 2016. p. 21–37.

[30] Besl PJ, McKay ND. A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1992 February;14(2):239–256.

[31] Liu MY, Tuzel O, Veeraraghavan A, Chellappa R. Fast directional chamfer matching. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE. 2010. p. 1696–1703.

[32] Drost B, Ulrich M, Navab N, Ilic S. Model globally, match locally: Efficient and robust 3D object

recognition. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE. 2010. p. 998–1005.

[33] Choi C, Taguchi Y, Tuzel O, Liu MY, Ramalingam S. Voting-based pose estimation for robotic assembly using a 3D sensor. In: 2012 IEEE International Conference on Robotics and Automation. IEEE. 2012. p. 1724–1731.

[34] Torii T, Hashimoto M. Model-less estimation method for robot grasping parameters using 3D shape primitive approximation. In: 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE). IEEE. 2018. p. 580–585.

[35] Zeng A, Yu KT, Song S, Suo D, Walker E, Rodriguez A, Xiao J. Multi-view self-supervised deep learning for 6D pose estimation in the Amazon Picking Challenge. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE. 2017. p. 1386–1383.

[36] Bian J, Lin W, Matsushita Y, Yeung S, Nguyen T, Cheng M. Gms: Grid-based motion statistics for fast, ultra-robust feature correspondence. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017 July. p. 2828–2837.

[37] Garcia Ricardez GA, von Drigalski F, El Hafi L, Ding M, Takamatsu J, Ogasawara T. Lessons from the Airbus Shopfloor Challenge 2016 and the Amazon Robotics Challenge 2017. In: 18th SICE System Integration Division Annual Conf.. Sendai, Japan. 2017 dec. p. 572–575.

[38] Garcia Ricardez GA, von Drigalski F, El Hafi L, Okada S, Yang PC, Yamazaki W, Hoerig V, Delmotte A, Yuguchi A, Gall M, Shiogama C, Toyoshima K, Uriguen Eljuri PM, Elizalde Zapata R, Ding M, Takamatsu J, Ogasawara T. Warehouse picking automation system with learning- and feature-based object recognition and grasping point estimation. In: 18th SICE System Integration Division Annual Conf.. Sendai, Japan. 2017 dec. p. 2249–2253.

[39] Zeng A, Song S, Lee J, Rodriguez A, Funkhouser T. Tossingbot: Learning to throw arbitrary objects with residual physics. arXiv preprint arXiv:190311239. 2019;.