Mechatronics and Automation Technology
J.-Y. Xu (Ed.)

doi:10.3233/ATDE231136

Performance Enhancement with Denavit-Hartenberg (D-H) Algorithm and Faster-RCNN for a 5 DOF Sorting Robot

Oluwaseun K. AJAYI[a,1], Timilehin A. AROJO[a], Ayodeji S. OGUNNAIKE[a], Abiola J. ADEYI[b] and Oladeji. O. OLANRELE[c]
[a] *Department of Mechanical Engineering, Obafemi Awolowo University, Ile-Ife Nigeria*
[b] *Department of Mechanical Engineering, Ladoke Akintola University of Technology, Ogbomoso, Nigeria*
[c] *Department of Mechatronics Engineering, First Technical University, Ibadan, Nigeria*

Abstract. This work demonstrates the fruit sorting improvement system architecture using the Faster-RCNN algorithm to identify fruits of varying size, colour selection, and segregation. The requirement is for a fruit sorting mechanism with accurate and precise detection with accurate placement capabilities. For this purpose, a 5 DOF robotic arm was built in the laboratory. For the control, an algorithm was developed using Denavit-Hartenberg (D-H) kinematic analysis model. The Image processing was implemented in OpenCV by adopting the Faster-RCNN image processing using NumPy and panda packages. Color clustering was employed to improve fruit detection, and segmentation techniques were used to focus more on the fruit image. The algorithm used was based on baseline architecture within relevant architecture, and with the results that are derived, it shows a great improvement from typical fruit detection systems using non-deep learning techniques, especially for position identification and the edge-to-edge estimation of the fruits. However, it was discovered that the time to detection was the same for all colours and weights, while the sorting duration varies with weight.

Keyword. Denavit-Hartenberg; Fruit Detection; Faster R-CNN; Colour Segmentation.

1. Introduction

Sorting is a common phenomenon that has found applications in agricultural product separations, and industrial and production activities. Automated sorting is preferable and is achievable through a set of algorithms built into simple unit self-controlled systems such as robots[1], conveyors[2], or more complex structures such as[3-4]. Sorting robots have been employed over the years for various ranges of products varying from food items[5-6], plastic products[7], raw materials for industry, and finished products[8].

Position, velocity, and acceleration are used to define the kinematics of robots in order to ascertain the system behaviour in terms of its relationship to orientation and

[1] Corresponding author, Ajayi Oluwaseun Kayode, Department of Mechanical Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria; Email: okajayi@oauife.edu.ng.

configuration references. The forward kinematics are used to define the coordinates and orientation of the gripper through the manipulator, while the inverse kinematics defines the links configurations. The above is the first step in developing a robot. This step is finalized by deriving a mathematical model of the kinematics which incorporates all axes and orientations.

Robotic manipulation has given more insight into the popularity of forward and inverse kinematics. Various methods have been developed for the kinematic analysis; two links[9-11], three links[12-15], four links[16-18], and five links[19-20], otherwise known as Degrees of Freedom (DOF). This is important in order to determine the appropriate lengths for the links and the reaction at the joints[21-22], as well as the relative movements corresponding to the total length of the robotic application[23].

This analysis results in a set of matrix forms[24-25] which are solved to determine the dimensions and the inputs for the algorithm[26] that is implemented into the microcontroller for the robotic control. In recent times, researchers have adopted various means to achieve robust and accurate detection of images and objects, however, this work was developed to adopt a Convolutionary Neural Network known as Faster R-CNN, an algorithm with a wider array of detection for a fruit sorting robot. The accuracy and promptness of detection would also be discussed.

2. Methodology

2.1. Robot Development and Kinematics Analysis

Quite a few similar robots have been developed prior to this work, therefore only a few areas of modifications made would be discussed. A pick and place robot with five axes representing five DOF comprising three major axes corresponding to the base, shoulder, and elbow; two minor axes representing the gripper pitch and the gripper spin was designed. In addition to these characteristics, there are three rotary joints in the system. Figure 1 shows the link coordinate of the axes and joints which describes the graphical basis of the mathematical model of the robot.

The Denavit-Hartenberg (D-H) analysis was used to describe the forward inverse kinematics. The direct transform of the n link defined in equation 1 was used thus and the D-H parameters used were those presented in table 1.

$$R_{T_H} = H_1 \times H_2 \times H_3 \times H_n \tag{1}$$

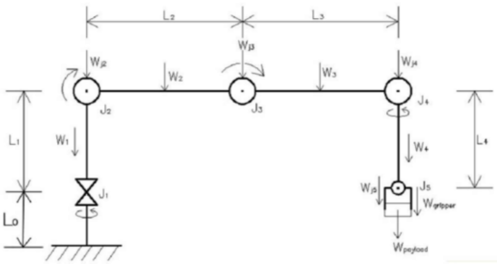

Figure 1. The free body diagram of the robot

Table 1 .Denavit-Hartenberg Parameters of the 5 DOF robotic arm

θn	αn	r	d
θ1	90	0	d1
θ2	0	a2	0
θ3	0	a3	0
θ4	90	a4	0
θ5	0	0	d5

2.2. The Inverse Kinematics

The orientation and coordinates which compares the manipulation matrix for the inverse kinematics was reduced to θ_1, θ_2, θ_3, θ_4, θ_5 with the general transformation matrix given as;

$$\begin{bmatrix} C\theta_1.C\theta_{234}.C\theta_5 + S\theta_1.S\theta_5 & -C\theta_1.C\theta_{234}.S\theta_5 + S\theta_1.C\theta_5 & -C\theta_1.S_{234} & C\theta_1(-d5.S\theta_{234} + a3.C\theta_{234} + a3.C\theta_2) \\ S\theta_1.C\theta_{234}.C\theta_5 - S\theta_1.S\theta_5 & -S\theta_1.C\theta_{234}.S\theta_5 + C\theta_1.C\theta_5 & -S\theta_1.S_{234} & S\theta_1(-d5.S\theta_{234} + a3.C\theta_{234} + a3.C\theta_2) \\ -S\theta_{234}.C5 & -S\theta_{234}.\theta_5 & C\theta_{234} & d1 - a2.S\theta_2 - a3.S\theta_{23} - d5.C\theta_{234} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The homogeneous transformation matrix was expressed and developed, such that the final end effector position can be determined using the mathematical model in 2.

2.3. Motor Torque Selection

In the determination of torque, the gravitational power is based on the gravitational attraction of the joints in relation to the focal point of the earth. Rules corresponding to a cluster of vertical and horizontal distances were set for the microcontroller; the more rules, the smoother the operation. The setting was to determine the position (μ) of objects to the radius of gyration of the robot as described below in equation 3.

$$\begin{cases} 0 \; if \; x \leq \alpha \\ \mu = \quad (x-\alpha)/(\beta-\alpha) \; if \; \alpha \leq x \leq \beta \\ 1 \; if \; x \geq \beta \end{cases} \quad (3)$$

where α and β is the upper and lower boundary indicating "Near" and "Far" distance between the target and the gripper. The proposed picker uses a two-point mode to hold the fruit, though limited to the hand, it required lesser points of control.

2.4. Color Detection for Identification

To make the identification easier, color was a distinct feature that was decided in the design to be used. There are 16,777,216 distinct color possibilities, and therefore, it can be a good feature to use in the easy identification of fruits [27]. In order to achieve accuracy of detection, OpenCV was adopted using a combination of colour detection and object recognition in the deep learning feature NumPy and pandas in python programming. For the camera module, the video was broken down frame by frame for image definition, then colors of the fruits to be detected were inputted as training data points. However, for this, a multi-approach was used where specifically defined features are not needed, but certain features, other than color, it was defined.

The introduction of the Faster R-CNN algorithm allows for real fruit images to be used to train the algorithm to understand the hexadecimal codes for the fruits in a stationary position and in a well-lit environment taking various positions and orientations at 480-pixel. This procedure is described as segmentation which was performed to decide which pixels to segregate each item. Though, the camera module produces a video stream (the different frames within the video) were evaluated by looking at the proportion of correctly classified pixels. There are four categories of detection in the algorithm; True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). TP refers to pixels detected which belong to the object, while TN are the pixels correctly detected which belong to the background (not part of the fruit). FP are those pixels which detected mistakenly which do not belong to the body of the fruit while FN are the pixels not detected and are part of the background. This procedure led to background subtraction at thresholds between 10 and 60.

Detection algorithm is described here. The Receiver operating characteristics curve (ROC) as presented in figure 2, shows TPR and TNR at various thresholds for the background subtraction for detection. This shows the integrity of the developed algorithm. The mean average precision (mAP) was determined using the 11-point interpolation with a intercession over union ratio (IoU) threshold of 0.5. Also, the precision of the algorithm depends on both negative and positive samples. The precision was calculated as 73.5% of the test samples within the computer vision, therefore confirming the integrity of the developed algorithm.

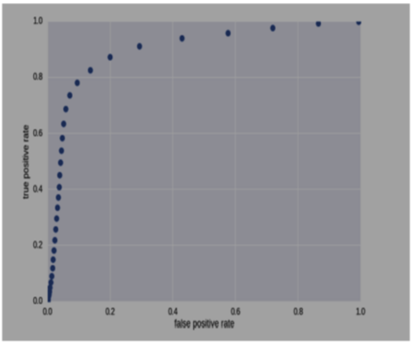

Figure 2. ROC Curve for the background subtraction over the fruit images

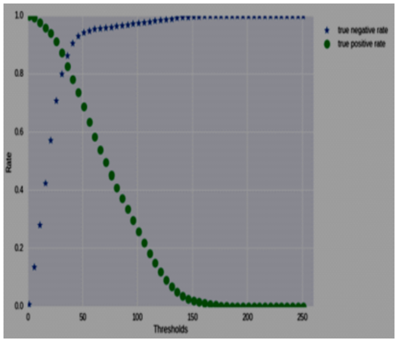

Figure 3. TPR and TNR at various background subtraction thresholds

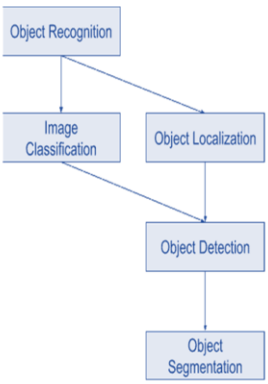

Figure 4. Breakdown of Fruit Detection using CNN

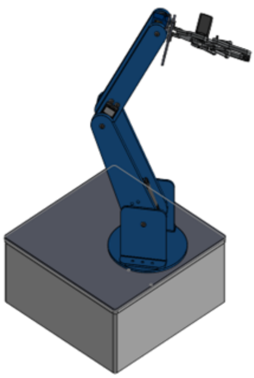

Figure 5. Robot CAD model

Figure 6. Developed Robot

2.5. Robot Development

The CAD models of the various parts were produced using the laser cutter and assembled with the motors and control unit. The grabber was made with plastic forks with bevel ends which to ensure effect holding due to the peculiarity of the fruits. However, this end effector can be changed to the specific operation required based on the physical properties of the items to be picked. The robot control was contained in the base which serves as the reference for the 360^0 - position rotation along the x-axis. The camera was placed at an angle positioned to view objects to be picked so that once a fruit is placed in proximity, the pigment sensed by the camera informs the action taken by the robot. Also, the proximity determines the order of rotation of the motors to give produce a linear movement (y–axis) by the arms.

3. Results

The results of the detection algorithm (figure 2 and figure 3), robot development (figures 4, figure 5 and figure 6) and the testing results (table 2 and table 3) of the developed pick and place robot are as presented.

In this section, the time taken to complete the task of identifying the fruit and sorting the fruits at different weight levels were measured. The fruits used for the testing were tomato, red ball pepper, and scotch bonnet. The tomato weight varied between 150-250g, red ball pepper was 113-142g, and 100g for the scotch bonnet. The time measured was broken into two main action period; detection and sorting. Furthermore, Sorting was classified into unripe and ripe in order to consider performance variability in terms of ripeness.

Table 2. Fruit Sorting Time Periods for Tomato, Red ball pepper and Scotch bonnet

Task	Tomato			Red ball pepper			Scotch bonnet		
	1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd
Detection (sec)	0.7	0.8	0.75	0.7	0.8	0.75	0.8	0.75	0.7
Sorting for unripe (sec)	18	19	17.5	17	18	17	16.5	16	17
Sorting for ripe (sec)	18	18	19	17.5	18	18	17	18	17

Table 3. Average sorting time for the vegetables

Task	Tomato	Red ball pepper	Scotch bonnet
Detection (sec)	0.75	0.77	0.75
Sorting for unripe (sec)	18.2	17.3	16.5
Sorting for ripe (sec)	18.3	17.8	17.3

The time period range is wide due to different variables such as different lighting, positioning of the fruits, and processing performance at the time of the task. Therefore, to prevent variable performance, it is important that the environment where the fruit sorting will be done should be kept stable.

The detection rate was 83% and the precision was pinned at 73.4%. Table 3 shows the breakdown of these data points across three different periods for tomatoes (150-250g). The detection time remained fairly constant throughout for the three fruits at 0.75, 0.77, and 0.75 respectively. The average detection time was 0.75 seconds. In terms of average sorting time for unripe and ripe is 18.2 seconds and 18.3 seconds respectively.

There is a difference in sorting performance mostly due to detection and gripping. Also, there was no major performance difference due to weight within the 150 to 250g weight level. For red ball pepper, table 3 shows that the average detection time for red ball pepper (113-142g) was 0.77 seconds, and the average sorting time for unripe and ripe is 17.3 seconds and 17.8 seconds respectively. Table 3 also shows the breakdown of these data points across three different periods for Scotch bonnet pepper (100g). The average detection time was 0.75 seconds.

In terms of average sorting time for unripe and ripe is 16.5 seconds and 17.3 seconds respectively. Across all these fruits, the time taken for the arm to get back to rest position at each sorting task is 8 seconds because it is not dependent on the fruit-induced variables (detection, gripping). Upon consideration, the sorting time performance was reduced across all three fruits concerning weight. Therefore, the lower the weight, the faster the sorting time for both unripe and ripe fruits. Across all these fruits, the time taken for the arm to get back to rest position at each sorting task is 8 seconds because it is not dependent on the fruit-induced variables (detection, gripping).

To prevent variable performance, it is important that the environment where the fruit sorting will be done should be stable. However, the algorithm can be further improved to cover such edge cases of poor or variable lighting, and improve concurrency in terms of machine learning processing. To prevent variable performance, it is important that the environment where the fruit sorting will be done should be stable. However, the algorithm can be further improved to cover such edge cases of poor or variable lighting, and improve concurrency in terms of machine learning processing. With the results derived, it shows a great improvement from typical fruit detection systems using non-deep learning techniques, especially for position identification and the edge-to-edge estimation of the fruits.

4. Conclusions

The project's main goal was to make an efficient algorithm model to detect fruit to be picked up using the sorting module designed by another researcher within the same team within a set time period. Therefore, the accuracy of the Faster R-CNN algorithm using TensorFlow was 76.4% with a 73.5% precision value with is better than a couple of baseline architectures stated within the evaluated literature under outer environmental conditions. The use of K-means clustering help with managing the colour segmentation process properly via input moderation.

However, an automatic process of selecting input parameters were used, and for this project, 10 was the efficient number of clusters. In addition, under optimum lighting, the fruit sorting process time duration was different, and reduced with lower weight. For unripe and ripe tomato, 18.95 and 19.05 seconds respectively. For red ball pepper, 18.07 and 18.57 seconds respectively. For Scotch bonnet pepper, 17.25 and 18.05 respectively.

To greatly improve the existing architecture, Mask R-CNN and ResNet can be used and applied to the computer vision and fruit detection system to increase the accuracy and precision value. Also, the computer vision algorithm needs to place a lot of focus on lighting conditions and ensure the robot can perform such a task regardless of the optimum condition.

References

[1] Alatise M. B. and Hancke G. P. A review on challenges of autonomous mobile robot and sensor fusion methods[J]. IEEE Access, 2020, 8:39830-39846, doi: 10.1109/ACCESS.2020.2975643.

[2] Simran E., Sangeeta K. and Aditi D. Development of automatic sorting conveyor belt using PLC[J]. International Journal of Mechanical Engineering and Technology, 2019, 10(8): 109-118. https://ssrn.com/abstract=3445937

[3] Tan Z. Y., Li H. L., He X. T. Optimizing parcel sorting process of vertical sorting system in e-commerce warehouse[J]. Advanced Engineering Informatics, 2021, Volume 48, https://doi.org/10.1016/j.aei.2021.101279.

[4] Zhang Z, Lu Y Z and Lu R F. Development and evaluation of an apple infield grading and sorting system[J]. Postharvest Biology and Technology, 2021, Volume 180, https://doi.org/10.1016/j.postharvbio.2021.111588.

[5] Kumar I., Rawat J., Mohd N., Husain S. Opportunities of artificial intelligence and machine learning in the food industry[J]. Journal of Food Quality, 2021, vol. 2021. https://doi.org/10.1155/2021/4535567

[6] Mei Z., Li D. and Chen T. Design of YUMI collaborative robot sorting system based on machine vision[C]. International Conference on Intelligent Computing, Automation and Systems (ICICAS), Chongqing, China, 2021, pp. 290-293, doi: 10.1109/ICICAS53977.2021.00066.

[7] Xiao, W., Yang, J., Fang, H. et al. Development of an automatic sorting robot for construction and demolition waste[J]. Clean Techn Environ Policy, 2020, 22:1829–1841. https://doi.org/10.1007/s10098-020-01922-y

[8] Krishnakumar S., Sneha K. and Reethika A. Review on sensor based colour sorting robot for candy manufacturing[C]. IOP Conf. Ser.: Mater. Sci. Eng. 2021. DOI 10.1088/1757-899X/1084/1/012094

[9] Yang B., Tan U. -X., McMillan A. B., Gullapalli R. and Desai J. P. Design and control of a 1-DOF MRI-compatible pneumatically actuated robot with long transmission lines[J]. IEEE/ASME Transactions on Mechatronics, 2011, 16(6):1040-1048, DOI: 10.1109/TMECH.2010.2071393.

[10] Zhang, Y., Kong, X., Yue, C. et al. Dynamic analysis of 1-dof and 2-dof nonlinear energy sink with geometrically nonlinear damping and combined stiffness[J]. Nonlinear Dyn, 2021, 105: 167–190. https://doi.org/10.1007/s11071-021-06615-9

[11] Ammar A., Bijan S., Mohammadali G., Tilok K. D., Yanling T. and Dawei Z. A fuzzy disturbance observer based control approach for a novel 1-DOF micropositioning mechanism[J]. Mechatronics, 2020, Volume 65. https://doi.org/10.1016/j.mechatronics.2019.102317.

[12] Tao K., Yi H., Tang L., Wu J., Wang P., Wang N., Hu L., Fu Y., Miao J. and Chang H. Piezoelectric ZnO thin films for 2DOF MEMS vibrational energy harvesting[J]. Surface and Coatings Technology, 2019, 359:289-295. https://doi.org/10.1016/j.surfcoat.2018.11.102.

[13] Wu J., Yu G., Ga Y. and Wang L. Mechatronics modeling and vibration analysis of a 2-DOF parallel manipulator in a 5-DOF hybrid machine tool[J]. Mechanism and Machine Theory, 2017, 121(2018):430-445. https://doi.org/10.1016/j.mechmachtheory.2017.10.023.

[14] Wang Y., Chen Z., Sun M., Sun Q. and Piao M. On sign-projected gradient flow-optimized extended-state observer design for a class of systems with uncertain control gain[J]. IEEE Transactions on Industrial Electronics, 2023, 70(1):773-782 DOI:10.1109/TIE.2022.3150096.

[15] Ramin G. and Clément G. Kinematic analysis of a new 2-DOF parallel wrist with a large singularity-free rotational workspace[J]. Mechanism and Machine Theory, 2022, Volume 175. https://doi.org/10.1016/j.mechmachtheory.2022.104942.

[16] Zou, Q., Zhang, D., Zhang, S. et al. Kinematic and dynamic analysis of a 3-DOF parallel mechanism[J]. Int J Mech Mater Des, 2021, 17: 587–599. https://doi.org/10.1007/s10999-021-09548-

[17] Kawamura S. and Deng M. Recent developments on modeling for a 3-DOF micro-hand based on AI methods[J]. Micromachines, 2020, 11(9):792. https://doi.org/10.3390/mi11090792

[18] Véronneau C., Denis J., Lebel L., Denninger M., Blanchard V., Girard A. and Plante J. Multifunctional Remotely Actuated 3-DOF Supernumerary Robotic Arm Based on Magnetorheological Clutches and Hydrostatic Transmission Lines[J]. IEEE Robotics and Automation Letters, 2020, 5(2): 2546-2553. doi: 10.1109/LRA.2020.2967327.

[19] Dewi T., Nurmaini S., Risma P., Oktarina Y. and Roriz M. Inverse kinematic analysis of 4 DOF pick and place arm robot manipulator using fuzzy logic controller[J]. International Journal of Electrical and Computer Engineering, 2020, 10(2): 1376-1386. DOI: 10.11591/ijece.v10i2.

[20] Chen H., Fang Y. and Sun N. An adaptive tracking control method with swing suppression for 4-DOF tower crane systems[J]. Mechanical Systems and Signal Processing, 2019, 123:426-442, https://doi.org/10.1016/j.ymssp.2018.11.018.

[21] Nguyen H., Le P. and Kang H. A new calibration method for enhancing robot position accuracy by combining a robot model–based identification approach and an artificial neural network–based error compensation technique[J]. Advances in Mechanical Engineering, 2019, 11(1): 1–11.

[22] Wu J., Zhang D., Liu J. and Han X. A Moment Approach to Positioning Accuracy Reliability Analysis for Industrial Robots[J]. IEEE Transactions on Reliability, 2020, 69(2): 699-714, doi: 10.1109/TR.2019.2919540.

[23] Ha S., Coros S., Alspach A., Bern J. M., Kim J. and Yamane K. Computational design of robotic devices from high-level motion specifications[J]. IEEE Transactions on Robotics, 2018, 34(5): 1240-1251. DOI: 10.1109/TRO.2018.2830419.

[24] Luo X., Xie F., Liu X, and Xie Z. Kinematic calibration of a 5-axis parallel machining robot based on dimensionless error mapping matrix[J]. Robotics and Computer-Integrated Manufacturing, 2021, Volume 70, https://doi.org/10.1016/j.rcim.2021.102115.

[25] Schonstein C. Kinematic control functions for a serial robot structure based on the time derivative Jacobian matrix[J]. Applied Mathematics, Mechanics, and Engineering, 2018, 61(2).

[26] Patle B. K., Parhi D. R. K., Jagadeesh A. and Sunil K. K. Matrix-binary codes based genetic algorithm for path planning of mobile robot[J]. Computers & Electrical Engineering, 2018, 67: 708-728, https://doi.org/10.1016/j.compeleceng.2017.12.011.

[27] Ahmad M., Abbas Z., Nasir A., Waseem M., Azim A. and Abba A. Pliable Algorithm RGB image Convert in Gray image Using Transformation Equation[J]. Computer Science and Software Engineering, 2019, 8(12): 2409-4285.