Sensors and Materials, Vol. 36, No. 4 (2024) 1389–1404
MYU Tokyo

S & M 3606

Hybrid Deep Learning and FAST–BRISK 3D Object Detection Technique for Bin-picking Application

Thanakrit Taweesoontorn, Sarucha Yanyong, and Poom Konghuayrob*

School of Engineering, Department of Robotics and AI Engineering, King Mongkut's Institute of Technology Ladkrabang, No. 1, Chalong-Kung, 1-Alley, Ladkrabang, Bangkok 10520, Thailand

(Received January 9, 2024; accepted February 28, 2024)

Keywords: computer vision, object detection and pose estimation, bin-picking application, robot operating system

In the field of industrial robotics, robotic arms have been significantly integrated, driven by their precise functionality and operational efficiency. We here propose a hybrid method for bin-picking tasks using a collaborative robot, or cobot combining the You Only Look Once version 5 (YOLOv5) convolutional neural network (CNN) model for object detection and pose estimation with traditional feature detection based on the features from accelerated segment test (FAST) technique, feature description using binary robust invariant scalable keypoints (BRISK) algorithms, and matching algorithms. By integrating these algorithms and utilizing a low-cost depth sensor camera for capturing depth and RGB images, the system enhances real-time object detection and pose estimation speed, facilitating accurate object manipulation by the robotic arm. Furthermore, the proposed method is implemented within the robot operating system (ROS) framework to provide a seamless platform for robotic control and integration. We compared our results with those of other methodologies, highlighting the superior object detection accuracy and processing speed of our hybrid approach. This integration of robotic arm, camera, and AI technology contributes to the development of industrial robotics, opening up new possibilities for automating challenging tasks and improving overall operational efficiency.

1. Introduction

In recent years, the robotics field has undergone significant advancements, resulting in transformations across various industries, including manufacturing and logistics. Notably, industrial robots have become prevalent owing to their capability to perform repetitive tasks with high accuracy, efficiency, and reliability. Concurrently, computer vision and artificial intelligence are two technologies that have particularly influenced the field of robotics. Computer vision facilitates the interpretation of visual information, such as images and videos, by machines, empowering them to execute complex tasks. In the same way, AI enables machines to learn from experience and adjust their behaviour to achieve specific goals. The integration of computer vision and AI empowers robots to perceive and interact with their environment or

*Corresponding author: e-mail: poom.ko@kmitl.ac.th
https://doi.org/10.18494/SAM4840

ISSN 0914-4935 © MYU K.K.
https://myukk.org/

workspace. By utilizing cameras and AI algorithms, robots can detect and recognize objects. This makes it possible to perform complex and flexible tasks such as object detection, pick and place, and bin-picking.

Bin-picking is one of the significant applications in robotics and involves retrieving objects from a bin or container. In this application, the sensor or camera that captures the images used to identify and detect the object as well as estimate the pose of the object is very important. Several studies, including the one cited in Ref 1, have made advancements in the field of robot work with cameras. Zhuang *et al.*(1) demonstrated successful implementation in real-world industrial settings. However, it is important to realize that the sensor or camera technology employed in these studies, such as 3D scanners or large-depth sensor cameras, can be associated with high costs. Particularly in industrial bin-picking applications, the use of expensive sensors such as 3D scanners is common. This dependence on costly sensors is attributed to the limitations of alternative options, such as low-cost cameras. Low-cost cameras often face constraints in terms of image resolution and capturing distance, which make them insufficient for handling large workspace tasks. This limitation is highlighted in the comparison described in Ref 2.

Previous studies have explored the integration of low-cost cameras in various methods for bin-picking tasks. Torres *et al.*(3) introduced an application utilizing low-cost cameras for picking up small objects such as car connectors, achieving 4D object detection and pose estimation. However, despite accurate object detection, the algorithm's processing time for recognizing objects and determining their pose/orientation remained lengthy. In contrast, Zhuang *et al.*(4) demonstrated faster algorithmic speed, albeit with lower object detection accuracy. Conversely, Wong *et al.*(5) achieved high object detection accuracy but observed limitations in object pose estimation accuracy. Remarkably, all these studies employed single-view approaches for object detection and pose estimation.

Performing object detection and pose estimation by capturing an image in a single view, such as the top view of the workspace, is challenging because of the camera's distance from the objects and results in low-resolution object images and lower accuracy in pose estimation. To overcome this issue, the eye-in-hand technique was adopted, which involves attaching the camera to the robot's end-effector. This configuration enables flexible camera movement by manipulating the robot, facilitating image capture from close view. Monica *et al.*(6) clearly showed the advantage of the eye-in-hand, which allows high depth measurements from the sensor, but the detection time is too long.

The objective of this research is to enhance the speed of object detection and pose estimation within bin-picking applications while maintaining high accuracy. This is achieved through the utilization of the Dobot CR5 collaborative robot in conjunction with a low-cost depth sensor camera, specifically the Intel Realsense D435i camera. Employing the eye-in-hand technique, as shown in Fig. 1, enables the acquisition of both a comprehensive top view of the entire workspace and close-up views for specific objects, thereby enhancing object resolution. To achieve the goal of object detection and pose estimation, we utilized the following two detection approaches.

- Firstly, You Only Look Once Version 5 (YOLOv5), a state-of-the-art deep learning algorithm for object detection, was employed. By leveraging YOLOv5, objects could be detected in real-time streams, and the robot could be precisely moved closer to the targeted object.

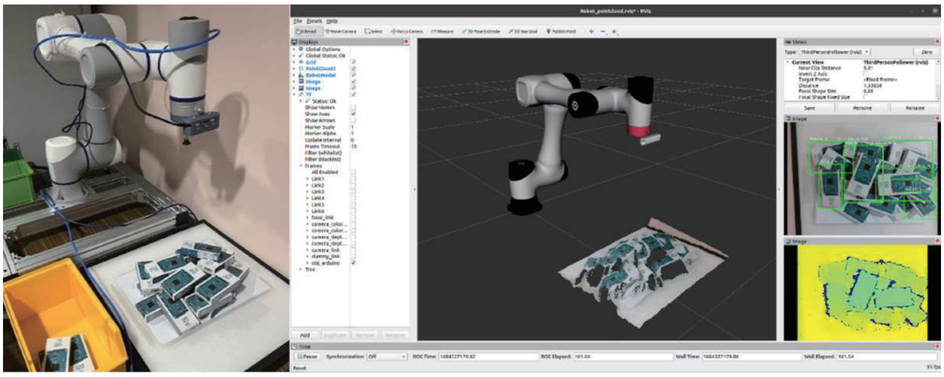


Fig. 1. (Color online) Actual robot and real-time synchronized robot simulation in Rviz with pointcloud, depth image, and detected object image.

- Secondly, the features from accelerated segment test (FAST) and binary robust invariant scalable keypoints (BRISK) algorithms, which are feature detection, description, and matching algorithms, were utilized. As the robot approached close to the object, a high-resolution object image was captured by the camera. This image was suitable for using the FAST and BRISK algorithms to rapidly estimate the object's pose. The combination of these techniques enabled high accuracy in picking up objects at the correct position and orientation.

By integrating these methods, as shown in Fig. 2, our technology enhances bin-picking applications by optimizing object detection and pose estimation processes. This contributes to the overall automation capabilities of industrial robots, offering a cost-effective solution for efficient and accurate object manipulation.

2. Related Theory

2.1 Traditional feature detection, description, and matching

In the field of computer vision, significant research has been dedicated to traditional feature detection, description, and matching algorithms. The widely adopted Harris corner detector(7) identifies corners by analyzing local intensity gradients. Scale invariant feature transform (SIFT)(8) and speeded up robust features (SURF)(9) algorithms excel in detecting scale-invariant keypoints and generating descriptors resilient to variations in scale, rotation, and viewpoint. Similarly, FAST(10) and BRISK(11) algorithms are designed for efficient feature detection. FAST is a corner detection method that can be used to extract feature points.(12) While these traditional methods have proven effective, they rely on hand-crafted features and may potentially overlook crucial information in an image. Consequently, they face limitations when it comes to recognizing objects and their poses under diverse conditions, since they require the object's image resolution to align closely with the reference image to accomplish accurate detection and recognition. In contrast, deep learning approaches have emerged as powerful tools that learn features directly from data, enabling more robust object detection, even in the presence of scale, rotation, and viewpoint variations.

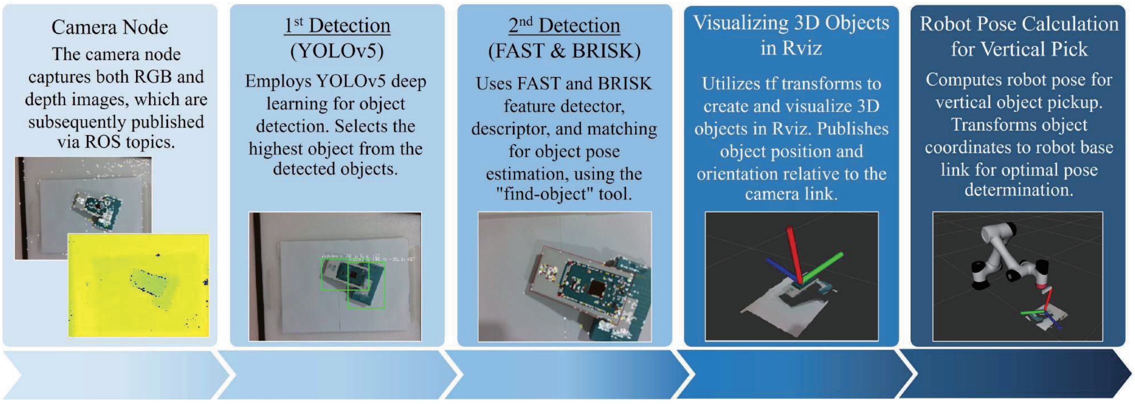


Fig. 2. (Color online) Schematic representation of the system workflow, from the camera to the detection nodes and ultimately to the robot node, highlighting the various stages involved in object detection, pose estimation, and robot pose computation.

2.2 Deep learning and convolutional neural network (CNN)

Deep learning has revolutionized various computer vision tasks, including object detection and recognition. CNNs have emerged as powerful models for analyzing image data, as they can automatically learn hierarchical representations from raw pixels.[(13)] CNNs have demonstrated their effectiveness in object detection tasks, providing accurate bounding box predictions and advancing applications in robotics, surveillance, and autonomous vehicles. One popular CNN-based approach for object detection is YOLO.[(14)] YOLO introduced a unified architecture that predicts object bounding boxes and class probabilities in a single pass, enabling real-time object detection.

2.3 Robot operating system (ROS) framework and visualization tool, Rviz

ROS has made significant contributions to the field of robotics by providing a versatile framework that facilitates seamless communication and collaboration among various components. In a foundational paper,[(15)] the fundamental principles and architectural design of the ROS were discussed, and it was established as an open-source operating system. This influential work has played a pivotal role in the widespread adoption and advancement of ROS within the robotics community. Researchers leverage the capabilities of ROS to effortlessly integrate cameras with computers and robots, enabling innovative research and development in the field of robotics. Within the ROS framework, ROS visualization, or Rviz, stands out as an exceptional 3D visualization tool. Rviz offers real-time visualization capabilities for various data types, such as robot models, images, pointcloud, and trajectories. Kam *et al.*[(16)] extensively explored the practical application of Rviz in constructing mobile robots. Their paper showcases Rviz's crucial role in the visualizing robot. In this study, we emphasize Rviz's significant contribution to the real-time visualization and evaluation functions of robotic systems, which further advances the capabilities of the ROS framework.

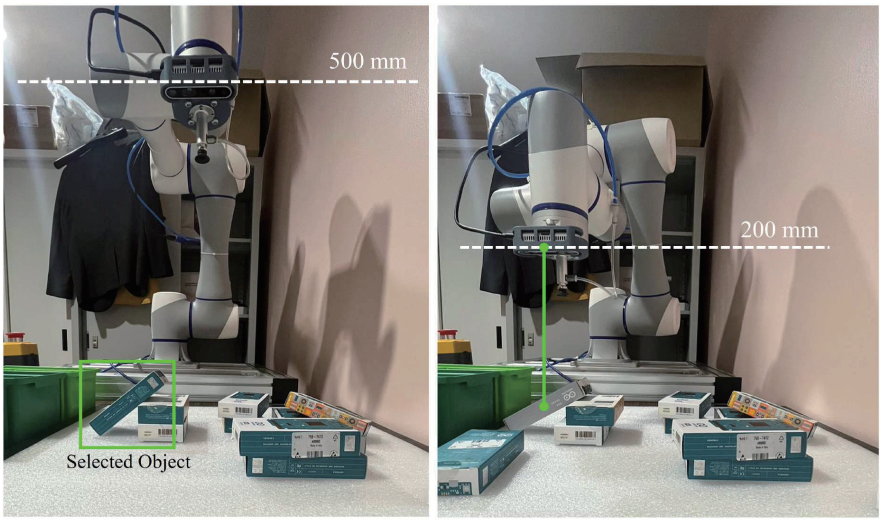
3. Object Detection and Pose Estimation

To detect objects and estimate their poses, the system employs two detection techniques, as shown in Fig. 2. Firstly, the robot positions itself in a top-view configuration, enabling the camera to capture the entire workspace for comprehensive coverage. This top-view image is processed by the first detection node, which utilizes a trained YOLOv5 model for object detection. By comparing the positions of all detected objects, the algorithm selects the object with the highest vertical position. Subsequently, the robot moves closer to the selected object while maintaining a fixed distance of 200 mm above the object, as shown in Fig. 3. This proximity allows for capturing detailed information. A close-up image of the object is then forwarded to the second detection node. This node employs algorithms to detect the object and estimate its pose, including its x, y, z coordinates and x, y, z orientation in 6D space.

3.1 First detection (YOLOv5)

The first technique (YOLOv5) is a state-of-the-art object detection framework based on deep learning CNNs. YOLOv5 (Fig. 4) is widely recognized for its high accuracy and real-time performance. By simultaneously predicting bounding boxes and class probabilities for multiple objects within an input image, YOLOv5 achieves efficient and precise object detection.

To train the YOLOv5 model, a dataset was prepared using Roboflow. The dataset was carefully labelled, and various augmentation techniques were applied to enhance its diversity and quality. During the training process, the YOLOv5l model was utilized with 100 epochs, batch size of 16, and the training dataset consisting of 1431 images. The example training dataset with augmentation techniques and training output results are shown in Fig. 5.



(a) (b)

Fig. 3. (Color online) (a) Robot's top-view position and (b) robot's position as it moves closer to the selected object.

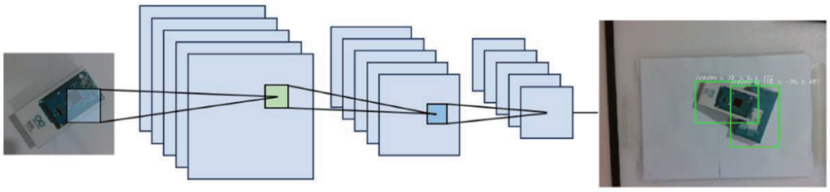
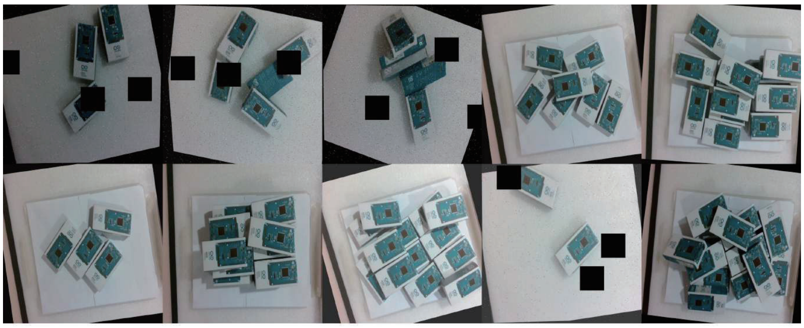
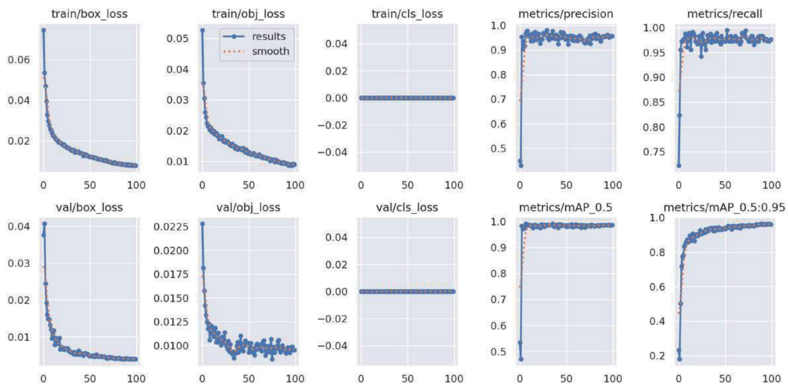


Fig. 4. (Color online) Schematic representation of a simple concept of CNNs.



(a)



(b)

Fig. 5. (Color online) (a) Training dataset with augmentation techniques. (b) Training output results using the YOLOv5l algorithm.

Table 1 presents the object detection training results for the YOLOv5l algorithm, showcasing high accuracy metrics including precision, recall, mAP50, and mAP50-95 indices. These evaluation metrics reflect the system's capability to detect all objects within the workspace when the robot is positioned in the top view. In scenarios where multiple objects are detected, the system selects the object with the highest vertical position based on depth information from the camera node. Subsequently, the robot is directed to move above the selected object, maintaining a fixed distance. Subsequently, the second detection technique is employed to estimate the pose using a close-up image.

Table 1
Object detection training result output.

Algorithm	Precision	Recall	mAP50	mAP50-95
YOLOv5l	0.955	0.983	0.986	0.965

3.2 Second detection (FAST and BRISK)

In addition to YOLOv5, the second technique employed in this research is a combination of FAST as the feature detector and BRISK as the feature descriptor.

3.2.1 Feature detector—FAST

Feature detection is a fundamental computer vision technique that identifies distinctive points within an image for further analysis. FAST, a highly efficient algorithm designed to detect corner points, was employed. FAST compares the intensities of pixels in a circular neighborhood surrounding a candidate point, evaluating whether enough contiguous pixels are either brighter or darker than the candidate pixel.

The algorithm utilizes a circle consisting of 16 pixels, labelled clockwise from 1 to 16, within which a candidate point is assessed, as shown in Fig. 6. By comparing the intensities of the candidate pixel (I) with a threshold value (t), FAST applies the following conditions for corner classification:

- Condition 1: If a set of (N) contiguous pixels (S) within the circle satisfies $\forall x \in S : I_x > I_p + t$, the candidate point is classified as a corner.
- Condition 2: If a set of (N) contiguous pixels (S) within the circle satisfies $\forall x \in S : I_x < I_p - t$, the candidate point is classified as a corner.

The selection of appropriate values for N and the threshold (t) involves consideration of the number of detected corner points and computational efficiency; typically, N is set to 12. Additionally, a high-speed test can be applied to exclude noncorner points, further improving the algorithm's overall performance. FAST's rapid intensity comparison process enables real-time corner keypoint detection with high accuracy and efficiency.

3.2.2 Feature descriptor—BRISK

Once keypoints are detected, it is essential to describe their visual attributes in a compact and distinctive technique. This is where feature descriptors come into play. In this research, the BRISK algorithm is utilized as the feature descriptor. BRISK captures the unique characteristics of keypoints by encoding their local appearance, texture, and shape information into a compact binary representation. This binary descriptor ensures robustness against image transformations and enables the efficient matching of keypoints, enhancing the accuracy and reliability of the object pose estimation process.

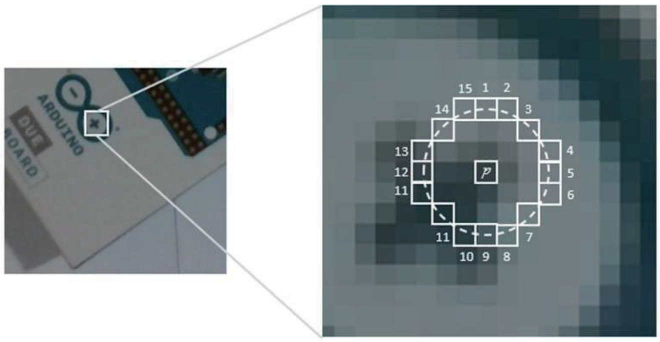


Fig. 6. (Color online) Schematic of the keypoint being evaluated, along with the surrounding 16 pixels arranged in a circular pattern.

3.2.3 Feature matching

Feature matching techniques are used to establish correspondence between keypoints in the reference and query images. The brute force strategy is applied for nearest neighbour matching, wherein the feature descriptors are compared to find suitable matches. Homography estimation techniques are then employed to determine the geometric transformation between the images. This estimation accurately calculates the object's pose, including its position and orientation.

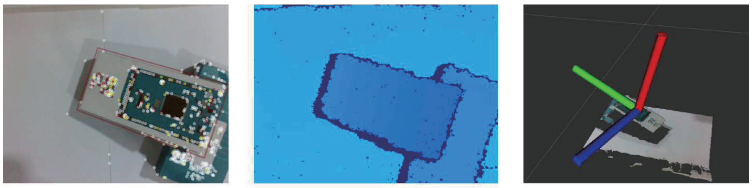
By integrating FAST and BRISK, the system achieves robust and accurate object pose estimation. Leveraging the "find-object" tool,(17) which incorporates feature detection algorithms such as SIFT, FAST, ORB, and feature description, results in effective object identification in images and real-time streams. The "find-object" tool utilizes advanced techniques such as RANSAC and homography functions to accurately estimate the pose of objects, even in challenging scenarios involving perspective transformations.

By sequentially employing these techniques, the system excels in object detection and pose estimation. The initial top-view detection provides comprehensive workspace coverage, while the subsequent close-up detection offers detailed information for precise pose estimation. This seamless integration within a ROS framework ensures efficient object detection and pose estimation, as depicted by the 6D object link shown in Fig. 7, serving as a reference for the object's position and orientation required for picking up.

3.3 Robot pose calculation and determination

To enable precise vertical object manipulation, the robot's picking-up pose was calculated with a fixed offset from the target object. The pose computation involved matrix-based calculations, referencing the robot frame for accurate positioning within the ROS framework. By aligning itself with the object in a controlled and efficient manner, the robot could enhance the overall performance and reliability of the bin-picking system.

Within the ROS framework, the object link was established to reference the camera link integrated into the robot's URDF file. This connection allowed for continuous derivation of the



(a) (b) (c)

Fig. 7. (Color online) (a) RGB image with feature detection from FAST. (b) Depth image represented as a heat map. (c) Object link visualized in Rviz.

object's position from the robot's base link, enabling accurate visualization of both the robot and the object within the Rviz (Fig. 8). This integration provided a comprehensive and dynamic representation of the robot-object configuration for research purposes.

The picking pose, denoted as $Robot_{xyz}$, was determined using matrix transformations based on the object's position Obj_x, Obj_y, Obj_z and orientation Obj_{rx}, Obj_{ry}, Obj_{rz}. The calculations for the robot's picking-up position on each axis $Robot_x$, $Robot_y$, $Robot_z$ were computed using the following equations.

$$Robot_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(R_y) & -\sin(R_y) \\ 0 & \sin(R_y) & \cos(R_y) \end{bmatrix} \tag{1}$$

$$Robot_y = \begin{bmatrix} \cos(R_x) & 0 & \sin(R_x) \\ 0 & 1 & 0 \\ -\sin(R_x) & 0 & \cos(R_x) \end{bmatrix} \tag{2}$$

$$Robot_z = \begin{bmatrix} \cos(R_z) & -\sin(R_z) & 0 \\ \sin(R_z) & \cos(R_z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

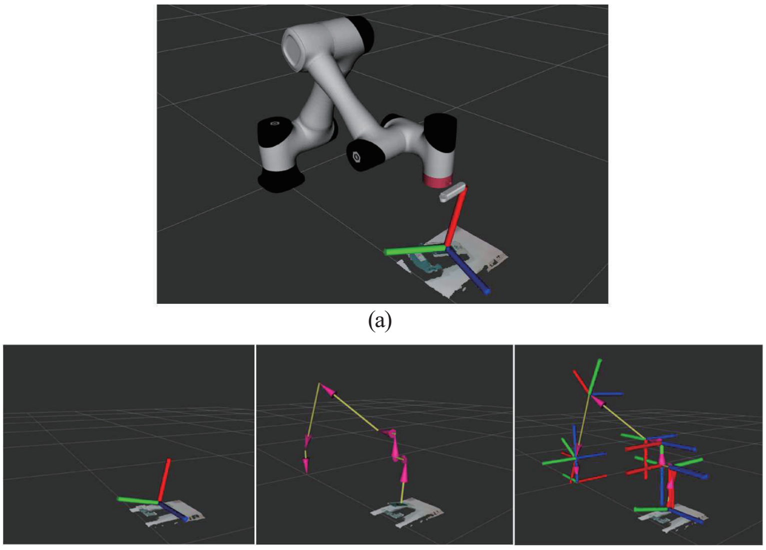
Equations (1)–(3) represent the calculations for the robot's picking-up position on the respective axes. The offset in the picking-up position is denoted as $Robot_{off}$ and can be represented as

$$Robot_{off} = \begin{bmatrix} 0 & 0 & -d \end{bmatrix}. \tag{4}$$

The final calculation of the robot's picking position denoted as $Robot_{xyz}$ is given by

$$Robot_{xyz} = Robot_x \cdot Robot_y \cdot Robot_z \cdot Robot_{off} + \begin{bmatrix} Obj_x & Obj_y & Obj_z \end{bmatrix}. \tag{5}$$

In Eq. (5), the position $Robot_{xyz}$ is added to the object position [Obj_x Obj_y Obj_z] to determine the picking-up position with offset. As a result, the robot stays at a distance of d mm from the object



(a) (b) (c) (d)

Fig. 8. (Color online) Robot pose visualization in Rviz (a). (b)–(d) Continuous derivation of the object's position from the robot's base link.

before moving directly towards it in a vertical manner within the object’s plane. Additionally, the orientation of the robot’s end-effector can be calculated as below.

$$Robot_{rx} = Obj_{rx} + 180 \tag{6}$$

$$Robot_{ry} = -\left(Obj_{ry} - 180\right) \tag{7}$$

$$Robot_{rz} = Obj_{rz} + 180 \tag{8}$$

Equations (6)–(8) provide the calculations for determining the values of $Robot_{rx}$, $Robot_{ry}$, and $Robot_{rz}$, which represent the orientation of the robot’s end-effector. By incorporating these calculations, the robot can achieve the desired picking-up pose. This approach ensures that the robot is properly positioned and oriented for efficient and controlled object manipulation, as shown in Fig. 9.

In summary, our proposed hybrid method demonstrates remarkable capabilities in terms of object detection and pose estimation. As the comparison in Table 2, the hybrid method approach surpasses traditional methods in detecting objects across all distances and orientations. Moreover, it provides accurate 6D pose estimation, along with the precise specification of x, y, and z coordinates.

4. Experiments and Results

In the experiment, the implemented algorithm successfully detected the target object, and the object link was displayed in Rviz, along with the robot model, camera link, point cloud, depth image, and detected object image, as shown in Fig. 10.

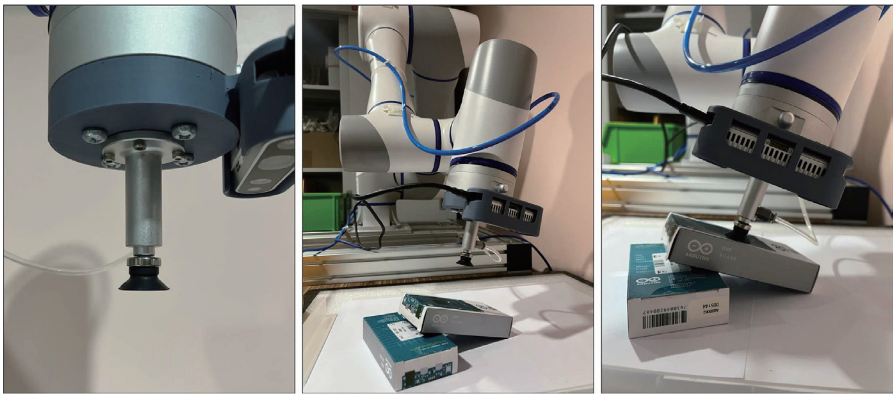


Fig. 9. (Color online) Photographs of the picking-up step, showing precise vertical pick-up direction and perfect articulation of the suction cup edge with the object.

Table 2
Detection capability.

	Our Hybrid Method	YOLOv5	FAST & BRISK	Torres *et al.*[3]	Wong *et al.*[5]
Detectable at any distance in workspace	✓	✓		✓	✓
Detectable at any orientation	✓	✓		✓	✓
6D pose estimation	✓		✓		✓
x and y coordinate specified	✓	✓	✓	✓	✓
z coordinate specified	✓		✓	✓	✓

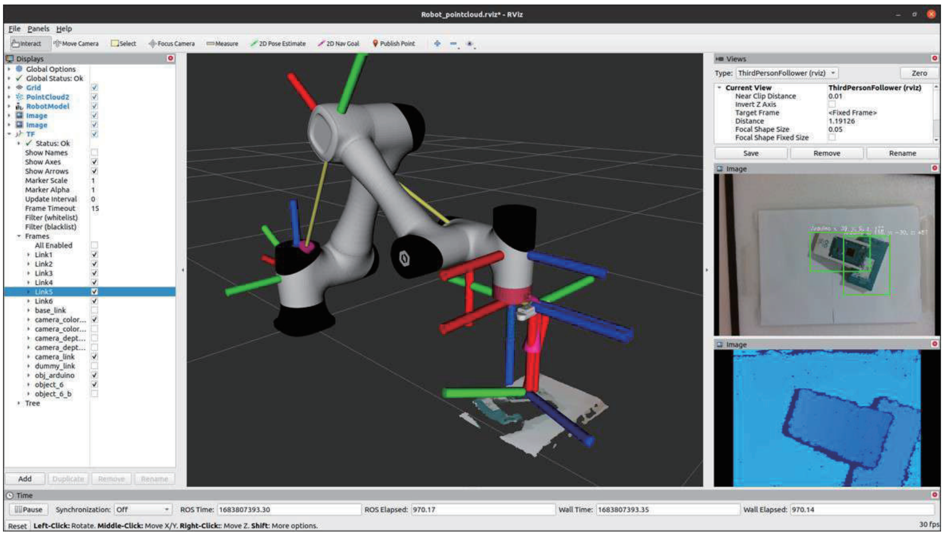


Fig. 10. (Color online) Rviz, the ROS visualization tool, displays the object link, robot model with the camera link, point cloud, depth image, and detected object image.

The performance of the bin-picking system was evaluated through an experiment that encompassed object detection, pose estimation, robot positioning for vertical pick-up, and real-time visualization of robot synchronization. The bin-picking task was divided into three subtasks, as shown in Fig. 11, to assess different scenarios: well-structured object positions, semi-randomized objects with slight rotation, and randomized object positions and orientations.

(1) In the first subtask, objects were neatly arranged in a well-structured position within the container with the aim of testing the system's accuracy in detecting and estimating object poses.
(2) The second subtask introduced semi-randomized objects with slight rotation, challenging the system to handle variations in object orientation.
(3) The third subtask involved randomly positioned and oriented objects, representing a more challenging scenario.

The hybrid technique considerably extends the object detection range, leveraging YOLOv5 capabilities to detect objects at a greater distance, resulting in a larger detectable area or workspace. This expanded detection area is evident in Table 3 and emphasizes the hybrid method's superiority over traditional techniques. Notably, our hybrid method achieves a detectable range that extends to the maximum reach of the robot arm.

The system successfully detected the object by the hybrid method, and the Rviz effectively depicted the actual robot and object link, showcasing the real-world implementation of the system (Fig. 12). This visualization enabled the precise tracking of the object's position and orientation, ensuring accurate manipulation and interaction. The seamless integration of the robot and Rviz facilitated smooth monitoring and control of the object detection and pose estimation process, confirming the system's effectiveness and reliability.

In terms of experimental outcomes, as visually depicted in each process showcased in Fig. 13, the system achieved a 100% single-attempt success rate in the first subtask, while maintaining a

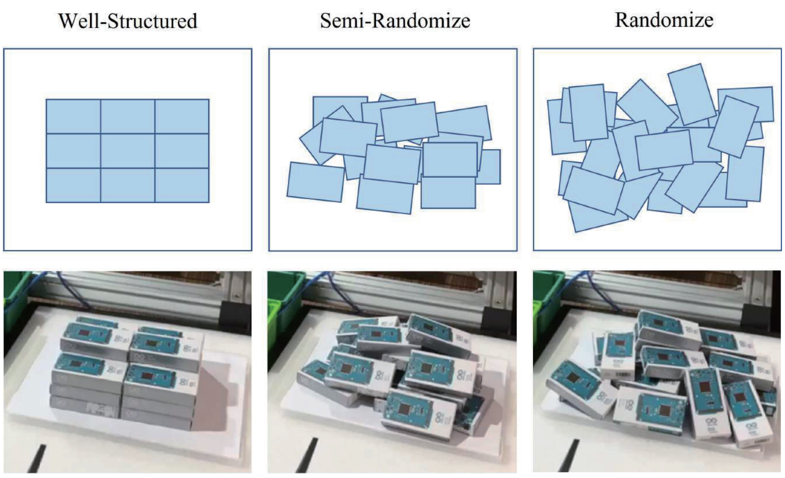


Fig. 11. (Color online) Bin-picking subtasks: well-structured, semi-randomized, and randomized arrangements.

Table 3
Detectable range for each technique.

Technique	Detectable Range (mm)
FAST & BRISK	150–280
YOLOv5 + FAST & BRISK	150–730

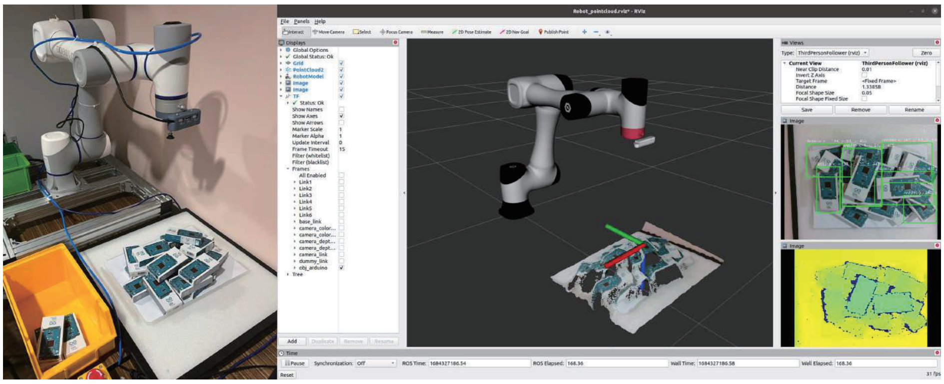


Fig. 12. (Color online) Actual robot and Rviz with target object link display, showcasing the system's real-world implementation and visualization.

Table 4
Our proposed hybrid method operation time.

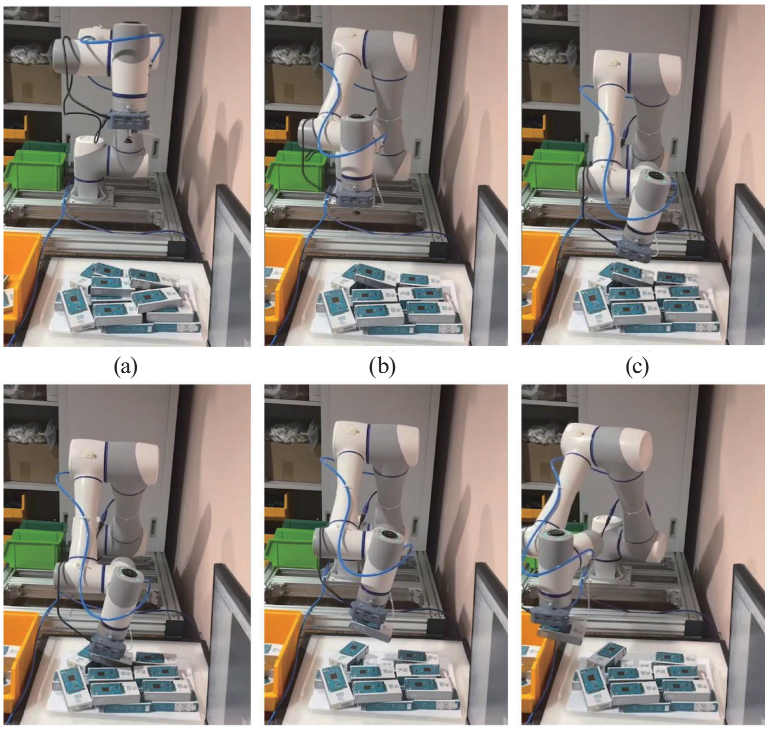
	First Subtask average time (s)	Second Subtask average time (s)	Third Subtask average time (s)
Object detection, positioning and selecting	0.12	0.12	0.12
Robot moving above object	5.30	5.84	5.79
6D Pose / orientation estimation	0.20	0.20	0.20
Picking pose calculation and operation	11.24	12.60	16.84
Total	16.86	18.76	22.95

Table 5
Comparison between the previous methods.

Algorithms	Experimental object	Sensor	Object detection		Pose estimation (5°)	Detection and pose estimation time (s)
			AP_{50}	$AP_{50:95}$		
Torres *et al.*[3]	Connector part	Low-cost	98%	—	—	7.4
Zhuang *et al.*[4]	Tissue Box	Low-cost	92.7%	—	95%	—
Wong *et al.*[5]	Box	Low-cost	98.9%		87.9%	—
Le and Lin[18]	USB Packs	Low-cost	100%	91.2%	—	0.862
Our Hybrid	Arduino Box	Low-cost	98.6%	96.5%	94.7%	0.32

97.5% success rate in the second subtask. However, the success rate decreased slightly to 86.67% in the last subtask. Table 4 provides a breakdown of the operation times for each subtask, including object detection, robot movement, pose estimation, picking pose calculation, and overall operation time.

In the experimental result and comparison, our hybrid method showed impressive performance, as summarized in Table 5. In terms of object detection, our approach achieved an $AP_{50:95}$ score of 96.5%, underscoring its capabilities in accurately detecting objects at any orientation, and a pose estimation accuracy of 94.7% within angle error of 5°. Additionally, our hybrid method demonstrated efficient processing, with a minimal detection and pose estimation time of merely 0.32 s, indicating its suitability for real-time applications. The inclusion of real-



(a) (b) (c) (d) (e) (f)

Fig. 13. (Color online) The sequence of pictures demonstrates the successful execution of the picking-up process. (a) The robot is positioned in the top-view configuration. (b) The robot moves for a close-up view of the target object. (c) The robot approaches the object with a fixed offset. (d) The robot initiates the picking-up motion. (e) The robot successfully picks up the object, (f) The robot transports the object to the storage crate. This series of images showcases the systematic and precise execution of the picking-up task by the robot.

time 3D visualization depicting the robot's synchronization with objects enhances the clarity and convenience of monitoring and controlling the system during operation. This result showing the significant contributions of our work in object detection and pose estimation in industrial robotics paves the way for enhanced automation across real-world applications.

5. Conclusion

We presented a comprehensive system for object detection, pose estimation with a low-cost depth camera, and robot positioning. The system was implemented using the eye-in-hand configuration in bin-picking tasks within the ROS framework. Through the integration of the YOLOv5, FAST, and BRISK algorithms, our system achieves accurate and efficient detection and pose estimation, even in challenging scenarios. Furthermore, our system offers real-time 3D visualization of robot movements, providing insights into its operations. The experimental results confirmed a remarkable improvement in both detection accuracy and processing speed over other methods. This advancement is significant in automation, particularly in the manufacturing and logistics sectors. By utilizing a low-cost depth camera, we not only reduce expenses but also make our system more practical for real-world use, ultimately boosting productivity and efficiency in industrial operations.

Acknowledgments

This research work is supported by the Department of Robotics and AI Engineering, School of Engineering, at King Mongkut's Institute of Technology Ladkrabang. The authors extend their appreciation to the members of the AIMs lab for their invaluable assistance and support in various aspects of this work.

References

1 C. H. Zhuang, S. H. Li, and H. Ding: Rob. Comput. Integr. Manuf. **82** (2023) 102541. https://doi.org/10.1016/j.rcim.2023.102541
2 P. Dolezel, J. Pidanic, T. Zalabsky, and M. Dvorak: Proc. 2019 20th Int. Carpathian Control Conf. (ICCC, 2019) 1−6.
3 P. Torres, J. Arents, H. Marques, and P. Marques: Electronics **11** (2022) 476. https://doi.org/10.3390/electronics11030476
4 C. H. Zhuang, Z. H. Wang, H. Zhao, and H. Ding: Rob. Comput. Integr. Manuf. **68** (2021) 102086. https://doi.org/10.1016/j.rcim.2020.102086
5 C. H. Wong, C. H. Tsai, R. Chen, S. H. Chien, Y. Yang, S. H. Wong, and C. H. Yeh: IEEE Access **10** (2022) 65257. https://doi.org/10.1109/ACCESS.2022.3182114
6 R. Monica, A. Saccuti, J. Aleotti, and M. Lippi: Proc. 2022 IEEE 27th Int. Conf. Emerging Technologies and Factory Automation (ETFA, 2022) 1−8.
7 C. G. Harris and M. J. Stephens: In Alvey Vision Conf. (1988). https://api.semanticscholar.org/CorpusID:1694378
8 D. G. Lowe: Int. J. Comput. Vision **60** (2004) 91. https://doi.org/10.1023/B:VISI.0000029664.99615.94
9 H. Bay, T. Tuytelaars, and L. G. Van: Proc. 2006 European Conf. Computer Vision (ECCV, 2006) 404−417.
10 Features from Accelerated Segment Test (FAST): https://api.semanticscholar.org/CorpusID:17031649 (accessed 2011).
11 S. Leutenegger, M. Chli, and R. Y. Siegwart: 2011 Int. Conf. Computer Vision (ICCV, 2011) 2548−2555.
12 E. Rosten and T. Drummond: Proc. 9th European Conf. Computer Vision (ECCV, 2006) 430−443.
13 A. Krizhevsky, I. Sutskever, and G. E. Hinton: Commun. ACM **60** (2012) 84. https://doi.org/10.1145/3065386
14 J. Redmon, S. Divvala, R. Girshick, and A. Farhadi: Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition (CVPR, 2016) 779−788
15 M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng: Proc. 2009 IEEE Int. Conf. Robotics and Automation (ICRA, 2009) https://api.semanticscholar.org/CorpusID:6324125
16 H. R. Kam, S. H. Lee, T. Park, and C. H. Kim: Telecommun. Syst. **60** (2015) 337. https://doi.org/10.1007/s11235-015-0034-5
17 Find-Object: http://introlab.github.io/find-object (accessed 2011).
18 T. Le and C. Lin: Sensor **19** (2019) 3602. https://doi.org/10.3390/s19163602

About the Authors

Thanakrit Taweesoontorn received his B.S. degree from King Mongkut's Institute of Technology Ladkrabang, Thailand (KMITL), in 2021. From 2021 to 2023, he was a teacher's assistant at KMITL, Thailand. Since 2023, he has been a remote service engineer at E80 Group. His research interests are in industrial robots, machine vision, and artificial intelligence.
(romthanakrit@gmail.com)

Sarucha Yanyong received his B.S. degree in 2012 and his M.S. and Ph.D. degrees in 2014 and 2023, respectively, from King Mongkut's Institute of Technology Ladkrabang (KMITL), Thailand. Since 2023, he has been a professor at KMITL. His research interests are in mobile robotics, embedded systems, and artificial intelligence. (sarucha.ya@kmitl.ac.th)

Poom Konghuayrob received his B.S. degree in 2012 and his M.S. and Ph.D. degrees in 2013 and 2017, respectively, from King Mongkut's Institute of Technology Ladkrabang (KMITL), Thailand. Since 2017, he has been a professor at KMITL. His research interests are in industrial robots, automation systems, manufacturing, adaptive control systems, and artificial intelligence. (poom.ko@kmitl.ac.th)