International Conference on Mechanical and Industrial Engineering
ICMIE2022
November 15-17, 2022 Tripoli-Libya

Kinect Based Strawberries Harvesting Robot using YOLO v2 Algorithm

Zakariya Abousabie, Azeddien Kinsheel
Department of Mechanical and Industrial Engineering University of Tripoli, Libya
zakryaznati@gmail.com, a.kinsheel@uot.edu.ly

Abstract— Automatic robotic fruit harvesting requires a robust computer vision system to detect fruits autonomously in the field. In this paper, a method for automatic detection and harvesting strawberries using a robot arm is proposed. The proposed method is based on YOLO v2 detector to detect the position of strawberry. then cutting points are estimated via a geometric approach based on strawberry detection bounding box. The three dimensional location of the cutting points is obtained using a Kinect v2 sensor. The coordinate frame transformation used to relate the cutting points to the robot arm is also developed and calibrated. In addition, this work proposes a low cost yet efficient gripper to pick the fruit.

Keywords—Harvesting Robot, Deep Learning, YOLO v2, Kinect v2, Robot

I. INTRODUCTION

Strawberries are one of the most famous and widely grown fruits in the world, they can be planted outdoors or in controlled environments such as polytunnels and greenhouses [1]. However, harvesting is the most laborious and time consuming step in strawberry production. Harvesting labor costs account for more than 75% of the total production costs, and this proportion is annually increasing [2]. In general manual harvesting of fruits and delicate crops is a laborious, tedious, and time-consuming task in agricultural [3]. Automatic harvesting has advantages over manual harvesting, e.g., reduced labor involvement, shorter time in crop management, better control over environmental effects, and higher quality. Over the past two decades, robots have been widely used in agriculture to harvest crops because of these potential benefits [4]. Typically, a harvesting robot consists of three sub-systems: a computer vision system for detecting crops, a robot arm, and an end-effector as a harvesting tool.

However, the development of harvesting robots has major challenges: first, the problem of strawberry detection and localization using computer vision which requires sophisticated algorithms; second strawberries are delicate, soft, and easily damaged, so the end-effector should not touch the fruit surface while harvesting; and finally the environment of harvesting is changing frequently, so the robot should be able to handle it. In general, the detection of picking points requires the detection of the target fruit, followed by the estimation of its location based on differences in the size, shape, color, and texture that distinguish the fruit from its background [5]. Many studies and publications on robot harvesting systems have been published during the last few decades for fruits such as, Apples [6], sweet paper [7], tomato [8], and strawberry [9] [10]. These studies show how challenging the harvesting issue is. The algorithm utilized for fruit detection in images determines the harvesting robot's efficiency. Various detection approaches based on one or more features such as color, shape, texture...etc. have been developed over the last two decades. In [11], a color threshold is used for each pixel to detect if it belongs to the fruit, this approach showed 95% of correctly detect apples. Generally, the color-based detection works very well with red apples, however, it doesn't provide satisfactory results if there is a slight change in apple color or if they were green for example. To solve this problem, researchers used different approaches, such as infrared spectra [12]. In [13], a shape based approach is used to detect fruit, systems based on such an approach work very quickly, but it is applicable only to circle shapes (spherical fruits). In [14], apples were detected based on image texture in combination with color analysis, this approach showed 90% of correctly detected apples. However, detecting fruits by texture required close-up images with good resolution in order to work with reliable results, also the low speed of texture based fruit detection algorithm makes it practically inefficient. Since 2012, various object detection including fruits in images received great development because of the presence of convolutional neural networks, in particular AlexNet [15]. In 2015, a convolutional neural network was proposed as an improvement of AlexNet called VGG16 [16]. In [17], kiwi harvesting robot based on VGG16 was able to detect up to 90% of kiwi fruit. The next improvement in computer vision was R-CNN network family: RCNN [18], FastRCNN [19], FasterRCNN [20], and MaskRCNN [21]. In [22], a Mask RCNN model is used to detect strawberries for harvesting robot the model F1 score exceeded 90%. In 2016, a new object detection algorithm called YOLO (You Only Look Ones) was proposed [23], What makes this algorithm unique is that, in prior models, the neural network was applied to images numerous times, whereas YOLO divides images into regions and then determines the scope of object and its probability in a single run of neural network. The YOLO algorithm is faster than other models, and it has been used in robots harvesting. In [24], YOLO V3 is used to detect apple Lesions, reaching 95.57% accuracy. A YOLO model is utilized in this study to detect strawberries.

The major challenge after the detection is to harvest without causing any damage to the fruit. Many harvesting tool designs have emerged as a result of research, including suction, swallow, and scissor-like cutters. In [25], a suction-

based gripper is used. This sort of gripper does not physically touch the fruit during the cutting process, however, contact happens while inhaling the fruit after cutting to place it in a container, causing harm, particularly to soft fruits, plus using a pump equipment increase the weight and complexity of the robot. [26] uses a scissor-like gripper consisting of a cutting blade and a force-controlled gripper it works well for soft fruits, but size constraints restrict it to regular shapes only. A gripper with a clamping and cutting motion is created to address this issue (see Fig.1). The remarkable feature of the gripper is that it performs the two operations with just one actuator, drawing inspiration from farmer skills that involve grasping the peduncle with the fingers and then performing a cutting with nails or a tool.

In this paper, a robotic strawberry harvesting robot in controlled environment is presented. The aim of this work is the following: (i) modifying and simplifying YOLO v2 neural network to use it as a strawberry detector, (ii) development of a geometric method to detect strawberry cutting point based on detector bounding box.

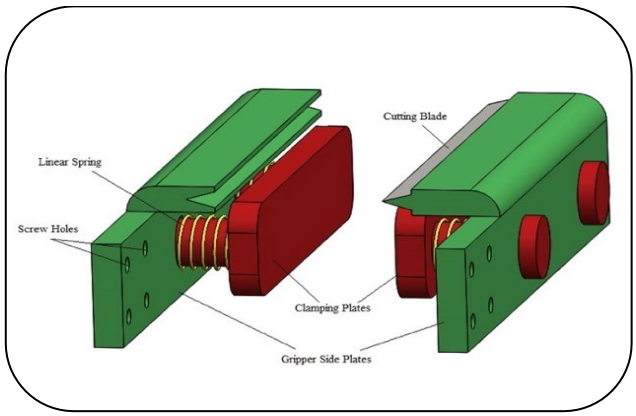

Fig. 1. 3D view of harvesting gripper

II. Materials and Methods

A. The Robot

The robot used in this work is Mitsubishi MELFA RV-2AJ with 5 DOF shown in Fig. 2. The robot offers good performers as it is small, compact, and powerful, it operates with AC servomotors with a maximum speed of 2100 mm/s and a repeatability of ±0.02 mm. The robot's maximum payload is 2Kg, and its working envelope is 220mm and 410mm inner and outer radius respectively.

B. Vision System

The vision system is a Microsoft Kinect v2 sensor, which's a low cost and available compared to other 3D sensors or cameras. The Kinect is composed of RGB camera with a resolution of 1920x1080, an infrared emitter, and an infrared camera with a resolution of 512x424 as shown in Fig. 3. The IR emitter and camera work as depth estimation sensor working with Tof (time of flight) principle. Simply this principle is based on the knowledge of the speed of light, then the distance can be estimated, since its proportional to the time needed for light to travel from emitter to target .Kinect v2 uses an indirect Tof system which depends on a light modulation frequency and a phase shift between emitted and received signal to measure depth, as shown in Equation (1) [29].

Fig. 2. Mitsubishi RV-2AJ Robot [28].

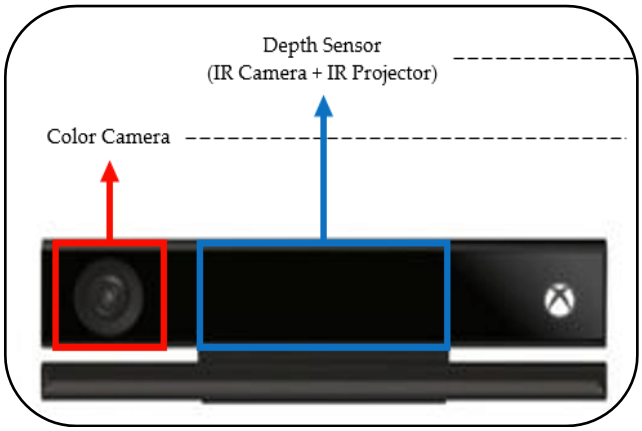

Fig. 3. Kinect V2 Sensor [30].

$$d = \frac{\Delta\varphi}{4\pi f} c \quad (1)$$

Where $\Delta\varphi$ = phase shift
f = modulation frequency
c = speed of light

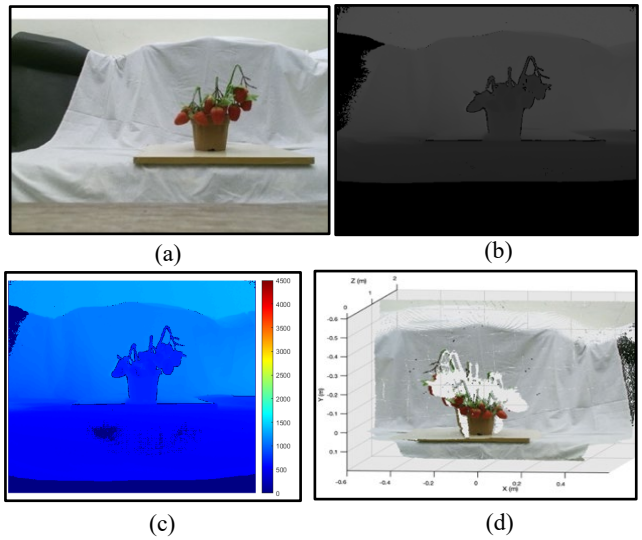

Fig. 4. (a) Kinect v2 RGB Image, (b) Depth Image, (c) Colored Depth Image (mm) , (d) Real World Coordinates Point Cloud

Kinect v2 sensor provides measurements of the distance between the sensor and the entire scene, with the measured distance corresponding to each pixel in the IR image. This data is referred to as a depth-map (see Fig. 4(b), (c)) which can be mapped into its corresponding RGB image , This mapping between two images leads to sufficient depth accuracy up to 80% [31], (see Fig. 4 (a)), Then using intrinsic and extrinsic camera parameters to create what is known as a point Cloud (see Fig. 4 (d)) which's a reconstruction of the real scene in 3D coordinates with respect to the camera.

C. *Strawberry detector*

Detecting strawberries' 2D positions in images is the first stage. RGB images are obtained from the Kinect, then a YOLO v2 [32] detector is utilized to perform the detection. However, before beginning the detection process, YOLO has to be modified, as known the original model was developed using data sets like COCO or VOC, which have 80 to 20 classes (e.g., person, door, cat, car,… etc.) that require a powerful GPU for processing and computation. Which's make it unsuitable for fruit detection where the target is less than 10 classes, in this case only strawberries. Therefore modification and simplification of YOLO structure is needed to make it suitable for fruit detection. In addition, the simplification will enhance the real-time performance of the detector and decreasing the required computational power, therefore single computer's CPU is sufficient . Fig. 5 illustrates the modification, which is a reconstruction of the base network into a straightforward neural network with fewer layers than the previous one in the original model.

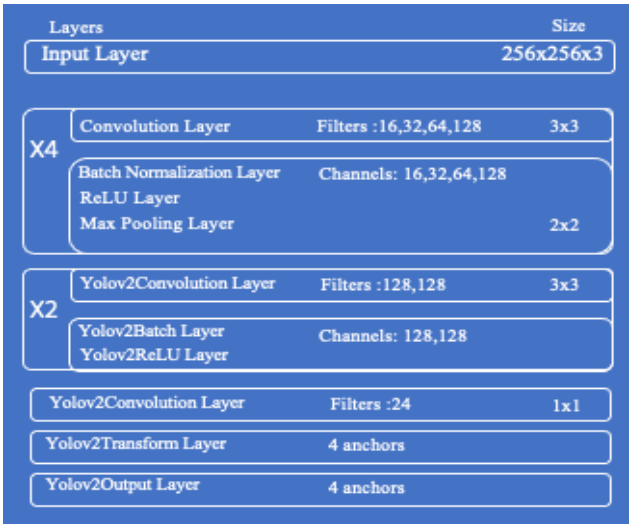

Fig. 5. YOLO v2 simplified neural network

D. *Cutting point detection*

As a result of the detection, a bounding box enclosing the strawberries is created, and the cutting point is estimated using the bounding box dimensions. In [33], the cutting point is estimated using a fixed value of 15 pixels above the bounding box, which is not always accurate. In [27] cutting point location is estimated based on the bounding box dimension, the work of [27] is employed in this study to estimate the cutting point location as shown in Fig. 6.

After getting the cutting point coordinates in pixels, its metric coordinates with respect to the Kinect can be obtained by mapping it to the produced point cloud from the Kinect, For simplicity and reducing constraints also based on the wide jaws of the gripper the fruit orientation will not be concerned and the fruit is assumed to be hanging vertically.

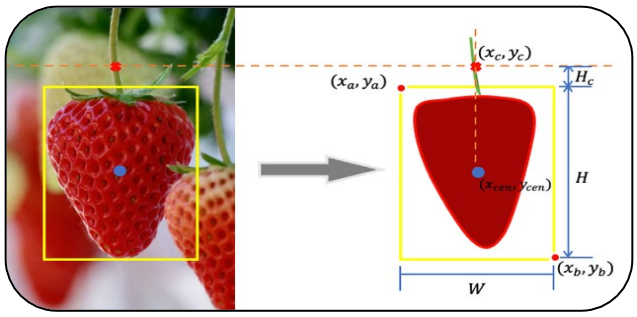

Fig. 6. Cutting Point Pixel Coordinates

The estimated location of the cutting point is shown in the following equations :

$$W = x_b - x_a \quad (2), \quad H = y_b - y_a \quad (3)$$

$$x_{cen} = x_a + \frac{W}{2} \quad (4), \quad y_{cen} = y_a + \frac{H}{2} \quad (5)$$

$$x_c = x_{cen} \quad (6), \quad H_c = y_{cen} - \frac{H}{2} \quad (7)$$

$$y_c = H_c - \frac{H}{4} \quad (8)$$

Where:
$(x_a,\ y_a)$ = coordinates of upper left corner.
$(x_b,\ y_b)$ = coordinates of bottom left corner.
W, H = Height and width of bounding box respectively.
H_c = cutting distance above bounding box.
$(x_{cen},\ y_{cen})$ = coordinates of center point.
$(x_c,\ y_c)$ = coordinates of cutting point.

E. Camera To Robot mapping (registration)

Since the Kinect point cloud data for strawberry cutting points is with respect to camera coordinates, it should be with respect to robot base coordinates in order to harvest the strawberries. To accomplish this, a mapping between the camera and robot coordinates must be established, as shown in Fig. 7.

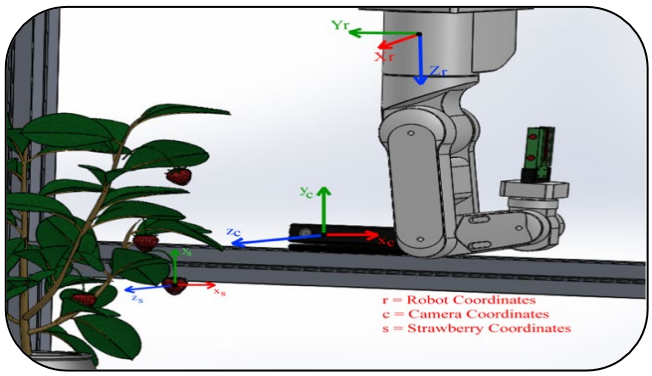

Fig. 7. Robot and Camera Coordinates

In this paper, a linear method is used to establish the transformation matrix between camera and robot. In genera,l any two frames can be mapped if a transformation matrix exists between them. This transformation matrix contains the rotation and the translation vectors between these two frames.

$$F_1 = \begin{bmatrix} R & T \\ 0\ 0\ 0 & 1 \end{bmatrix} F_2 \quad (9)$$

Where
F_1=the coordinates of frame 1
F_2=the coordinates of frame 2
R=the rotation matrix
T=the translation vector

So using the provided cutting point coordinates from the Kinect, the following matrix is utilized to transform those coordinates into cutting points coordinates with respect to the robot base, in order to perform a successful harvesting.

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & D_x \\ r_{21} & r_{22} & r_{23} & D_y \\ r_{31} & r_{32} & r_{33} & D_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (10)$$

Where:

X, Y, Z = coordinates of one point with respect to robot base
x, y, z = coordinates of one point with respect to camera from point cloud
D = elements of translation vector
r = elements of rotation matrix

The matrix has a 12 unknowns rewrite the matrices in equation form:

$$X = r_{11}x + r_{12}y + r_{13}z + D_x \quad (11)$$

$$Y = r_{21}x + r_{22}y + r_{23}z + D_y \quad (12)$$

$$Z = r_{31}x + r_{32}y + r_{33}z + D_z \quad (13)$$

Equations 11, 12, and 13 shows that every point generates 3 equations, since the robot can provide the coordinates of any point in the workspace, and Kinect can provide the coordinates of this corresponding point with respect to its coordinates. Therefore, 4 reference points from the scene are required to generate 12 equations (see Fig. 8), which can then be solved simultaneously to determine the unknowns which's are elements of the transformation matrix. This transform is then applied to find any point with respect to robot base using data points from the Kinect.

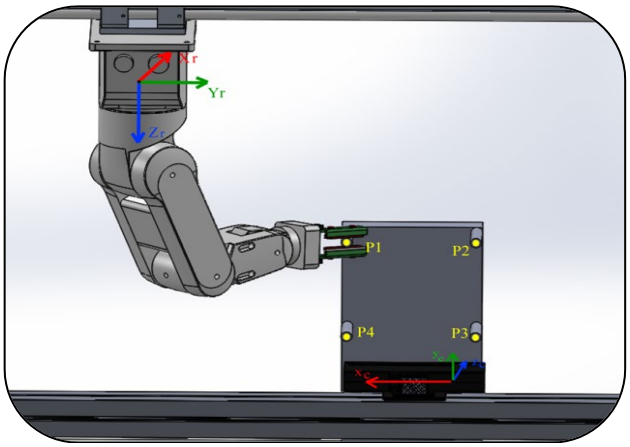

Fig. 8. Transformation Matrix Method for Mapping

III. RESULTS

A. Training and detection results

A round 500 images containing real and plastic strawberries were used to train the detector. Images were collected from Google, and some of them were captured manually with background noise using a camera and cell phone camera, then labeled with bounding boxes for each strawberry. 80% of images were used for training, 10% for validation, and 10% for testing. Stochastic Gradient Descent with Momentum (SGDM) algorithm was used to train data to converge to a solution with a learning rate of 1e-03. 80 Epochs and a mini batch size of 16 also were used as training options. The training was done with intel core i7 –

dual core CPU (2.9GHz) and took above 2 hours to complete training on MATLAB software.

A mean average precision curve and average miss rate are utilized, as shown in Fig. 9, result show mean average precision (mAP) of 78.45% and average miss rate of 39.77% . Detection time takes on average 60ms.

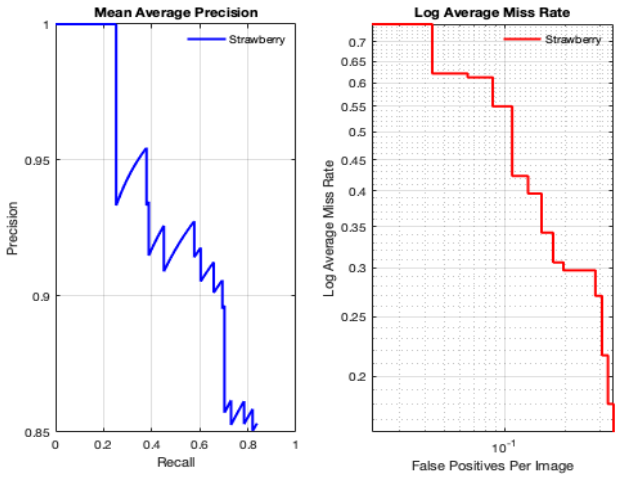

Fig. 9. mean average precision and average miss rate curves

Fig. 10 and 11 show some detection results of strawberries, and Fig. 12 shows the detection and strawberry cutting point estimation.

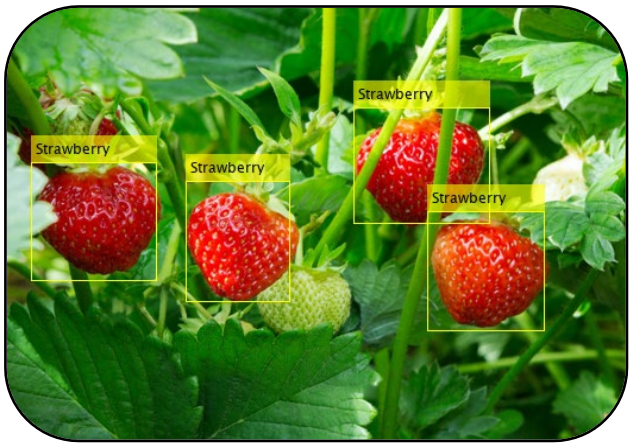

Fig. 10. Strawberries Detection Results

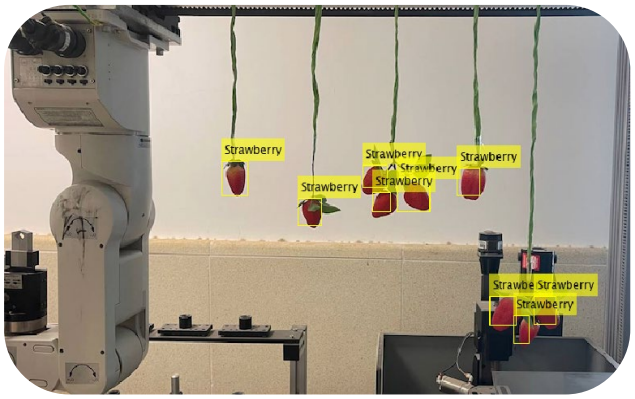

Fig. 11. Strawberries Detection in The Lab

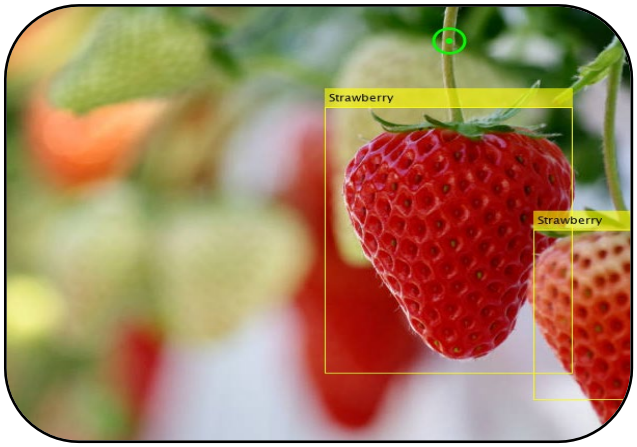

Fig. 12. Estimated Cutting Point (Green Circle)

B. *Harvesting Robot*

Approach point motion planning is performed for the harvesting process, The robot starts from a fixed offset home point from the target strawberry, then moves to an approach point closer to the estimated cutting point, then finally moves towards the cutting point and performs the harvesting as shown in Fig. 12 (a, b, c).

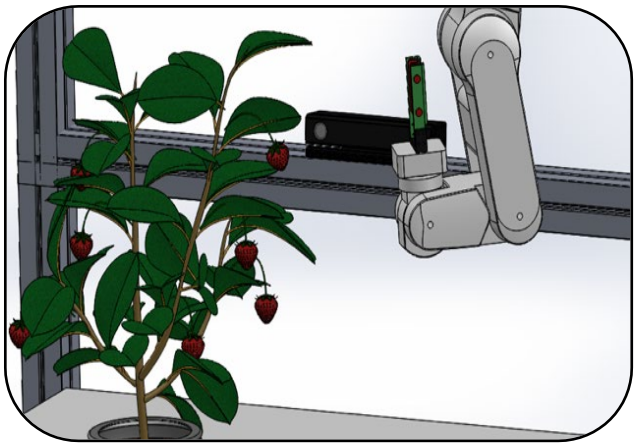

Fig. 13. a. Robot Home Point

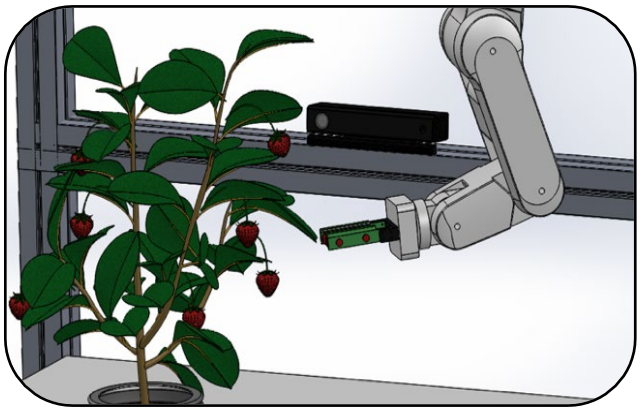

Fig. 13. b. Approach Point

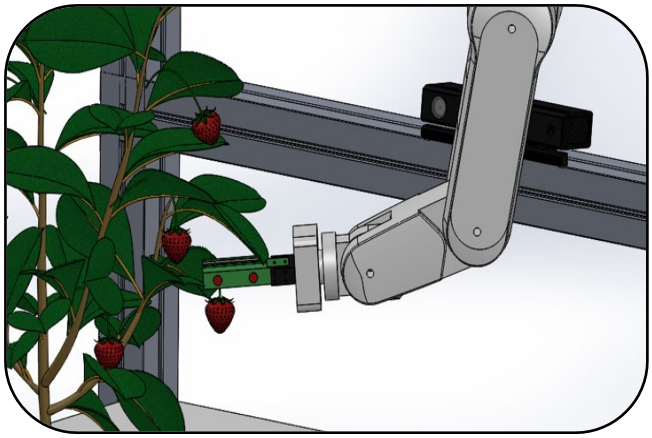

Fig. 13. c. Cutting Point

Table 1 and Table 2 show the results of applying the obtained transformation matrix to two test points in the workspace using Kinect data.

TABLE I. TRANSFORMATION MATRIX POINT 1 TEST RESULTS

Coordinates	Actual (cm)	Estimates (cm)	Error (cm)
X	34.49	34.77	0.28
Y	0.071	0.4	0.33
Z	57.19	57.17	0.02

TABLE II. TRANSFORMATION MATRIX POINT 2 TEST RESULTS

Coordinates	Actual (cm)	Estimates (cm)	Error (cm)
X	27.0	27.17	0.17
Y	0.070	0.2	0.13
Z	71.99	72.08	0.09

IV. CONCLUSION

In this paper, a strawberry harvesting robot was developed using a modified YOLO v2 detector and a Kinect v2 sensor. The cutting points of the fruit are estimated using a geometrical approach based on detection results. Since strawberries are easily damaged, a low-cost and efficient gripper is designed to harvest fruit without causing damage. In order to perform precise harvesting, a frame coordinate transformation is developed and calibrated between the robot and the Kinect using a linear model, which shows a maximum error of around 0.3cm. Although the whole system appears to be expensive, However with the development of research in this field, its costs will be reduced and become more practical.

REFERENCES

[1] G. Y. G. L. e. a. Xiong Y, "An autonomous strawberry - harvesting robot: Design, development, integration, and field evaluation," Journal of Field Robotics, p. 37, 2020.

[2] S. J. X. W. e. a. Linnan J, "A new type of facility strawberry stereoscopic cultivation mode," journal of china agricultural university, 2019.

[3] K. Kapach, E. Barnea, R. Mairon, Y. Edan and O. Ben-Shahar, "Computer vision for fruit harvesting robots-state of the art and challenges ahead," *International Journal of Computational Vision and Robotics* , pp. 4-34, 2012.

[4] C. Bac, J. Hemming, B. Tuijl, R. Barth, E. Wais and E. Henten, "Performance evaluation of a harvesting robot for sweet pepper.," *Field Robot,* 2017.

[5] K. Z. H. L. L. Y. A. D. Z. YANG YU, "Real-time Visual Localization of the Picking Points for a Ridge-planting Strawberry Harvesting Robot," *IEEE Access,* 2020.

[6] T. Y. H. K. T. F. H. A. a. A. I. Yuki Onishi1*, "An automated fruit harvesting robot by using deep learning," *ROBOMECH,* pp. 6-13, 2019.

[7] I. S. C. M. B. U. a. T. P. Christopher Lehnert, "Sweet Pepper Pose Detection and Grasping for Automated Crop Harvesting," in *IEEE*, stockholm, sweden, 2016.

[8] B. L. T. S. K. I. Shinsuke Yasukawa, "Development of a Tomato Harvesting Robot," in *International Conference on Artificial Life and Robotics*, miyazaki,japan, 2017.

[9] C. P. L. G. P. J. F. V. I. Ya Xionga, "Development and field evaluation of a strawberry harvesting robot with a cable-driven gripper," *ELSEVIER,* pp. 393-402, 2019.

[10] K. S. S. Y. K. K. Y. K. J. K. M. K. Shigehiko Hayashi, "Evaluation of a strawberry-harvesting robot in a field test," *ELSEVIER,* pp. 161-171, 2009.

[11] K. J. J. L. Y. L. Y. Z. C. W. Xiangqin Wei, "Automatic method of fruit object extraction under complexagricultural background for vision system of fruit picking robot," *ELSEVIER,* pp. 5685-5689, 2014.

[12] H. I. S. T. B. a. V. A. J. P. Wachs1, "Low and High-Level Visual Feature Based Apple Detection from Multi-modal Images," *Springer Link,* p. 717–735, 2010.

[13] G. E. M. O. R. M. L. D. G. A. Dale Whittaker, "Fruit Location in a Partially Occluded Image," *American Society of Agricultural Engineers,* pp. 591-596, 1987.

[14] J. T. a. J. K. Jun Zhao, "On-tree Fruit Recognition Using Texture Properties and Color Data," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Canada, 2005.

[15] A. Krizhevsky, I. Sutskever and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *the Advances in Neural Information Processing Systems Conference—NIPS*, NV,USA, 2012.

[16] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *International Conference on Learning Representations*, San Diego, CA, USA, 2015.

[17] Z. Liu, J. Wu, L. Fu, Y. Majeed, Y. Feng, R. Li and Y. Cui, "Improved kiwifruit detection using pre-trained VGG16 with RGB and NIR information fusion," *IEEE Access,* vol. 8, pp. 2327-2336, 2020.

[18] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, 2014.

[19] R. Girshick, "Fast R-CNN," in *2015 IEEE International Conference on Computer Vision—ICCV*, Santiago, Chile, 2015.

[20] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN Towards real-time object detection with region proposal networks," *IEEE Trans,* vol. 39, pp. 1137-1149, 2017.

[21] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," in *2017 IEEE International Conference on Computer Vision—ICCV*, Venice, Italy, 2017.

[22] Y. Yu, K. Zhang, L. Yang and D. Zhang, "Fruit detection for strawberry harvesting robot in non-structural environment based on Mask-RCNN," *ELSEVIER, Computers and Electronics in Agriculture,* 2019.

[23] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once Unified, Real-Time Object Detection," in *29th IEEE Conference on Computer Vision and Pattern Recognition—CVPR*, Las vegas, NV, USA, 2016.

[24] Y. Tian, G. Yang, Z. Wang, E. Li and Z. Liang, "Detection of apple lesions in orchards based on deep learning methods of CycleGAN and YOLO-V3-Dense," *Journal od sensors ,* pp. 1-13, 2019.

[25] J. Hemming, C. Bac, B. van Tuijl, R. Barth, J. Bontsema, E. Pekkeriet and E. van Henten, "A robot for harvesting sweet-pepper in greenhouses," in *International Conference of Agricultural Engineering*, Zurich, Switzerlnd, 2014.

[26] G. Tian, J. Zhou and B. Gu, "Slipping detection and control in gripping fruits and vegetables for agricultural robot," *Int J Agric & Bio Eng,* vol. 11, pp. 45-51, 2018.

[27] Z. H. W. Y. J. L. X. T. a. H. H. Tan Zhang, "An Autonomous Fruit and Vegetable Harvester with a Low-Cost Gripper Using a 3D Sensor," *Sensor,* vol. 20, p. 1_15, 2019.

[28] P. R. Joerg Wolf, "Mitsubishi RV-2AJ Industrial Robot Programming and Calibration," The University of Plymouth, Plymouth, England, Nov2005.

[29] H. M. M.-A. M. T. L. P. G. E. Lachat, "FIRST EXPERIENCES WITH KINECT V2 SENSOR FOR CLOSE RANGE 3D MODELLING," *The International Archives of the Photogrammetry,* pp. 93-100, 2015.

[30] J. M. Y.-J. C. a. M. H. Nak-Jun Sung, "Real-Time Augmented Reality Physics Simulator for Education," *MDPI, Applied Sciences,* vol. 9, pp. 1-12, 2019.

[31] M. L. J.-K. K. A. H. Burak Teke , "Real-time and Robust Collaborative Robot Motion Control with Microsoft Kinect® v2," in *IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, Finland, 2018.

[32] A. F. Joseph Redmon, "YOLO9000: Better, Faster, Stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition*, Washington, 2017.

[33] K. S. ,. S. Y. ,. K. K. ,. Y. K. ,. J. K. ,. M. K. Shigehiko Hayashi, "Evaluation of a strawberry-harvesting robot in a field test," *ELSEVIER (bio systems engineering),* vol. 105, pp. 160-171, 2010.