International Journal of Robotics and Automation, Vol. 40, No. 10, 2025
An Open Access Paper

QUICK-PICK CNN: A NOVEL ALGORITHM FOR QUICKER DUAL-ARM GRASP LOCALISATION IN A CLUTTERED ENVIRONMENT

A. Josin Hippolitus* and R. Senthilnathan*

Abstract

In a progressing and complex world, robot grasping and manipulation in a cluttered environment is a challenging activity. Especially when the object to be manipulated is of unknown geometry and located in a cluttered environment. In this work, a novel quick-pick CNN(QP-CNN) algorithm is implemented to identify the best grasp for a 3D object in real time. The potential impact of this research can range from improving the speed, efficiency, and accuracy in object manipulation of unknown objects in a cluttered environment assuming a model-free context. RGB-D data from the real world about the object to be manipulated is acquired and mapped to the objects. This information acts as the input for the pre-trained networks to provide input to a 7-DOF ABB YuMi dual-arm robot. The objectwise grasping accuracy of QP-CNN is 98.1% and grasp time is 2 s with 100% reliability.

Key Words

Dual-arm grasping, robot control, CNN, deep learning

1. Introduction

The advances made in robotic manipulation through the years are staggering and yet there is a void in how the robot interacts with unknown objects in a cluttered environment. By achieving a solution for this problem there would be an elevated usage of robots in different fields relating to exploratory robotics and agricultural robotics. This requires the robot to decide how and where to grasp the object. In an industrial setup which is a structured environment due to a systematic pattern of working, the robot is well aware of the target object which eliminates the need for an intelligent decision for grasp, however, in an unstructured environment where there is no prior knowledge about the environment as well as geometry of the target object. Hence, it becomes difficult for the robot to grasp without a certain level of intelligence. In these situations, the robot must identify the best-grasping points to pick the target object successfully and efficiently. For the grasp to be considered a success, a major factor that acts as an indicator providing inferences is the grasping points. One of the major strengths of deep learning is its ability to process and learn useful features from the data without explicit feature engineering.

The primary objective of this research is to generate a deep-learning-based grasp locations for a dual-arm robot and execute grasp in real time using an ABB YuMi dual-arm robot. This includes the following steps:

- Obtain custom images and provide them as compatible input to the network.
- Acquire the inference of custom inputs through the algorithm.
- Convert the obtained results into robot-recognizable units and execute the grasp.

Depth image is rendered from the overhead mounted Kinect sensor. The depth image has been used by the quick-pick CNN (QP-CNN) to get information about which region the object to be grasped is located (R or L) and its key configurations like grasp quality (Q), grasp width (W), and grasp angle (Φ). This information is then sent to the controller which executes the grasp. The overall framework for this process is shown in Fig. 1.

1.1 Related Works

Robot grasping starts with grasp synthesis, which can be analytical or empirical [4], [11]. Analytical methods rely on geometrical features of the robot and mathematical calculations to derive a good grasp. They also consider the kinematics and dynamics to result in a stable and successful grasp [5], [12]. Whereas empirical methods solely rely on model and experience-based grasp synthesis [6]–[8].

* Department of Mechatronics Engineering, SRM Institute of Science and Technology Kattankulathur, Chennai, India; e-mail: josinhippolitus@gmail.com; senthilr4@srmist.edu.in
Corresponding author: A. Josin Hippolitus

Recommended by Hani Hagras
(DOI: 10.2316/J.2025.206-1151)

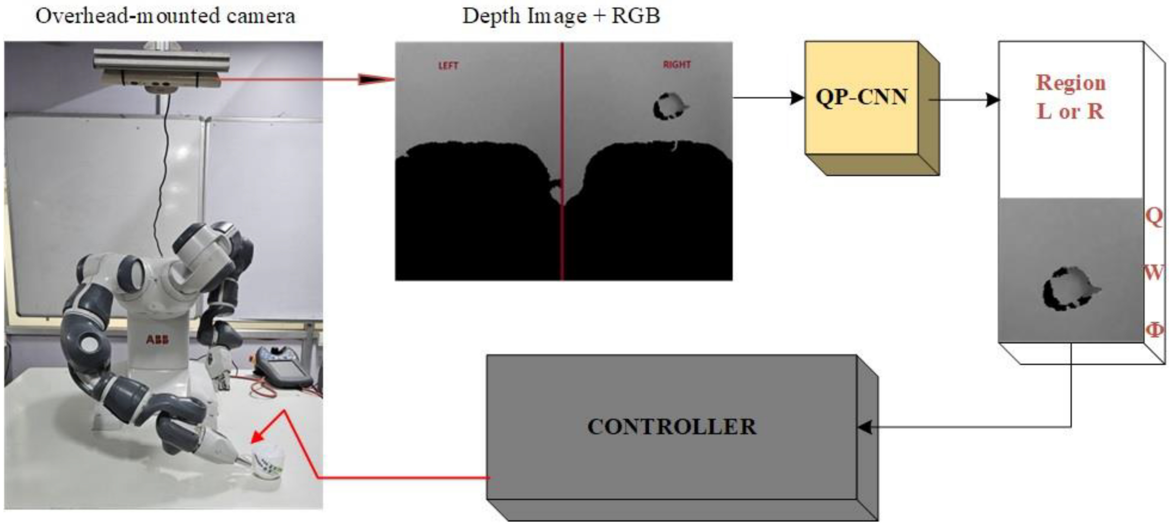


Figure 1. Our novel QP-CNN grasping pipeline.

Analytical methods are successful when the geometry of the object is known, but suffer when the geometry of the object is not known. Empirical methods work well with trained objects but suffer when the objects are new and novel. The general pipeline for robot grasping is rendering a depth image using a depth sensor, identifying potential grasps, ranking those grasps, and choosing the best grasp to execute the grasp.

There are two major parameters by which the quality of the grasp is evaluated: accuracy and time taken for execution of grasp. When there are many methods used to detect the accuracies of the generated grasps, the execution time faced a major roadblock due to inbound dexterity limitations of multiple degrees of freedom arm.

Hippolitus *et al.* [1] have discussed the importance of estimating a grasp pose for a successful grasp using exteroceptive sensors. An overmount Kinect sensor is used as a depth-sensing device to get information about the objects in the workspace. Antipodal grasp-configurations are predicted using an autoencoder network, which extracts very compact representations for a given object [2]. This is highly helpful for simultaneous object recognition and grasping.

DL algorithms, especially convolutional neural networks (CNNs) [16] have shown high advances in computer vision tasks. They are less time-consuming, show improved performance, and use skip connections which leaves out some of the layers in the neural network and ensures reusability but can be computationally expensive and require high GPUs. Colour and depth streams obtained from images can be given as input to these networks to get the corresponding outputs in grasp localisation applications. Although there are several methods to obtain the grasp coordinates of an object, the geometric approach [9] and the data-driven approach [10] are frequently used. The geometric approach is an analytical method that works based on the physical knowledge (shape and size) of the object. The data-driven approach is an empirical method that uses ML techniques and has grown rapidly in recent years.

Deep learning algorithms, such as DexNet 2.0 [3], also focus on these similar challenges. DexNet 2.0 uses dual-arm parallel grippers as QP-CNN. However, they are limited to bin-picking applications, where the objects are placed in a bin and the robot will be picking the object from the bin. Deep learning algorithms, such as DexNet 3.0 [24] and DexNet 4.0 [23], are focused on heterogeneous grippers, which are dual-arm grasping with two different gripper modules. For instance, a parallel gripper in one hand and a suction gripper in the other hand, which doesn't focus on using the broad workspace offered by the dual-arm robot. Rather they focus only on picking from a bin and dropping in the bin, which is pretty much a tabletop setup. Segmented object proposals [22] can be obtained with a fully convolutional network which is also an inspiration for QP-CNN to be a fully convolutional network. QP-CNN uses the broad workspace of the dual-arm collaborative robot, not only on the small bin placed in front of the robot.

Vision-based systems have emerged as a critical component for enhancing the safety of collaborative robot operations by enabling accurate perception and navigation. These systems use cameras and other imaging devices to recognize and locate objects in the environment, allowing robots to navigate autonomously in unfamiliar indoor spaces [25]. Recent advancements in deep learning have propelled object detection and recognition significantly improving the precision of vision systems, thereby strengthening operational safety in collaborative robot operations and its ability to operate safely in human-shared spaces [26].

1.2 Challenges in the Existing System and Problem Description

• Previous works on robotic grasping primarily focused on single-arm localisation and execution.

• This idea of having the best grasp location for a single-arm grasping can lead to either longer pick-up time due to the best grasp location on the farther side of the robot arm or unsuccessful picking due to wrong choices of end-effector pose due to limited dexterity.

In a robot manipulation scenario, some objects have the best grasp in any of its physical features. For instance, a teacup which can be grasped from multiple locations will have the best grasp in its handle. In a robot manipulation scenario, the farther the best grasp location from the gripper, the longer will be the execution time. So, it is significant to create an algorithm to reduce the execution time while maintaining the best accuracy.

2. QP-CNN

The proposed QP-CNN takes the depth image as input. Then it decides which region the object is located in, whether the right arm region or the left arm region. Once the region is identified, the best grasp is predicted on the right side or the left side or the top of the object. Figure 2 shows a teacup placed in various locations in the workspace. In Fig. 2(a) and (b), a teacup is located on the left side of the robot where the best grasp points are on the left side and right side, respectively. In Fig. 2(c) and (d), a teacup is located on the right side of the robot where the best grasp points are on the right side and left side, respectively. The arm to be employed for the robot manipulation is calculated by the Algorithm. Then the algorithm predicts the quality and pose of grasps at every pixel. Contemporary deep learning techniques face the challenge of high computation time. When the input is sourced from a depth image, this challenge can be overcome. The Jacquard and Cornell grasp datasets are used, in which the inputs to the network are the depth images and RGB images. The QP-CNN gives gripper selection, grasp angle, grasp width, and grasp quality, which when combined, predicts the best grasp pose among the grasp sets generated. The architecture of QP-CNN is shown in Fig. 3.

3. Working Methodology

Depth image is the source of information for the robot to make decisions on object manipulation. This information is obtained using a Kinect sensor, which is mounted above the workspace. The depth image thus obtained will be processed into QP-CNN, which gives four outputs, namely, grasp quality, grasp width, grasp angle, and selected arm. The grasp width and grasp angle would be sent to the robot in action along with the highest quality grasp information for the selected arm. Validating the outputs is done in the physical environment. ABB YuMi dual-arm collaborative robot is the physical robot used for validation. Cornell or Jacquard datasets are used for training the network. The algorithm was tested on 89 objects and the results are tabulated.

The IoU is calculated by the mathematical equation:

$$\text{IoU} = \frac{\text{Number of successful grasps}}{\text{Number of unsuccessful grasps}} \quad (1)$$

3.1 Depth Data Imaging

The Kinect Xbox 360 was used to capture the required inputs. It uses structured lighting for capturing depth images with 30 frames per second of depth sensing. The sensor used is in a fixed position throughout the process (overhead position) as shown in Fig. 6. There is sufficient lighting around the camera. The camera's position directly above the workspace facilitated the capturing of object images with its orientation. The robot, when moved, does not collide with the camera and it does not appear in the camera's frame which makes it more convenient to capture images of the objects to perform grasping. The field of view of the object in the image is made comparable with the images in the Cornell dataset and Jacquard dataset. As shown in Figs. 4 and 5, it is verified that the obtained images are comparable to the dataset. The processing is done in a computer with Ubuntu 18.04 LTS with a AMD A6-7480 with Radeon R5 Graphics Desktop Processor 2 cores up to 3.8GHz, 16GB RAM and NVIDIA GeForce GTX 1080 Ti GPU. We used a dual-arm ABB YuMi with a payload capacity of 0.5 kg for each arm.

3.2 Grasp Detection:

The depth image (D) rendered is divided into two regions: left region and right region. This depth image decides which arm to act according to the grasp object location as well as best grasp localisation.

To execute a grasp successfully, we should understand that there are two grasps to be considered, the grasp generated in the image g' and the grasp executed by the robot end effector g. The relation between g' and g is given by

$$g = t_{\text{RC}}(t_{\text{CI}}(g')) \quad (2)$$

Where, t_{RC} = Camera to robot frame transform
t_{CI} = Image to camera frame transform
g is the end effector grasp = (P, Φ, w, q)
g' is the grasp generated in the image = (l, Φ', w', q)
P is the pose of the end effector in (x, y, z)
l is the pixel location in the image (x', y')
Φ, Φ' are the grasp angle of end effector and image
w, w' are the grasp width of the gripper and image
q is the grasp quality

If we map all the grasps generated in the image as a grasp map G, which we denote

$$G = (\Phi,\ W,\ Q)\ \epsilon\ \mathrm{R}^{3\times H\times W} \quad (3)$$

The efficient way of predicting the best grasp is to calculate grasp g' for every pixel in the depth image D. To calculate the same, a function NN(D) can be defined, where NN(D) = G, *i.e.*, the function provides grasp map as output. As using a neural network is the most efficient way of approximating a complex function, we also use the same: NN : D →G with an L2 loss function.

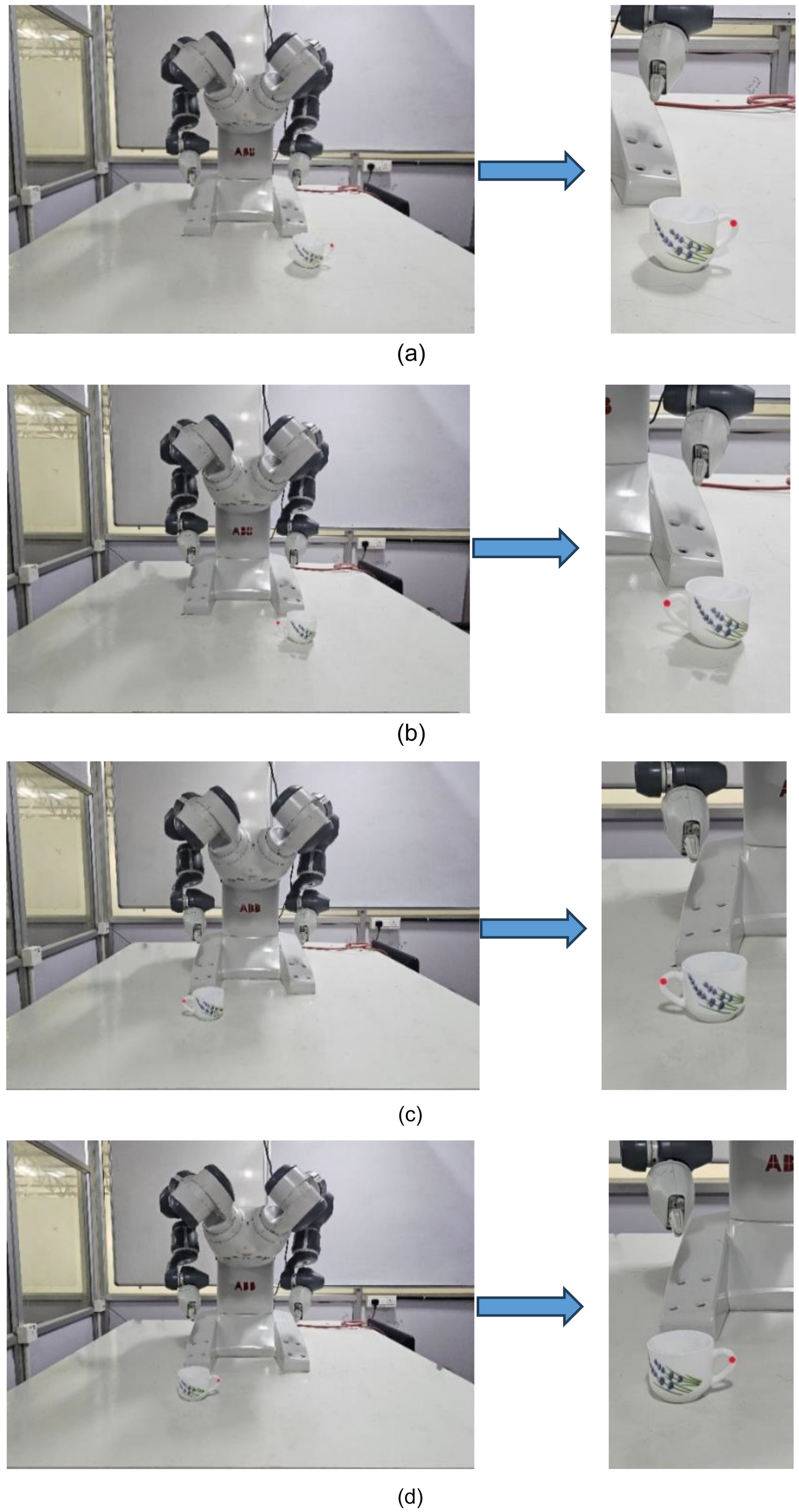


Figure 2. Object handling by a dual-arm robot.

NN_θ is the neural network with weights θ. Which can be determined by

$$\theta \; = \underset{\theta}{\mathrm{argmin}} \mathcal{L}(G_T,\ \mathrm{NN}_\theta(D_T)) \tag{4}$$

Q is an image that indicates grasp quality at each point $(x',\ y')$ in the image, which is a key parameter in classifying a grasp as a successful grasp or unsuccessful grasp. The range of Q is between 0 and 1. If the value of Q is close to 0, it indicates the lowest grasp quality and if Q is close to 1, it indicates the highest grasp quality.

Φ is another image which describes the grasp angle. The grasp angle is another key parameter in deciding a successful grasp. The range of Φ is between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$. The major task in computing a grasp angle using neural network is to provide two components in the range and we

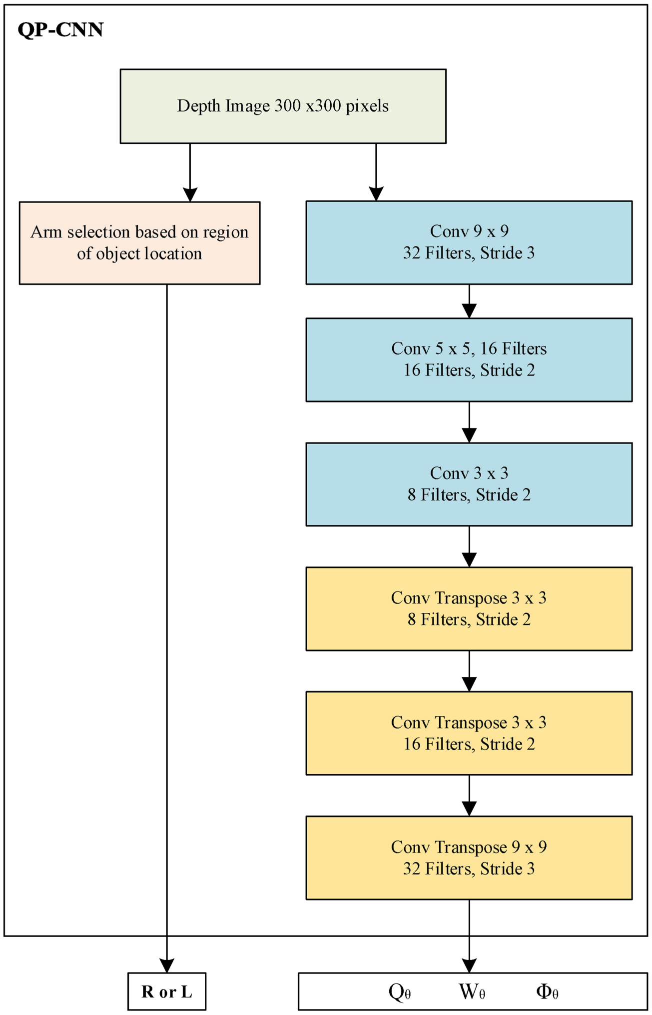


Figure 3. QP-CNN architecture.

Figure 4. RGB and depth frame obtained for object 1 through Kinect.

have used two components, $\sin(2\Phi_T)$ and $\cos(2\Phi_T)$ as they are symmetrical and provide values in the range.

W is also another image which describes the gripper width, which is an important parameter in deciding a successful grasp. If the gripper is not sufficiently open, then the robot will not be able to pick that object and results in an unsuccessful grasp. It is measured in pixels and in between 0 pixels and 150 pixels. To fit 0 to 150 pixels in the range of [0,1], we scaled the pixels by $\frac{1}{150}$.

The QP-CNN is a fully CNN as they work well with any image domain transfer tasks [21]–[23]. When fed into

Figure 5. RGB and depth frame obtained for object 2 through Kinect.

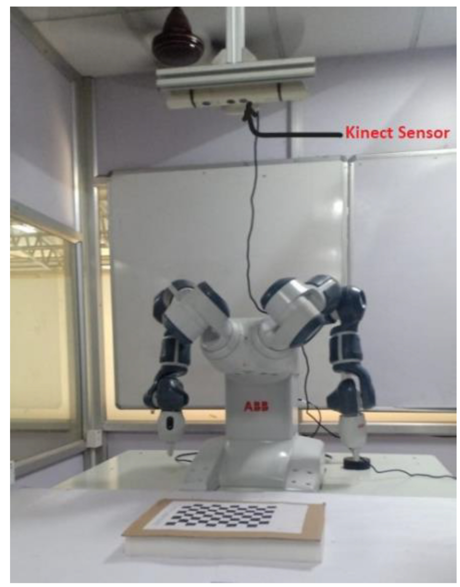


Figure 6. Placement of the Kinect in the workspace.

the network, all the images in Q_θ, Φ_θ, and W_θ are converted into 300 × 300 pixel images. From the output, we can calculate the grasp angle by

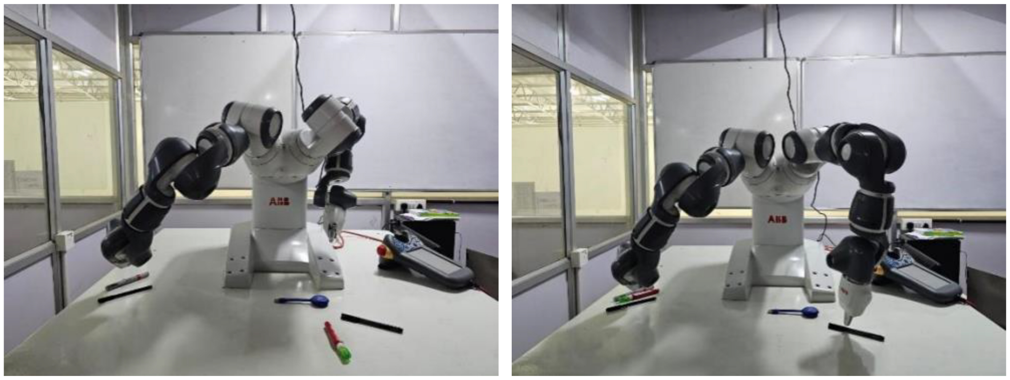
$$\Phi_\theta = \frac{1}{2} \arctan \frac{\sin(2\Phi_\theta)}{\cos(2\Phi_\theta} \tag{5}$$

3.3 Dataset

The training dataset used is based on the Cornell Grasping Dataset [13], which comprises of 885 RGB-D real objects images. This dataset also has 5,110 positive grasps and 2,909 negative grasps which are labelled by humans. We have augmented the dataset by rotating, zooming and cropping to build a set of our own 8,850 depth images. We used 80% for our training dataset and 20% for our evaluation dataset.

3.4 Workspace

As seen in Fig. 6, the workspace is aligned in such a way that the robotic arm does not interfere with the images captured by the Kinect sensor. Possible noises created by the sunlight are neglected as the project is carried out in a closed space. A camera placed directly



(a) (b)

Figure 7. Experiments.

Table 1
Detection Accuracy (%) of Different Methods

Input Size	Method	Accuracy (%) Object-wise	Reliability (Φ)	Time (ms)
300×300	Efficient grasping-D	95.5	73	6
	Efficient grasping-RGB	91	47	6
	Efficient grasping-RGB-D	97.8	89	6
	GR-ConvNet	96.8	91	4
	QP-CNN	**98.1**	**100**	**2**

above the object captures a depth image which is then provided as input to the algorithm which generates the best antipodal grasps with parameters such as grasp quality angle and width. This grasp location is provided to the controller which then grasps the object with a manipulator. A significant limitation of using a single-arm robot is its limited workspace, which has been addressed using a dual-arm robot. Hence, the size of the workspace is also larger.

4. Results and Discussions

4.1 Performance Analysis

Our algorithm QP-CNN is implemented for dual-arm grasping, where objects are placed in the broad workspace as shown in Fig. 7. The performance analysis is carried out on two parameters: a) Object-wise accuracy and b) Execution time with four other prominent methods. The four prominent methods are (i) Efficient Grasping-D, (ii) Efficient Grasping-RGB, (iii) Efficient Grasping RGB-D, and (iv) GR-ConvNet. These four prominent methods are selected based on the input size to the network which is 300×300. The performance results are tabulated below in Table 1.

The results show that the object-wise grasp accuracy has increased from GR-ConvNet, which gives an accuracy of 96.8%. This increment in accuracy is due to the improved dexterity in handling the object as a result of the novel QP-CNN algorithm. The results are charted in Fig. 8.

The grasp execution time for our method has been reduced to one-third of the efficient grasping-D, efficient grasping-RGB, and efficient grasping-RGB-D methods and half of GR-ConvNet. This is because, in our method, QP-CNN, the nearest gripper to the object is selected for the manipulation task. As the gripper approaches the object quicker, the overall execution of the grasp has become faster. The results are charted in Fig. 9. The algorithm showed positive results on custom images that were given as input. It almost accurately predicted the grasp rectangle for each object. The only possibilities of unsuccessful grasp localisation were either due to any obstacle present or bad lighting.

Reliability (Φ) is another key feature in a successful robotic grasping task for a given policy [23]. In this study, we involved level 1 objects as seen in Fig. 7 and level 2 objects [23] as seen in Figs. 2, 4, and 5. We added a total of 43 house-hold objects, weighing less than 500 g, placed in the extended workspace.

Reliability [23] is measured by,

$$\Phi(\pi) = \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1} R_T\right] \tag{6}$$

Where, T = Number of grasp attempts

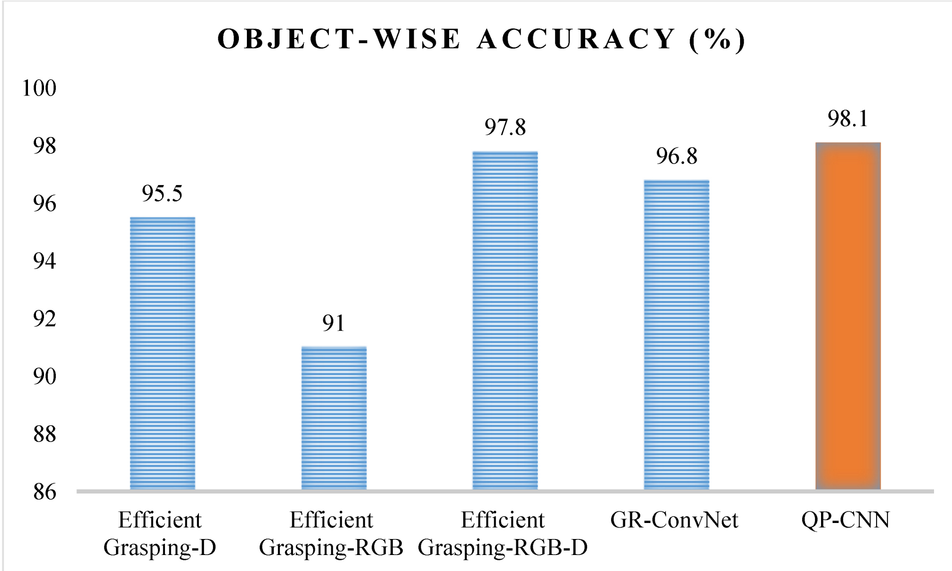


Figure 8. Object wise accuracy (%).

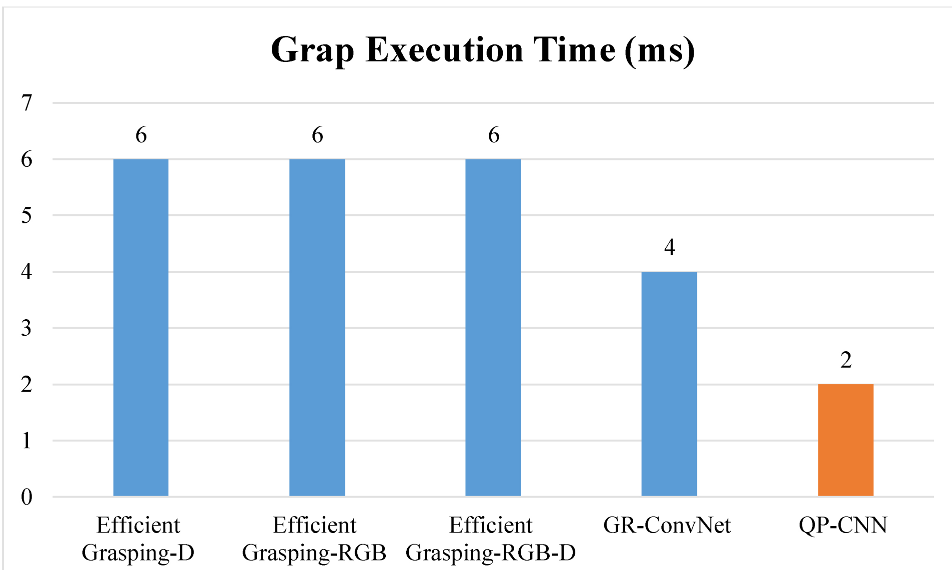


Figure 9. Grasp execution time (ms).

π = QP-CNN

R_T is the reward given if the robot pick is successful, which is going to be 1 for a successful grasp and zero for any unsuccessful grasp.

The QP-CNN has demonstrated a reliability of 100% for level 1 as well as level two objects. The agility offered by the QP-CNN to the dual-arm collaborative robot is the reason for these humongous feet.

5. Conclusion

QP-CNN algorithm had simple requirements and was quite efficient as well. It presented an instantaneous and object-independent grasp localisation technique. The network was able to identify objects and predict quality and the angle of grasp pose at every pixel along with the choice of arm to be used. Taking depth images as input, the algorithm was known to produce effective results within a short period. This approach presented an opportunity to overcome a common limitation in many other DL algorithms, namely, extended computation time due to overused dexterity to handle and manipulate objects. It can enable closed-loop grasping [15] as well. The algorithm gives the grasp coordinates and also enables visualisation of the grasping location on an RGB image that was provided as input. The algorithm was trained on the Cornell grasp dataset and can be tested on both the Cornell grasp dataset [13] and the Jacquard dataset [14]. The Cornell grasp dataset and a sample of the Jacquard dataset are publicly available. This makes it easier to access the dataset for running inference. The novel algorithm produces an accuracy of 98.1% and a grasp execution time of 2 s with a reliability of 100%. This

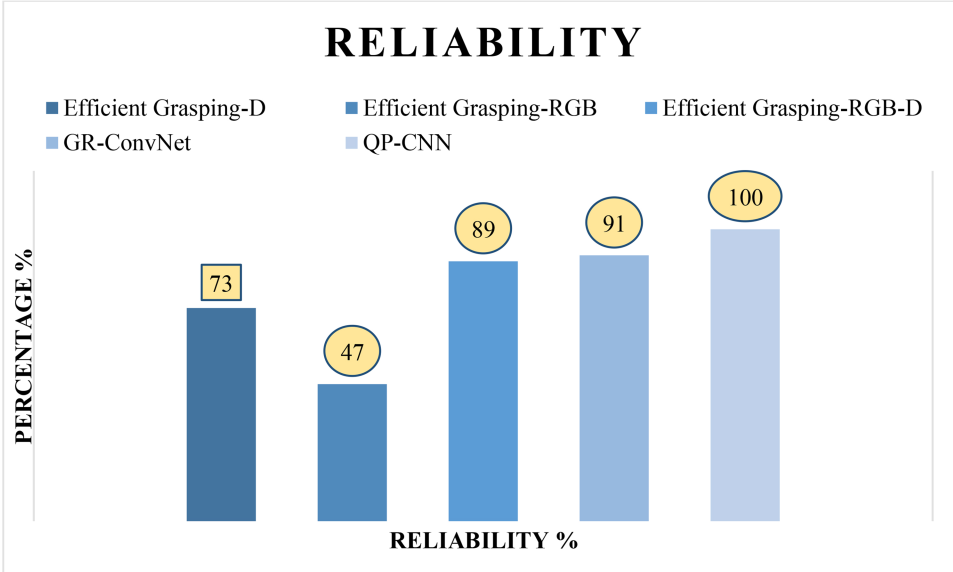


Figure 10. Reliability (%).

research can be extended in the future by simultaneous localization techniques [18]–[20] for dynamic objects and using multiple modalities suggested in [17]. QP-CNN is trained to pick level 1 and level 2 objects successfully in a standard-clutter environment, where objects are placed in the workspace separated by a distinct space from each other with 100% reliability. But it is not trained to pick in an unstructured environment where objects are either touching each other or overlapping with each other. Because the grasp location of the target objects may be classified as false positives or the target objects are partially obscured from the camera's view, the best grasp locations may be missed or hidden, which may result in reducing the success and reliability of the grasp. Hence In a future study, QP-CNN can be trained to pick objects successfully in unstructured environments with 100% reliability

References

[1] A. Hippolitus, R. Josin Senthilnathan, and O. Malla, Simulation of grasp localization inferences for a dual-arm collaborative robot, *IOP Conference Series: Materials Science and Engineering*, *1012*(1), 2021.

[2] K. Santhakumar and H. Kasaei, Lifelong 3D object recognition and grasp synthesis using dual memory recurrent self-organization networks, *Neural Networks 150*, 2022, 167–180.

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

[4] A. Sahbani, S. El-Khoury, and P. Bidaud, An overview of 3D object grasp synthesis algorithms, *Robotics and Autonomous Systems*, *60*(3), 2012, 326–336.

[5] D. Prattichizzo and J.C. Trinkle, Grasping, in *Handbook of Robotics*. Berlin: Springer, 2008, 671–700.

[6] A.T. Miller, S. Knoop, H.I. Christensen, and P.K. Allen, Automatic grasp planning using shape primitives, *Proceeding of the IEEE International Conference on Robotics and Automation (ICRA)*, Taipei, 2003, 1824–1829.

[7] C. Goldfeder, P.K. Allen, C. Lackner, and R. Pelossof, Grasp planning via decomposition trees, *Proceeding of the IEEE International Conference on Robotics and Automation (ICRA)*, Rome, 2007, 4679–4684.

[8] R. Detry, E. Baseski, M. Popovic, Y. Touati, N. Kruger, O. Kroemer, J. Peters, and J. Piater, Learning object-specific grasp affordance densities, *Proceeding. of the IEEE International Conference on Development and Learning (ICDL)*, Shanghai, 2009, 1–7.

[9] L. Porzi, S.R. Buló, A. Penate-Sanchez, E. Ricci, and F. Moreno-Noguer, Learning depth-aware deep representations for robotic perception, *IEEE Robotics and Automation Letters*, *2*(2), 2016, 468–475.

[10] J. Bohg, A. Morales, T. Asfour, and D. Kragic, Data-driven grasp synthesis—A survey, *IEEE Transactions on Robotics 30*(2), 2013, 289–309.

[11] A. Bicchi and V. Kumar, Robotic grasping and contact: A review, *Proceeding. of the IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, CA, 2000, 348–353.

[12] Cornell grasping dataset. Accessed: Oct. 1, 2013. http://pr.cs.cornell.edu/grasping/rect data/data.php

[13] A. Depierre, E. Dellandrea, and L. Chen, Jacquard: A large-scale dataset for robotic grasp detection, *Proceeding IEEE International Conference on Intelligent Robots and Systems*, Madrid, 2018, 3511–3516.

[14] I. Lenz, H. Lee, and A. Saxena, Deep learning for detecting robotic grasps, *The International Journal of Robotics Research*, *34*(4–5), 2015, 705–724.

[15] H. Cao, G. Chen, Z. Li, J. Lin, and A. Knoll, Lightweight convolutional neural network with Gaussian-based grasping representation for robotic grasping detection, 2021, *arXiv:2101.10226*.

[16] D.S. Maini and A.K. Aggarwal, Camera position estimation using 2D image dataset, *International Journal Innovations in Engineering Technology*, *10*, 2018, 199–203.

[17] A.K. Aggarwal, GPS-based localization of autonomous vehicles, *Autonomous Driving and Advanced Driver-Assistance Systems (ADAS)*. Boca Raton, FL: CRC Press, 2021, 437–448.

[18] A. Kumar, T. Oishi, S. Ono, A. Banno, and K. Ikeuchi, Global coordinate adjustment of 3D survey models in world geodetic system under unstable GPS condition, *Proceeding 20th ITS World CongressITS Japan*, Tokyo, 2013, 1–10.

[19] A.K. Aggarwal, Autonomous navigation of intelligent vehicles using vision-based method, *International Journal Research Electronics & Communication Technology*, *3*(5), 2015, 1–10.

[20] V. Badrinarayanan, A. Kendall, and R. Cipolla, SegNet: A deep convolutional encoder-decoder architecture for image

segmentation, 2015, *arXiv:1511.00561* 5.

[21] J. Long, E. Shelhamer, and T. Darrell, Fully convolutional networks for semantic segmentation, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, 3431–3440.

[22] J. Yang, B. Price, S. Cohen, H. Lee, and M.H. Yang, Object contour detection with a fully convolutional encoder-decoder network, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 2016, 193–202.

[23] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg, Learning ambidextrous robot grasping policies, *Science Robotics*, *4*(26), 2019, eaau4984.

[24] J. Mahler, M. Matl, X, Liu, A. Li, D. Gealy, and K. Goldberg, Dex-net 3.0: Computing robust vacuum suction grasp targets in point clouds using a new analytic model and deep learning, *Proceeding IEEE International Conference on robotics and automation (ICRA),* 2018, 5620–5627.

[25] J. Qiao, J. Guo, and Y. Li, Vision-based robot indoor-positioning and navigation method research, *International Journal of Robotics and Automation*, *39*(10), 2024, 1–9.

[26] X. Zhu, K. Zhang, and X. Hua, Consistency analysis and suggestions of collision measurement in human–robot collaboration safety evaluation, *International Journal of Robotics and Automation*, *39*(10), 2024, 1–13.

Biographies

A. Josin Hippolitus received the master's degree in mechatronics engineering from Anna University. He is currently pursuing the doctoral degree in robotics and deep learning. He is working as an Assistant Professor with the Department of Mechatronics Engineering and also currently heading the Mechatronics Laboratory, SRM Institute of Science and Technology, Chennai, India. His research interests include deep learning, computer vision, and robotics. His past experiences include teaching and guiding student projects in developing various robotics and automation projects. He has presented his research findings in various national and international conferences.

R. Senthilnathan received the master's and doctoral degrees in mechatronics engineering from Anna University, Chennai, India. He is working as a Professor with the Department of Mechatronics Engineering and also currently heading the Center for Immersive Technologies, SRM Institute of Science and Technology, Chennai. His areas of interest include computer vision, deep learning for perception, robotics, general automation through mechatronics approach and immersive technologies. He has published over 40 research articles in various international journals and conferences. His past experience in development of multiple automata, such as autonomous mobile robots, computer vision system for tracking, measurement, *etc.*. He has successfully completed multiple sponsored projects.