ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/357610159

Robotic Pick and Assembly Using Deep Learning and Hybrid Vision/Force Control

Conference Paper · August 2021

DOI: 10.1109/NIR52917.2021.9666138

CITATION
1

READS
4

5 authors, including:

Riby Abraham Boby
Innopolis University
23 PUBLICATIONS **123** CITATIONS

SEE PROFILE

Abdelaziz Wasfy Shaarawy
Innopolis University
2 PUBLICATIONS **1** CITATION

SEE PROFILE

Alexandr Klimchik
Innopolis University
176 PUBLICATIONS **1,539** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Development of parallel robotic platforms employing twisted string actuators View project

Project Research and Development of Software Solutions for Static and Dynamic Based Control of Anthropomorphic Bipedal Robots Locomotion View project

All content following this page was uploaded by Riby Abraham Boby on 11 January 2022.

The user has requested enhancement of the downloaded file.

Robotic Pick and Assembly Using Deep Learning and Hybrid Vision/Force Control

Karam Almaghout Riby Abraham Boby Mostafa Othman Abdelaziz Shaarawy Alexandr Klimchik

Innopolis University
Innopolis, Russian Federation
k.almaghout@innopolis.university, ribyab@gmail.com,m.othman@innopolis.university,a.shaarawy@innopolis.university, a.klimchik@innopolis.ru

***Abstract*—Pick up of featureless cylinders using deep learning and assembly using combination of visual and force feedback is discussed in this article. Object detection using YOLO technique is used to detect the object for robotic pick up. It was possible to identify the object even in cluttered environments. The assembly of the cylindrical object in a hole with tight clearance is also discussed. The method involves hybrid control i.e., position control along the assembly plane and force control along the direction of normal to the plane. The method is implemented using KUKA iiwa robot and the results are presented.**

***Index Terms*—robotics, object detection, pose measurement, assembly, vision, deep learning, localization**

I. Introduction

Manipulators are widely used to achieve repetitive common assembly tasks in industry, due to their speed and repeatability [1]. However, the less the clearance between the parts that should be assembled the more the task becomes challenging to be robotized. Peg-in-hole assembly represents a proper example of these challenging tasks where in some cases the clearance between the peg and the hole is less than the robot's accuracy.

Many approaches have been used to solve this [2]. Control strategies that have been used for this task can be classified into three main categories: position control, force control, and hybrid position/force control. Vision sensor is a proper and affordable solution for position control to localize the pose of the hole and complete the assembly task. Korta et al. introduced a vision-based approach for robotic arm to achieve electronics circuits assembly task [3]. In [4], researchers proposed a vision-guidance method for the Baxter robot for cylindrical peg-in-hole assembly. Triyonoputro et al. [5] developed a new method based on deep learning and multi-view images for peg-in-hole assembly. The trained deep neural network predicts the quadrant of the hole, and by iterative visual servoing the robot moves the peg towards the hole step by step. In force control, the position of the hole is predefined. However, since the clearance is smaller than the robot accuracy, force perception is used to localize the hole for these small errors. Spiral search that was proposed by [6] and [7] is one of the most popular blind approaches that use only force perception for localization in small clearance. In this class, the force oscillations, that occur due to a partial engagement between the peg and the hole, are analyzed to overcome the positioning inaccuracy and complete the assembly. In the vision-based systems, the robot can detect the hole pose. However, the kinematics of the robot [8], uncertainty in estimating camera parameters [9], affect the robustness of the process, and cause a failure in obtaining the accurate position during assembly. On the other hand, force control provides a feasible solution for the small clearance and also prevent excessive interaction forces during the assembly. However, it fails in dynamic environment, where the uncertainty in hole and the peg poses may change within larger range. The hybrid position/force control methods utilize both vision and force sensors which offers more reliable and robust systems. Abdullah et al. [10] suggested a new strategy for the peg-in-hole assembly that mimics the human behavior. In their strategy, the robot firstly, locates the hole position by a camera. Then, it proceed with accomplishing the task depending on the interaction forces measured by a 6-DOF force/torque (F/T) sensor. Another strategy based on vision/force guidance was introduced in [11]. In this research, a dual-arm Baxter robot has been used to achieve the peg-in-hole for a clearance 0.5 mm. The vision feedback is used for the position adjustment between the peg and the hole, while the orientation adjustment is held by the F/T sensor. In regard to the peg grasping, the use of camera and computer vision techniques allow the robot to detect the the peg pose and grasp it.

Conventional object detection methods depend on the shape [12], color [13], or edge [14] of the object. These methods work efficiently in where there is no significant fluctuation in the environment characteristics, like the illumination and the number of existing objects. In cluttered environments, the grasping task is not only a matter of detection, but also robot should recognize the objects within the environment and pick up the desired one. Machine Learning techniques, especially Deep learning, proved their superiority over the conventional object detection ones, and showed a reliable performance for robot grasping in the dynamic and cluttered environments [15]–[17]. Sometimes the task makes it necessary to use point cloud data as well [18], which increase the computation load, total cost of the system and processing time. Therefore the use of minimum number of images from a single camera is more desirable. Most of the conventional

978-1-6654-2407-3/21/$31.00 ©2021 IEEE

approaches uses objects with textures or features that can be tracked, however this is not true for objects in industry [19].

In this paper, we propose a new approach to robotize the peg-in-hole assembly using a serial manipulator equipped with an eye-on-hand camera and F/T sensor. The proposed approach addresses the challenges in the both parts of the peg-in-hole task: the pick-up of a featureless object from a cluttered environment, and assembly with a clearance less than the robot accuracy. We implemented a deep learning technique to detect and recognize a cylindrical object (peg) randomly located among other objects. In addition to detect the peg, visual servoing is used to detect the hole in three stages i.e., rough, fine and continuous modes of localization. F/T sensor measurements are considered to maintain the interaction forces during the assembly, and to adjust the camera orientation in case the hole is not visible after. The main contributions of this paper are:

1) Deep learning based detection of black featureless object;
2) assembly of a cylindrical object using hybrid control i.e., position and force control in a hole with low clearance; and practical implementation of pick up and assembly using KUKA iiwa robot.

The remainder of this paper is organized as follows: In section II, we discuss the object picking-up part of the method, including the selected deep learning model and the steps of preparing the dataset and training the model. The assembly part of the method and its three localization stages are presented in Section III. Experiments results are discussed in Section IV. Finally, Section V concludes the the paper.

II. Object Picking-up

In robot visual perception, object detection in a cluttered environment is considered as a challenging problem by the computer vision community [20]. Redmon and Farhadi developed a real-time object detection method named YOLO based on Convolutional Neural Networks [21]. In this work, we utilize YOLOv5, the up-to-date version of YOLO [22]. Fig. 1 shows the cylindrical object, which the robot should detect and grasp, along with two other random objects (cubes).

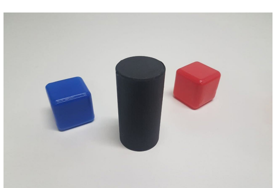


Fig. 1. The object of interest, a black featureless cylindrical object, along with two other objects, blue and red cubes.

A. Training the Model

Multiple (75) images were captured in various configurations for the objects including those in which the objects are occluded. Fig. 2 shows the labeled images. Then, they were re-oriented and resized to 640x640 to suit the model requirements. In addition, some augmentation are applied such as vertical and horizontal flip, rotations, cropping, skewing, illumination, and adding noise. These augmentations exposes the model to a variety in the dataset, thus makes the model more generalized. The model is trained for 300 epochs considering batch size 64, image size 640, and default pre-trained weights. In the mean time, the training process was being monitored online using weights and biases platform to visualize the losses and overall performance of the model [23]. Fig. 3 presents the recall and precision of the trained model. More information of the trained model are available at https://wandb.ai/mostafa_metwally/yolov5-simple3d-shapes/reports/NIR-Conference-YOLO5-v4-on-Simple-shapes-dataset--Vmlldzo0OTA4OTE

Fig. 2. Labeled training images.

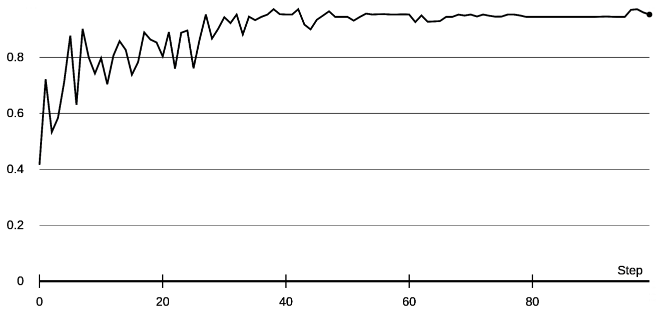
B. Model Implementation

The trained model is implemented in the task pipeline as the first stage. Using an RGB camera, the model detects and localizes the black cylinder that may be arbitrary located among a bunch of other objects. Then, it will return the image coordinates of the cylinder position which is further transformed to the robot base coordinates using the camera calibration data. The robot is commanded to move to the estimated position of the cylinder to be grasped.

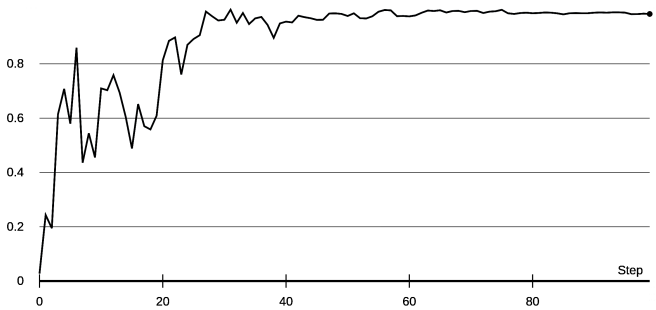
III. Assembly

After the cylinder is picked up, the robot moves to the assembly area. The position of the hole is then determined using sensor information. The robot will accomplish the final localization and assembly using visual and force servoing. Due to the inaccuracies in the camera calibration, robot kinematics, etc., this was done in three main stages:

1) Rough localization wherein the hole center position is detected when the robot is over the assembly region.
2) Fine localization using sensor information after positioning the robot closer to the plane of assembly according to the rough estimate obtained earlier.



(a) Recall



(b) Precision

Fig. 3. Model evaluation metrics: recall and precision of the trained YOLO model

3) Continuous localization that involves contact with the assembly plane and obtaining continuous visual feedback while in force control mode.

Once the peg engages with the plane of the hole, an Image Based Visual Servoing Approach as shown in Fig. 4 is applied. Note that the hole is partially occluded by the peg in this situation and only a small region of the hole may be detected. Therefore at this stage only the edge of the hole is detected within a smaller region of interest. A difference of Gaussian image is obtained for the region of interest. Later the edge is located in this image. Depending upon the edge direction, the direction of movement for engaging with the hole is determined. Figure 5 shows the region of interest, Difference of Gaussian images and the calculated direction. There exists a possibility that the hole position is completely occluded by the peg. In such a case the force information is used to re-orient the peg and restart the hole search process. The information about this is added in Appendix A. A successful localization is detected by movement of the peg beyond the plane of assembly and decrease in the force measured in the direction of the normal to plane.

IV. Experiments Results

A. System Setup

The system used in the experiments is shown in Fig. 6. We used KUKA LBR iiwa 14 equipped with an RGB camera of resolution 1280×480 mounted on the robot's end-effector. KUKA iiwa robot has inherent torque sensors mounted on its joints, and it is possible to make measurement in any of the coordinate frames of the robot. F/T measurements at the end-effector are obtained from the robot's controller. The robot is connected to the computer via an Ethernet interface. The pipeline of the robot is written in C++ in Robot Operating System (ROS) [24] and based on ROS metapackage for the KUKA LBR iiwa developed by Hennersperger et al. [25]. The pseudocode of the pipeline is presented in Algorithm 1.

TABLE I
Results of hand eye calibration for images captured from 8 different poses

	Position (mm)			Orientation (rad)		
Mean	-6.6	51.8	-29.4	0.006	-0.005	3.1
STD	2.6	1.6	1.1	0.003	0.005	0.0009

Algorithm 1: Pseudocode of the system pipeline

Data: Intrinsic Matrix and Hand-eye calibration data.
Zg: Z coordinate of the gripper.
Zt: Z coordinate of assembly plane.
Fz: external force acting on the end-effector along Z axis.
Ft: the allowable engagement force.
$pos1$: position over object for detecting the pose.
$pos2$: position over hole for detecting hole position.

begin
  Initialization;
  Move to $pos1$;
  Localize the Peg using YOLO;
  Move to the detected XY postion;
  while $|Fz| > Ft$ **do**
    Move downwards along Z;
  end
  Pick up the Peg;
  Move to $pos2$;
  Do $Rough\ localization$ for the hole position;
  Move Closer and approach the detected XY position;
  Capture image and do $Fine\ localization$;
  Align in XY plane with estimated position;
  while $|Fz| < Ft$ **do**
    Reduce Z_g;
  end
  Measure Zt;
  while $|Fz| \geq Ft$ **do**
    while *Hole is not detected in the image* **do**
      Measure moments mx and my;
      Rotate the camera around Z according to $\tan^{-1} \frac{-m_x}{m_y}$;
    end
    Detect the hole edge and measure the furthest point of the center of the Peg;
    Move, by steps, towards the direction of the furthest point of the hole edge;
    while $|Fz| < Ft$ **do**
      Move downwards by dz;
      if $Zg < Zt$ **then**
        break;
      end
    end
    Assemble the peg;
  end
end

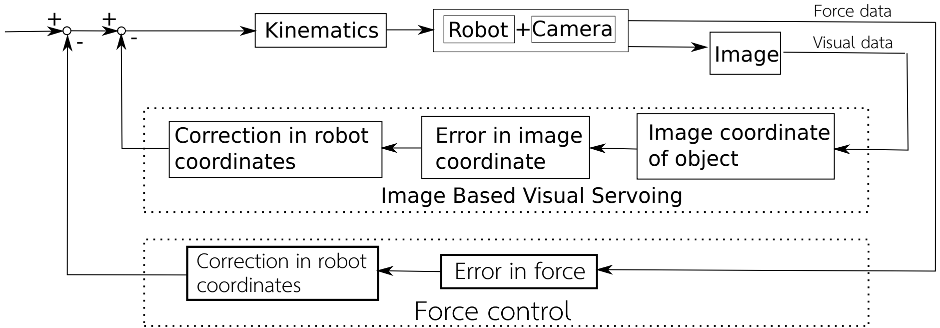
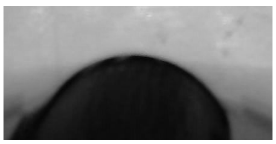
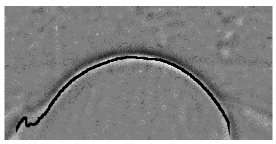


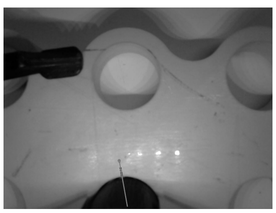
Fig. 4. Block diagram of the Visual servoing strategy. While the inner loop deals with the visual feedback and searches for the hole, the outer loop maintains the interaction forces during the task.



(a) Region of interest



(b) Difference of Gaussian



(c) Detected direction

Fig. 5. Illustration of the IBVS method

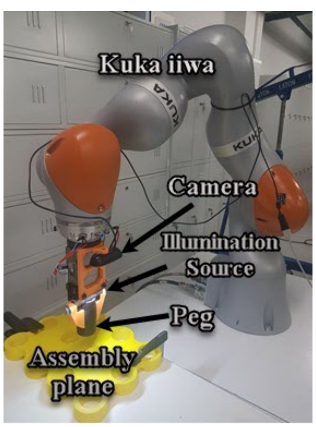


Fig. 6. System Setup: KUKA iiwa robot, equipped with two-jaw gripper and an RGB camera at its flange. Illumination source is added to enhance the detection process of the peg and the hole.

B. Calibration

Camera calibration and image processing was done in Python using OpenCV libraries [26]. The calibrated camera parameters were used for the hand-eye calibration. The calibration method proposed in [27] was used to achieve the hand-eye calibration of the camera. Fig. 7 shows the calibration grid

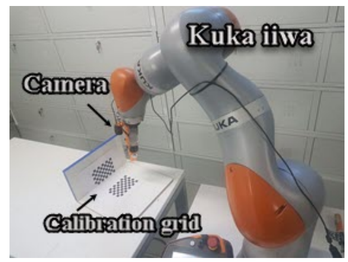
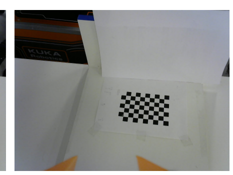


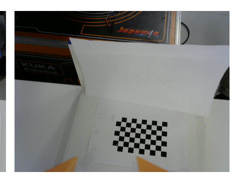
Fig. 7. Hand-eye calibration with two grids attached perpendicular to each other.

(a) Pose 1: horizontal

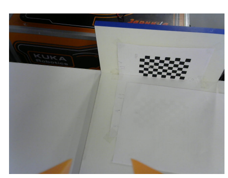


(b) Pose 1: Perp.

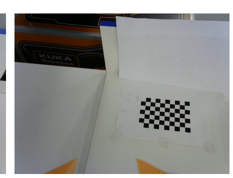
(c) Pose 2: horiz.



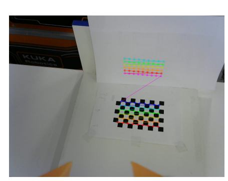
(d) Pose 2: Perp.



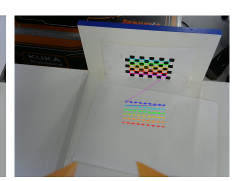
(e) Pose 3: horiz.



(f) Pose 1: Perp.



(g) Pose 1: horiz.



(h) Pose 1: Perp.

Fig. 8. Sample images used for hand-eye calibration, (a) to (f): three sample poses. (g) to (h): projected points on (a) and (b), respectively.

as well as the camera mounted on the robot end-effector. The robot kinematics was used to define a coordinate frame on the calibration grid. Using the relative pose between the robot and the calibration gird as well as the pose of the robot-end effector the hand-eye relation was estimated. Different random poses of the robot such that the calibration grid was visible in the camera to verify the estimate obtained in this fashion. The results of hand-eye estimate obtained for different poses of the camera are shown in Table I. The sample images used for calibration are shown in Fig. 8.

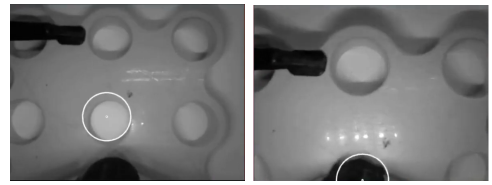
C. Results

Screenshot samples of the deep learning output detecting the object of interest within a cluttered environment are shown in Fig. 9. The model was able to detect the object in different positions and orientation for the robot. The obtained position of the object is commanded to the robot to be grasped.
Fig. 10 presents the rough and fine localization of the hole in the assembly stage. The robot detects the closest hole in the rough localization. Then, it approaches the detected position and re-localizes the circle. Next step, the robot moves towards the assembly plane. Once the collision with the assembly plane occurs, the IBVS method starts searching the hole edge and commands the robot to move, in steps, towards the farthest point of the edge of the hole. Fig. 11 illustrates how the robot approaches the hole in this phase of the assembly stage. Video of process can be accessed from video.

Fig. 9. Samples of the deep learning model output.

V. Conclusion

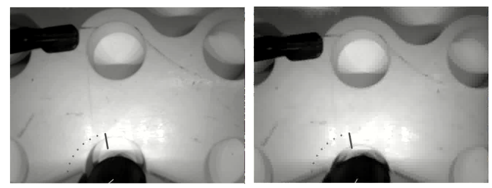
In this paper, the problem of fully robotizing the peg-in-hole task starting from picking the peg to placing it into the hole with a tight clearance is addressed. The proposed approach has two parts. Initially a deep learning model is used to detect the peg. It was possible to localize the peg



(a) Rough Localization: over the assembly region, closest hole is detected

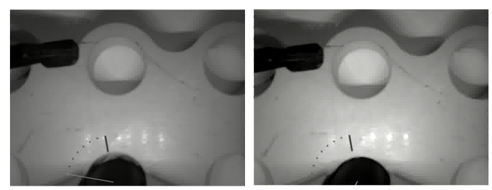
(b) Fine Localization: aligns with the detected hole position and re-localizes

Fig. 10. Camera output of the system during the rough and fine localization phases of assembly



(a) step 1: peg engaged with the assembly plane, the direction towards the hole is detected.

(b) step 2: re-detect and approach towards the hole.



(c) step 3: re-detect and approach towards the hole.

(d) step 4: the peg is completely over the hole, and the assembly task is accomplished.

Fig. 11. Image during the rough and fine localization phases of assembly

even in cluttered environment with multiple other objects. The next stage is assembly in a hole with tight clearance. The inherent positioning accuracy of a robot is lesser than the clearance commonly encountered in assembly tasks. Therefore the assembly part consists of three main stages to detect the hole position namely rough, fine, and continuous localization in hybrid control mode. Visual servoing is used for detection in all phases of the assembly one. F/T measurements were considered to maintain the interaction forces between the robot and the environment during the task and to reorient the camera in case the hole is not visible in the continuous localization stage based on F/T measurements. The proposed method has been implemented on KUKA iiwa robot and the details of the experimental results are presented.

Appendix

Exploiting F/T measurements to re-orient the camera

In some cases the robot may approach the assembly plane in a way that the peg is partially at the hole, Fig. 12. However, the hole edge is not visible to the camera. There is an equilibrium due to the presence of an action and reaction force. The force $\mathbf{f}_A$ is applied by the robot on the planar surface on which the holes are located. Whereas, $\mathbf{f}_R$ is the reaction force. The action force can be assumed to be applied along the Z axis of the frame T. The corresponding reaction force can be assumed to act at point R. If the position of the tool and F/T sensor coordinates are not known, the vector $\mathbf{p}_{O_T}$ is to be identified. Note that, $\mathbf{p}_R$ is the displacement in frame T. From Fig. 12, the following may be interpreted:

$$\mathbf{r} = \mathbf{p}_{O_T} + \mathbf{p}_R \tag{1}$$

The moment about O_T due to $\mathbf{f}_R$ can be expressed as

$$\mathbf{m} = \mathbf{r} \times \mathbf{f}_R \tag{2}$$

where both $\mathbf{f}_R$ and $\mathbf{m}$ can be measured. Note that Z axis of the F/T measurement frame is aligned with the axis of the peg along which force is applied. It can then be considered that the moment about Z axis of the sensor vanishes i.e $m_z \approxeq 0$. Also, $f_x \approxeq 0$ and $f_y \approxeq 0$, which gives the following expression:

$$\begin{bmatrix} m_x \\ m_y \end{bmatrix} = \begin{bmatrix} 0 & f_{R_z} & -f_{R_y} \\ -f_{R_z} & 0 & f_{R_x} \end{bmatrix} \mathbf{p}_R \tag{3}$$

Thus, the following relation is obtained:

$$\mathbf{p}_R = \begin{bmatrix} m_y / f_{R_z} \\ -m_x / f_{R_z} \end{bmatrix} \tag{4}$$

It can also be inferred that the direction of the hole is $\tan^{-1}(\frac{-m_x}{m_y})$. Once the estimate of $\mathbf{p}_R$ is obtained, it can be used for localizing the hole and for orienting the camera in case the hole position is not visible in the field of view.

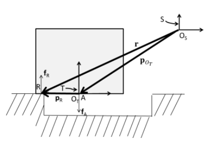


Fig. 12. Schematic of robotic peg-in-hole assembly.

References

[1] R. Ahmad and P. Plapper, "Safe and automated assembly process using vision assisted robot manipulator," *Procedia Cirp*, vol. 41, pp. 771–776, 2016.

[2] S. R. Chhatpar and M. S. Branicky, "Search strategies for peg-in-hole assemblies with position uncertainty," in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*, vol. 3. IEEE, 2001, pp. 1465–1470.

[3] J. Korta, J. Kohut, and T. Uhl, "Opencv based vision system for industrial robot-based assembly station: calibration and testing," *Pomiary Automatyka Kontrola*, vol. 60, 2014.

[4] Y. Huang, X. Zhang, X. Chen, and J. Ota, "Vision-guided peg-in-hole assembly by baxter robot," *Advances in Mechanical Engineering*, vol. 9, no. 12, p. 1687814017748078, 2017.

[5] J. C. Triyonoputro, W. Wan, and K. Harada, "Quickly inserting pegs into uncertain holes using multi-view images and deep network trained on synthetic data," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 5792–5799.

[6] I. F. Jasim, P. W. Plapper, and H. Voos, "Position identification in force-guided robotic peg-in-hole assembly tasks," *Procedia Cirp*, vol. 23, pp. 217–222, 2014.

[7] A. D. Udai, S. K. Saha, and A. Dayal, "Overlaid orthogonal force oscillations for robot assisted localization and assembly," *ISME J. Mech. Des.*, vol. 2, no. 1, pp. 09–25, 2019.

[8] R. A. Boby and A. Klimchik, "Combination of geometric and parametric approaches for kinematic identification of an industrial robot," *Robotics and Computer-Integrated Manufacturing*, vol. 71, p. 102142, 2021.

[9] R. A. Boby, "Kinematic identification of industrial robot using end-effector mounted monocular camera bypassing measurement of 3d pose," *IEEE/ASME Transactions on Mechatronics*, 2021.

[10] M. W. Abdullah, H. Roth, M. Weyrich, and J. Wahrburg, "An approach for peg-in-hole assembling using intuitive search algorithm based on human behavior and carried by sensors guided industrial robot," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 1476–1481, 2015.

[11] Y. Zheng, X. Zhang, Y. Chen, and Y. Huang, "Peg-in-hole assembly based on hybrid vision/force guidance and dual-arm coordination," in *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2017, pp. 418–423.

[12] S. Gupta and Y. J. Singh, "Object detection using shape features," in *2014 IEEE International Conference on Computational Intelligence and Computing Research*. IEEE, 2014, pp. 1–4.

[13] T. Gevers and A. W. Smeulders, "Color-based object recognition," *Pattern recognition*, vol. 32, no. 3, pp. 453–464, 1999.

[14] A. P. James, "Edge detection for pattern recognition: a survey," *International Journal of Applied Pattern Recognition*, vol. 3, no. 1, pp. 1–21, 2016.

[15] Q. Bai, S. Li, J. Yang, Q. Song, Z. Li, and X. Zhang, "Object detection recognition and robot grasping based on machine learning: A survey," *IEEE Access*, vol. 8, pp. 181 855–181 879, 2020.

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

[17] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[18] M. Roy, R. A. Boby, S. Chaudhary, S. Chaudhury, S. D. Roy, and S. Saha, "Pose estimation of texture-less cylindrical objects in bin picking using sensor fusion," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 2279–2284.

[19] P. Tiwan, R. A. Boby, S. D. Roy, S. Chaudhury, and S. K. Saha, "Cylindrical pellet pose estimation in clutter using a single robot mounted camera," in *Proc. Advances in Robotics (AIR)*. ACM New York, NY, USA,, 2013, pp. 1–6.

[20] J. Han, D. Zhang, G. Cheng, N. Liu, and D. Xu, "Advanced deep-learning techniques for salient and category-specific object detection: a survey," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 84–100, 2018.

[21] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.

[22] G. Jocher, A. Stoken, J. Borovec, NanoCode012, A. Chaurasia, TaoXie, L. Changyu, A. V, Laughing, tkianai, yxNONG, A. Hogan, lorenzomammana, AlexWang1900, J. Hajek, L. Diaconu, Marc, Y. Kwon, oleg, wanghaoyang0106, Y. Defretin, A. Lohia, ml5ah, B. Milanko, B. Fineran, D. Khromov, D. Yiwei, Doug, Durgesh, and F. Ingham, "ultralytics/yolov5: v5.0 - YOLOv5-P6 1280 models, AWS, Supervise.ly and YouTube integrations," Apr. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.4679653

[23] L. Biewald, "Experiment tracking with weights and biases," 2020, software available from wandb.com. [Online]. Available: https://www.wandb.com/

[24] Stanford Artificial Intelligence Laboratory et al., "Robotic operating system." [Online]. Available: https://www.ros.org

[25] C. Hennersperger, B. Fuerst, S. Virga, O. Zettinig, B. Frisch, T. Neff, and N. Navab, "Towards mri-based autonomous robotic us acquisitions: a first feasibility study," *IEEE transactions on medical imaging*, vol. 36, no. 2, pp. 538–548, 2017.

[26] OpenCV, "Opencv: Open source computer vision library," 2016, http://www.opencv.org/.

[27] R. A. Boby, "Hand-eye calibration using a single image and robotic picking up using images lacking in contrast," in *2020 International Conference Nonlinearity, Information and Robotics (NIR)*. IEEE, 2020, pp. 1–6.

View publication stats