mekatronika – Journal of Intelligent Manufacturing & Mechatronics
VOL. 01, ISSUE 01, 80 – 86
DOI: https://doi.org/10.15282/mekatronika.v1.1148

ORIGINAL ARTICLE

Autonomous Tomato Harvesting Robotic System in Greenhouses: Deep Learning Classification

Toon Ooi Peng[1], Muhammad Aizzat Zakaria[*1], Ahmad Fakhri Ab. Nasir[1], Anwar P.P. Abdul Majeed[1], Tan Chung Young[2], Leonard Ng Chong Yew[2]

[1]Innovative Manufacturing, Mechatronics and Sport (iMAMS) Laboratory, Faculty of Manufacturing Engineering, Universiti Malaysia Pahang
[2]KUKA Robot Automation (Malaysia) Sdn Bhd

ABSTRACT – Solanum lycopersicum or generally known as tomato came from countries of South America and has been growing in many tropical countries and its healthy nutrients in tomato becomes one of the food demand by the locals in Malaysia when their lifestyle shifted to more concern for healthy food. Since export value and production has increased for the past few years, a vast amount of labours considered for the fruit-picking process. Hence, farmers are now preferring to look for automation to replace labour problems and high cost that they are facing. To pick a correct fruit within clusters, a harvesting robot requires guidance so that it can detect a fruit accurately. In this study, a new classification algorithm using deep learning specifically convolution neural network to classify the image is either a tomato or not tomato and next, the image is classified into either a ripe or unripe tomato. Furthermore, there are two classification neural networks which are tomato or not tomato and ripe and unripe tomato. Each network consists of 600 training data and 33 testing data. The accuracies that obtained from network 1 (tomato or not tomato) and network 2 (ripe or unripe tomato) are 76.366% and 98.788% respectively.

ARTICLE HISTORY
Received: 29 December 2018
Revised: 25 January 2019
Accepted: 28 January 2019

KEYWORDS
Convolution Neural Network
deep learning
tomato
harvesting robot
classification

Introduction

Solanum Lycopersicum or known as tomato came from countries of South America and has been growing in many tropical countries such as Malaysia. It is actually a fruit instead of vegetable and it has an important anti-oxidant that helps in the fight against the formation of a cancerous cell and the nutrients can contribute to health benefits for the human body (Bhowmik et al., 2012). Healthy nutrients in tomato become one of the factors that tomato becomes one of the food demand by the locals in Malaysia when their lifestyle shifted to more concern for healthy food (Islam, Arshad, Radam, & Alias, 2012).

The export value from the year 2009 is 910,000USD and progressively increased to 2,882,000USD in the year 2014. This increase of its values show there's an advancement technology such as open system, hydroponic and fertigation has contributed to this improvement. (Rahim et al., 2017). Since export value and production has increased for the past few years, a vast amount of labours considered for the fruit-picking process. Besides, labour cost is one of the challenges in agriculture as it is one of the largest costs which making out about 35% of operational costs in greenhouse processes. Hence, farmers are now preferring to look for automation to replace labour problems and high cost that they are facing (Yang, Dickinson, Wu, & Lang, 2007).

Hand-picking has been a conventional way of harvesting tomatoes in Malaysia. However, depending on fruits that to be harvested, the technique may differ. In the early 1960s, the idea of shaking or knocking fruits down has been introduced but it can cause several problems such as physical damage of fruits that fell from the tree and physical damage of tree (Kurhade et al., 2015). Hence, the concept of an automatic harvester was first proposed by Schertz and Brown (1968). The proposed of the device is uses a robotic arm to position within a picking range before picking off from the tree. Besides, Jutras and Coppock (1968) proposed a first system that reduces unproductive time in picking fruits. They proposed several solutions to mechanized of citrus fruit picking by eliminated picking bag and replaced with a moveable catch frame that attached at the centre of four adjacent trees to relieve the pickers from the weight of the picked fruits. Furthermore, a self-propelled grove hoist that can move vertically and horizontally is to replace the ladder to pick a tall tree fruit and lastly, a shaker to replace human hand to comb and shake the fruits off. (Jutras & G.E.Coppock, 1968). However, Sistler (1987) claimed that research and development vision of machine, intelligent machine and robotics are the answers to improve

*CORRESPONDING AUTHOR | Muhammad Aizzat Zakaria | ✉ maizzat@ump.edu.my

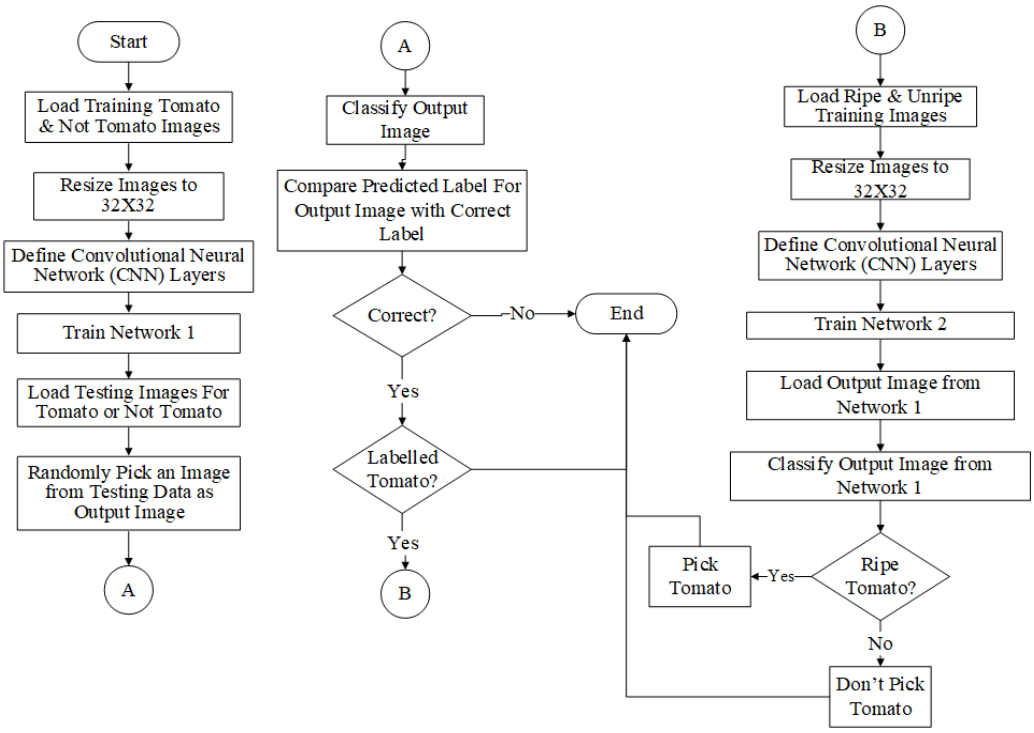

Figure 1: Deep learning algorithm flow chart for identifying ripe or unripe tomato.

sustainability. Besides, harvesting technology using robots for vegetables and fruits has been developed for nearly half a period and has been a popular subject for research in recent years such as in Feng, Wang, Wang, & Li (2015), Yaguchi, Nagahama, Hasegawa, & Inaba (2016), Yuanshen et al. (2016) and Taqi, Al-Langawi, Abdulraheem, & El-Abd (2017).

However, clustered fruits and vegetable recognition has been a challenging topic in researchers on the vision system of harvesting robot. The visual system, as an essential part of the harvesting robot that can assure harvesting quality and timing harvesting. It also plays an important role in information acquisition of the vision system to recognize the harvesting object (Xiang, Ying, & Jiang, 2013b). There are several techniques and algorithm that have been developed for image segmentation in fruits such as mathematical morphology (Xiang et al., 2013b), HSV and watershed algorithm (Malik et al., 2018), circle regression (R. Xiang, Y. Ying, & H. Jiang, 2013) and K-means clustering (Yin, Chai, Yang, & Mittal, 2009). However, there is no general solution to any image segmentation problem.

In this study, a tomato classification system was developed for images based on Convolutional Neural Network (CNN) which has good robustness in conventional image processing methods.

Deep Learning Algorithm for Identifying Ripeness Level Of Tomato

When developing an autonomous tomato harvesting robotic system, tomato recognition is a vital part before picking a tomato. Furthermore, it is not enough to provide a good efficiency of harvesting of the system if it is unable to identify the ripeness of the tomato as tomato farmers only harvest ripe ones only. Hence, a deep learning model is developed to identify the images whether is a tomato or not a tomato and followed by ripe or unripe tomato.

Figure 1 illustrates the flow chart of deep learning algorithm for identifying the images whether it is a ripe or unripe tomato. The algorithm consists of two networks:

Network 1: Classification of Tomato and Not Tomato
Network 2: Classification of Ripe and Not Tomato

where network 1 only classifies each input images into either tomato or not tomato and network 2 only classifies each input images into either ripe or unripe tomato.

The networks' structures of CNN have 15 layers which consists of convolution layer, maximum pooling, ReLU, average pooling, fully-connected layer and softmax layer. Besides, the training options are also similar. However, this algorithm randomly picked an output image from a set of testing data of

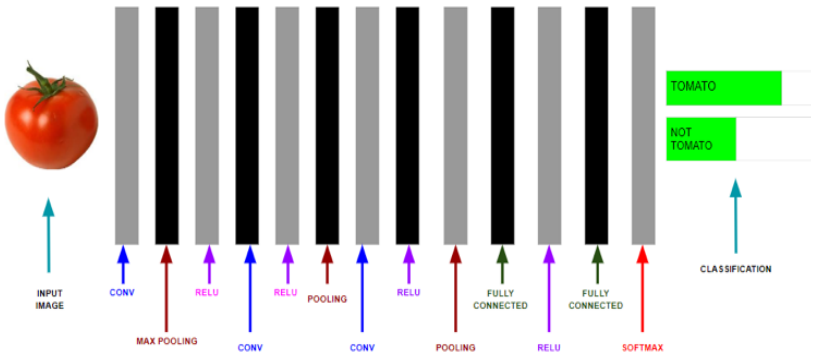

Figure 2. CNN layers for tomato and not tomato.

tomato and not tomato network and this output image must be a tomato so that network 2 can classify the output image is either a ripe tomato or unripe tomato. If it is a ripe tomato, then the tomato is ready to pick but if it is an unripe tomato, then the tomato is not ready to pick and shall not pick.

For both networks, the first process in the algorithm starts with loading images into image data store but before the input images direct to the network, the images are resized to [32x32x3] of width 32, height 32 with 3 colour channels of RGB to avoid any error in extracting feature or information from the images. Then, convolutional neural network (CNN) layers are defined and the network is trained. After the network is trained, a set of testing data is loaded into the image data store in the programming software and begin with a classification for each testing images that is loaded. Lastly, the accuracy of classification is calculated.

Training and Testing Data

Table 1 shows the amount of training data and test data for both networks. The total amount of images that loaded into the image data store is 866. Every each of the images is taken from Google Images and only .jpg format are chosen for training and testing in both networks. For network 1, 300 training data for both tomato and not tomato categories, 21 and 12 testing data for tomato and not tomato respectively. Moreover, network 2 consists of 100 training data for both ripe and unripe tomato, 15 and 18 testing data for ripe and unripe tomato respectively. The total training data that loaded into both network are 800 images and testing data is 66 images.

Defining Convolutional Neural Network (CNN) Layers

To begin with, developing an algorithm for the convolutional neural network, each layer must be defined. Figure 2 illustrates the layers used for tomato and not tomato CNN model and there is total of 15 layers in the network that is selected. An input image pass through the first layer which is convolution layer then followed by maximum pooling, ReLU, convolution layer, ReLU, pooling, convolution layer, ReLU, pooling, fully-connected layer, ReLU, fully-connected layer and lastly softmax that turn the integers into probability.

Table 1. Number of Training Data for Tomato and Not Tomato Network.

Categories	Training Data	Test Data	Total
Tomato	300	21	321
Not Tomato	300	12	312
Ripe Tomato	100	15	115
Unripe Tomato	100	18	118
Total	800	66	866

A convolution layer is a layer that involves filter that contains an array of number or weights that produce new images from a matrix multiplication is performed and sums into a feature map. This step is to reduce the size of the image and making the processing faster and easier. However, there are some parameters need to be defined such as stride padding, number of filter and size of filter as these parameters control the behaviour of each convolution layers. A stride is an amount of filter shift in the image. The size of feature map output depends on the size of the stride. The larger the stride, the feature map output will be smaller. In this case, stride value of 1 is used to shift in the image. Since the reduced size of input image is 32x32, after it performs matrix multiplication is performed, the feature map size would be 16 as the filter size is 3.

The size of feature map is always smaller compared to the input image, to prevent feature map from shrinking, padding is used. A zero padding is a process of padding the border of the input images to give dimensionality of output volume of images. Padding also improves performance and ensures the filter and stride size will fit the input.

After the convolution layer, maximum pooling layer is defined. Maximum pooling takes the

maximum value in each window. It decreases the feature map size at the same time preserve the significant information. However, for average pooling, it takes the average element in a window. Next, network of rectified linear unit (ReLU) layer is defined to perform threshold operation to each of the element of input where the values that are below negative will returns to 0. By using ReLU, network could be improved the learning speed.

Train Network

Furthermore, once the network has been structured and the layer is defined, the network is ready to be trained. Training option for deep learning that is selected to train the network are:

Table 2. Training options parameters.

Options	Descriptions
Solver	Stochastic Gradient Descent with Momentum (SGDM)
Initial Learn Rate	0.001
Learn Rate Drop Factor	0.1
Learn rate Drop Period	8
Maximum Epoch	10
Regularization	L_2
Mini Batch Size	100

In other words, the maximum number of epochs to use for training is 10 where iteration is taken in gradient descent to minimize loss function using mini batch of 100. A mini batch is a subset of training used to evaluate gradient of loss function and update the weights of each hidden layers. The initial learning rate is set to 0.001 in order to converge or can obtain optimal results. Furthermore, learn rate drop factor applies learning rate every time when number of epoch of 8 (learn rate drop period) passes. L_2 regularization is used to smoothen distribution of parameter and reduce the magnitude of parameter to avoid overfitting model.

After numbers of convolutional and pooling layers that stacked together to extract more features representations in the network, fully connected layers take a role to interpret these feature representations and perform function of reasoning. This composite information from input images that held by fully connected layer is then allow softmax layer to get a probability from neural outputs. Softmax layer do generalization to calculates a set of positive number of probability that adds to 1. If the probability that calculated by softmax layer in a network is higher in tomato category, hence, it is chosen to label the input image.

Accuracy of Network

After the testing images are classified into labels or categories, the predictions are tabulated into:

Table 3. Network 1 classification prediction.

	Tomato (Predicted)	Not Tomato (Predicted)
Tomato (Actual)	True Positive (TP)	False Negative (FN)
Not Tomato (Actual)	False Positive (FP)	True Negative (TN)

Table 4. Network 2 classification prediction.

	Ripe Tomato (Predicted)	Unripe Tomato (Predicted)
Ripe Tomato (Actual)	True Positive (TP)	False Negative (FN)
Unripe Tomato (Actual)	False Positive (FP)	True Negative (TN)

After the classification are tabulated, the accuracy is calculated using

$$accuracy(\%) = \frac{TP+TN}{(TP+TN+FP+FN)} \times 100 \quad (1)$$

where TP = True Positives where the image is predicted as tomato and the actual is tomato, TN = True Negatives where the image is predicted as not tomato and the actual is not tomato , FP = False Positives where the image predicted as tomato but actual is not tomato, FN = False Negatives where the image predicted as not tomato but actual is a tomato.

Then, True Positive Rate, False Positive Rate, True Negative Rate, False Negative Rate are calculated to see how often it predict each category. True Positive Rate is the rate when the image is actually tomato, how often does it predict tomato. False Positive Rate is the rate when it's actually not tomato, how often does it predict tomato. Furthermore, True Negative Rate is rate when it's actually not tomato, how often does it predict not tomato and False Negative Rate is rate when it's actually tomato, how often does it predict a not tomato.

Lastly, each image is labelled with predicted categories such as tomato, not tomato, ripe tomato and not tomato. If the predicted label differs from the actual label, red text is indicated, and the predicted label is same from actual label of image, green text is indicated. Figure 3 illustrates images with correct classification with label and incorrect classification with label. If the image is with correct classification, the text is green while with incorrect classification, the text is in red.

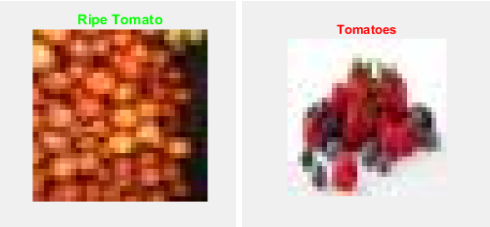

Figure 3. Images with correct classification with label and incorrect classification with label.

Results and Discussion

This section details the prediction of tomatoes for both network 1 and network 2. Table 5 shows the classification prediction of network 1 for 10 trials.

Table 5. Classification prediction of Network 1 for every trial.

Trial	Actual Photo	Tomato Predicted	Not Tomato Predicted
1	Tomato	0.9167	0.0833
	Not Tomato	0.1905	0.8095
2	Tomato	0.8333	0.1667
	Not Tomato	0.3810	0.6190
3	Tomato	1.0000	0.0000
	Not Tomato	0.1905	0.8095
4	Tomato	0.7500	0.2500
	Not Tomato	0.1905	0.8095
5	Tomato	0.9167	0.0833
	Not Tomato	0.2857	0.7143
6	Tomato	0.8333	0.1667
	Not Tomato	0.4286	0.5714
7	Tomato	0.8330	0.1667
	Not Tomato	0.1905	0.8095
8	Tomato	1.0000	0.0000
	Not Tomato	0.2857	0.7143
9	Tomato	0.5000	0.5000
	Not Tomato	0.5238	0.4762
10	Tomato	0.9167	0.0833
	Not Tomato	0.1905	0.8095

Table 6. Accuracy for Network 1.

Trial	Accuracy
1	0.8485
2	0.6970
3	0.8788
4	0.7879
5	0.7879
6	0.6667
7	0.8182
8	0.8182
9	0.4849
10	0.8485
Average	**0.7637**

Table 6 shows the results of accuracy for network 1 (tomato and not tomato) for 10 trials and the average accuracy that obtained is 76.37%. This accuracy is affected by multiple factors which include the type training data for not tomato category that is loaded to the net. The type training data that loaded into the network consist of images of fruits or vegetables that have similar feature or colour to tomatoes such as rambutan, dragon fruit, cherry, strawberry, raspberry, pomegranate, and apple. Hence, the network is confused with the features and colours that are extracted by the layers in CNN.

Table 7. Classification prediction of Network 2 for every trial.

Trial	Actual Photo	Ripe Tomato Predicted	Unripe Tomato Predicted
1	Ripe Tomato	0.9333	0.6667
	Unripe Tomato	0.0000	1.0000
2	Ripe Tomato	1.0000	0.0000
	Unripe Tomato	0.0000	1.0000
3	Ripe Tomato	1.0000	0.0000
	Unripe Tomato	0.0000	1.0000
4	Ripe Tomato	1.0000	0.0000
	Unripe Tomato	0.0000	1.0000
5	Ripe Tomato	0.9333	0.6667
	Unripe Tomato	0.0000	1.0000
6	Ripe Tomato	1.0000	0.0000
	Unripe Tomato	0.0000	1.0000
7	Ripe Tomato	0.9333	0.6667
	Unripe Tomato	0.0000	1.0000
8	Ripe Tomato	1.0000	0.0000
	Unripe Tomato	0.0000	1.0000
9	Ripe Tomato	1.0000	0.0000
	Unripe Tomato	0.0000	1.0000
10	Ripe Tomato	0.9333	0.6667
	Unripe Tomato	0.0000	1.0000

Table 8. Accuracy for Network 2.

Trial	Accuracy
1	0.9697
2	1.0000
3	1.0000
4	1.0000
5	0.9697
6	1.0000
7	0.9697
8	1.0000
9	1.0000
10	0.9697
Average	0.9879

Table 8 shows the results of accuracy for network 2 (Ripe & Unripe Tomato) for 10 trials and the average accuracy that obtained is 98.79% which only one image that classified wrongly. The percentage accuracy that obtained from the network is almost

100% for network 2. This is due to the features and colours that are extracted by the layers in convolutional neural network. Hence, the network can differentiate each image by colour as ripe tomato is red and unripe tomato is green. However, for accuracy with 96.97%, most trials that are running, the same image was labelled wrongly (Figure 4).

Figure 4. Image of ripe tomato that predicts wrongly by network 2.

Figure 5 and Figure 6 shows the red histogram from image and green histogram from image from Figure 4 respectively. The histograms represent the distribution of colour of image. It represents the number of pixels of that colour in red and green space. The pixel in both histograms is almost equal as the pattern is almost similar. Hence, it is hard for the network to differentiate the image whether it is a ripe or an unripe tomato.

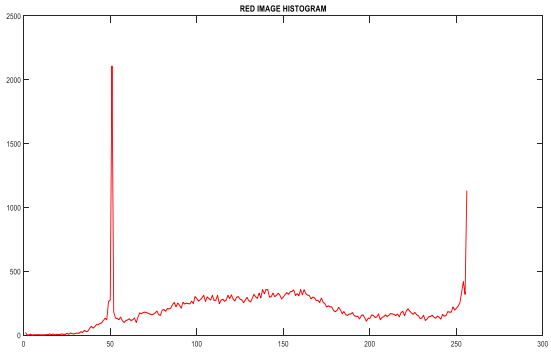

Figure 5. Red histogram from image from Figure 4.

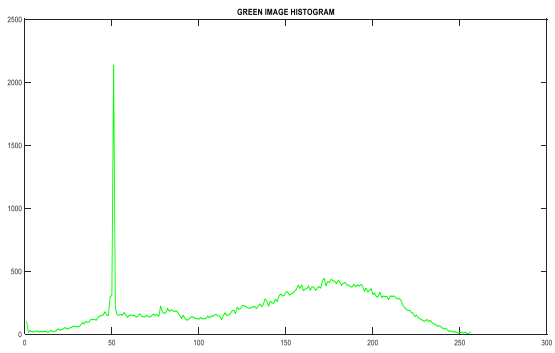

Figure 6. Green histogram from image from Figure 4.

Conclusion

To develop a tomato harvesting system, the main part is not only limited to the mechanical but the recognition of an object. Hence, deep learning is used to classify and recognize a tomato. A set of training data are loaded into a network to extract the features from the images, and from the trained network, each testing images are classified. From the classified images, accuracy is calculated based on the predictions that being made from the trained network. The accuracy for network 1 (tomato and not tomato) and network 2 (ripe and unripe tomato) is 76.366% and 98.788%. The accuracy is higher for network 2 compared to network 1. This accuracy shows that the networks able to classify both tomato or not tomato and ripe or unripe tomato.

However, the accuracy for network 1 can be improved by increasing the training data and tuning the training option that can extract more features and information from the images. The total training data that used in this experiment is 866 images. It can be improved by increasing the training data to more than 1000 images so that the network can extract more features from the images. Besides, the training option can be improved by decrease the learning rate over epochs and batch size that equal to training data size.

Acknowledgement

This research is supported by FRGS Research Grant Funding RDU190104 and “in-kind” support from KUKA Robot Automation Sdn. Bhd.

References

[1] D. Bhowmik, K. P. S. Kumar, S. Paswan, and S. Srivastava, "Tomato-A Natural Medicine and Its Health Benefits," *Journal of Pharmacognosy and Phytochemistry,* vol. 1, no. 1, pp. 34-43, 2012.

[2] G. M. N. Islam, F. M. Arshad, A. Radam, and E. F. Alias, "Good agricultural practices (GAP) of tomatoes in Malaysia: Evidences from Cameron Highlands," *African Journal of Business Management,* vol. 6, no. 27, pp. 7969-7976, 2012.

[3] H. Rahim, M. A. M. A. Wahab, M. Z. M. Amin, A. Harun, and M. T. Haimid, "Technological adoption evaluation of agricultural and food sectors towards modern agriculture: Tomato," *Economic and Technology Management Review,* vol. 12, pp. 41-53, 2017.

[4] L. Yang, J. Dickinson, Q. M. J. Wu, and S. Lang, "A Fruit Recognition Method for Automatic Harvesting," 2007.

[5] A. J. Kurhade, A. M. Deshpande, and R. D. Dongare, "Review on “Automation in Fruit Harvesting," *International Journal of Latest Trends in Engineering and Technology (IJLTET),* vol. 6, no. 2, pp. 1-15, 2015.

[6] C. E. Schertz and G. K. Brown, "Basic Considerations in Mechanizing Citrus Harvest," *Transactions of the ASAE,* vol. 11, no. 3, pp. 343-0346, 1968.

[7] P. J. Jutras and G.E.Coppock, "Mechanization of Citrus Fruit Picking," pp. 343-346, 1968.

[8] F. E. Sistler, "Robotics and Intelligent Machines in Agriculture," *IEEE Journal Of Robotics And Automation,* vol. RA-3, no. 1, pp. 3-6, 1987.

[9] Q. Feng, X. Wang, G. Wang, and Z. Li, "Design and Test of Tomatoes Harvesting Robot," 2015.

[10] H. Yaguchi, K. Nagahama, T. Hasegawa, and M. Inaba, "Development of An Autonomous Tomato Harvesting Robot with Rotational Plucking Gripper," *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* pp. 652 - 657, 2016.

[11] Z. Yuanshen, L. Gong, C. Liu, and Y. Huang, "Dual-arm Robot Design and Testing for Harvesting Tomato in Greenhouse," pp. 161–165, 2016.

[12] F. Taqi, F. Al-Langawi, H. Abdulraheem, and M. El-Abd, "A Cherry-Tomato Harvesting Robot," *International Conference on Advanced Robotics (ICAR),* pp. 463-468, 2017.

[13] R. Xiang, Y. Ying, and H. Jiang, "Tests of a Recognition Algorithm for Clustered Tomatoes Based On Mathematical Morphology," *6th International Congress on Image and Signal Processing,* pp. 464-468, 2013.

[14] M. H. Malik, T. Zhang, H. Li, M. Zhang, S. Shabbir, and A. Saeed, "Mature Tomato Fruit Detection Algorithm Based on improved HSV and Watershed Algorithm," *IFAC PapersOnLine,* pp. 431–436, 2018.

[15] R. Xiang, Y. Ying, and H. Jiang, "A Recognition Algorithm for Occluded Tomatoes Based on Circle Regression," in *2013 6th International Congress on Image and Signal Processing (CISP)*, 2013, vol. 2, pp. 713-717.

[16] H. Yin, Y. Chai, S. X. Yang, and G. S. Mittal, "Ripe Tomato Recognition and Localization for a Tomato Harvesting Robotic System," presented at the 2009 International Conference of Soft Computing and Pattern Recognition, 2009.