Convolutional Neural Networks Based Optimal Management of Agricultural Crops

Mohammed Thajeel Abdullah[1,2], Wathiq Laftah Al-Yaseen[1], Qusay Abdullah Abed[1]

[1]Kerbala Technical Institute, Al-Furat Al-Awsat Technical University, 56001, Kerbala, Iraq
[2]Informatics Institute for Postgraduate Studies, Iraqi Commission for Computers and Informatics, Baghdad, Iraq
Corresponding Author:
Qusay Abdullah Abed,
Kerbala Technical Institute,
Al-Furat Al-Awsat Technical University,
56001, Kerbala, Iraq.
Email: inkr.ks@atu.edu.iq

Abstract
Given the importance of agriculture, food supply, and food security, as well as population growth, the use of state-of-the-art technologies to increase agricultural productivity and mechanization with the least amount of loss and damage to crops and human beings has been highly prioritized. A great body of research has been conducted on and many solutions have been adopted for agricultural mechanization and reduced and optimized consumption of the available herbicides. Using convolutional neural networks and deep learning, this study sought to increase the accuracy of detecting grapes in the vineyard and of weeds in fields. For this purpose, the VGG16 Standard was utilized. The results indicated a 99% learning accuracy in the learning section for grape and weed detection. The validation and the final accuracy of detection for the machine designed was 63% for grapes detection and 95% for weeds. It was also demonstrated that the proposed method outperformed the KNN, decision tree, and random forest algorithms compared to the other algorithms and methods.

Keywords: Machine Vision*,* Field and Garden Management*,* Convolutional Neural Network*,* Deep Learning

1. INTRODUCTION

The world population is expected to reach 9.7 billion by 2050 and 11.2 billion at the turn of this century. Therefore, agriculture plays an essential role in ensuring food security, reducing poverty, and improve economic development [1]. In general, foodstuff production is affected by several factors, including pests, weeds, pathogens, nutrients, water, sunlight, soil erosion, environmental effects, and farmlands. The process of the manual examination of agricultural products advances gradually and is error-prone due to human error. The application of technology to produce food stuff is of great importance. The past few years have witnessed the significant breakthrough of intelligent systems in the agricultural technologies in machine learning, especially artificial neural networks (ANNs) with deep learning. Drawing on high-efficiency calculations and accessibility of the frameworks for applying computer vision techniques, machine learning has significantly contributed to solving many agricultural industry problems. Agricultural technology (a.k.a. agrotechnology) refers to advanced monitoring and data analysis to optimize the productivity and quality of crops. To this end, the implementation of intelligent systems allows the agricultural industry to improve the productivity of agricultural products by making timely decisions and optimum use of valuable resources, including lands and water [2]. This domain encompasses several applications associated with sensors, artificial intelligence, big data, and robotics utilized to improve the global sources of food. Typical applications, including product management [3], livestock management [4], water and soil management [5], etc.

Grapes are a common type of fruit worldwide, and the production of grapes is increasing due to the development of the grape industry. However, grape harvesting is a risky and vibrant task [6]. Thus, it is of vital importance to develop an automated grape harvesting system. Automated harvesting primarily involves discovering and locating grapes in a vineyard using artificial vision. This is the core of each grape harvesting robot. Nevertheless, the target detection becomes challenging due to the overcrowded vineyard environment. It is challenging to develop the vision system to the extent of human vision to easily detect grapes in a given vineyard, particularly when grapes and the background are of the same color [7].

Weed can be defined as plants of vigorous growth among agricultural products. In agriculture, weeds are considered a serious threat by farmers. Weeds cause damage to lands and water. They host pests, pathogens, and parasites. Some of them also pose a threat to human and animal health [8]. Mechanical techniques (using tractor and agricultural tools) can be applied to fight against weeds for row crops, such as maize (*Zea mays*), sugar beat (*Beta vulgaris subsp*), wheat (*Triticum*), and potato (*Solanum tuberosum L.*). This method can help kill 50% of weeds between two crop rows. The most important weeds grown in potato fields are *Amaranthus spp.*, *Chenopodium album*, *Echinochloa crus galli*, *Sorghum halepense*, *Convol vulus*, and *Portulaca oleracea* [9] (Vista News Hub, 2020).

Autonomous robots (a.k.a. autorobots or autobots), increasingly developed and widely used in agricultural systems, are intelligent machines capable of carrying out heavy agricultural activities, such as continuous circulation in the field and identification of different stages of plant growth without human intervention. The precise detection and location of different parts of a plant, including fruits, flowers, etc., is vital in agricultural environments performed by different target detection algorithms. Hence, an increase in farmers' capacity for producing more products is contingent upon using such autorobots with high flexibility to operate under various conditions and with different products [10]. Using machine vision, this study seeks to present a method to detect grapes in vineyards and weeds in potato fields.

2. RELATED WORKS

In the research by Abouzahir et al., [11], three distinct classes were established using a histogram based on color indicators: soil, soybean (*Glycine max*), and weeds. Soybeans were investigated and detected using two classification methods, i.e., BPNN and SVM. The results indicated an overall accuracy of 96.601% and 95.078% for BPNN and SVM, respectively [12]. Tomatoes (*Solanum lycopersicum*) in the fields were detected based on the HIS color model for image segmentation, accuracy detection, and better accuracy. The results suggested that a simple harvesting epoch lasted for about 24 seconds, and the success rate of tomato harvesting was 83.9%.

Moreover, Hughes et al., [13] designed a system for separating healthy lettuce leaves using machine vision and robotics. Islam et al., [14] introduced a hybrid image processing-machine learning method to detect the disease from the plant leaf image. The proposed classification method and SVM indicated the classification of more than 300 images with an accuracy of 95%. In the research by Jiang et al., [15], the Apple Leaf Diseases Dataset (ALDD), constituted of laboratory images and complex images under real field conditions, was created by image labeling. The CNN method was utilized to detect the disease. The results demonstrated that the disease detection accuracy and speed were 78.8% and 23.13 detection per second, respectively.

Furthermore, Kamath et al., [16] employed random forest and SVM methods to detect weeds in rice fields. The results indicated an increase in weed detection in the rice fields on the WSN platform controlling these fields. Kaur and Min [17] presented an agricultural detection system based on the identification of crop rows in maize fields in the presence of weeds. The experimental results indicated the effectiveness of this technique in the better management of maize fields. Alchanatis et al., [18] presented a weed detection mechanism in cotton fields using an imaging system. A weed detection accuracy of 86% was obtained in this study. Armstrong et al., [19] proposed a method for detecting low-density weeds in maize fields using multispectral imaging (MSI). An accuracy of 91% was obtained in this study. Rehman [20] introduced a weed detection mechanism in wild blueberry fields. The best performance in the laboratory-scale was obtained with an accuracy of 94.98% and 80.93% for the training and test data sets, respectively. Yu et al., [21] proposed a method for detecting weeds in grasslands based on a deep convolutional neural network (DCNN) model. The results suggested that DCNN-based weed detection can apply an effective decision-making system on the sub-system of a machine vision system and can be an appropriate herbicide for weed control. In their research, Sabzi et al., [22] presented 4299 samples of five weed species based on video processing and metaheuristic classifiers for online identification and classification of Solanum tuberosum L. The classification results indicate a classification accuracy of 98% compared to the test set with a maximum speed of 0.15 m/s. Wang et al., [23] summarized the advances in weed detection using terrestrial vision and image processing techniques. In their study, Smith et al., [24] obtained an accuracy of 95.6% and an accuracy of 84% in weed classification using CNN with continuous training. Yu et al., [25] obtained an accuracy of 84% in weed detection using DCNN.

3.   RESEARCH METHOD

CNNs are usually employed for processing data with a known spatial relation or a network-like topology. A CNN is composed of an input layer and an output layer with several hidden layers therein, where they can be convective, synergistic, or fully connected, as shown in Figure 1. In CNNs, accumulation operation is carried out, and each energy transfer operation possesses a core. The matrix core or filter is smaller than the original image. This step demonstrates how many pixels are moved horizontally or vertically by the filter when using the input image (A Beginner's Guide to Understanding Convolutional Neural Networks Part 2, 2020).
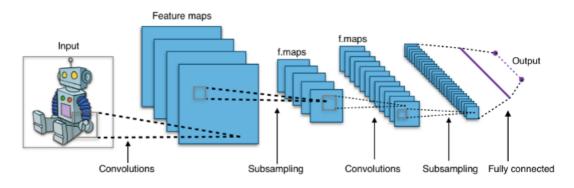

Figure 1. A review of a CNN

3.1. Excitation Layer

A CNN is comprised of several convolutional layers utilized for extracting features from the network input. A major concept of CNNs is that the same transformation is applied in all locations. Given a two-dimensional (2D) image, i, and a small matrix, K, with the size h_w, the solved image can be computed. Matrix K is utilized to extract the features of an $I*K$ image. It should be noted that not all input nodes are connected to output nodes. Besides, as the filter moves around itself, the same weights are applied to the whole image. The image is modified as a result of using weights. The weight is a parameter set during training; however, weights are constant during calculations. The weight of a filter can take any combination of values, depending on the filter training procedure.

$$\begin{bmatrix} 2.0 & 3.0 & 0.0 & 5.0 & 2.5 \\ 2.0 & 1.5 & 0.5 & 0.0 & 7.0 \\ 1.5 & 5.0 & 5.0 & 3.0 & 2.0 \\ 3.0 & 5.0 & 7.0 & 1.5 & 0.0 \\ 2.0 & 5.0 & 2.0 & 1.5 & 2.0 \end{bmatrix} \times \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} = \begin{bmatrix} 4.25 & 2.5 & .. & .. \\ .. & .. & .. & .. \\ .. & .. & .. & .. \\ .. & .. & .. & .. \end{bmatrix}$$

Given a 2D image, i, and a small matrix, K, with the size h_w, the solved image can be computed. Matrix K is the kernel core, which is assumed to be a way of extracting features of an $I*K$ image by making the core cover the image using any method. It is then obtained by summing up all products with the logical element between the image and the core, as expressed by:

$$(I * K)_{xy} = \sum_{i=1}^{h} \sum_{j=1}^{w} K_{ij} * I_{x+i-1,y+j-1} \tag{1}$$

Filters are collected from channels. Each input channel has a weighted matrix as follows.

$$O_{h.i.j} = \varphi(I_{1(i.j:i+1.j+1)} * W_{h.k}) \tag{2}$$

where O is the output, i is the input, W is the weights, and φ is the activation function.

3.2. Pooling Layer

In the pooling layer in a CNN, a volume is reduced spatially. Sampling is lowered in each input cut independently. The accumulation operation in CNNs is conducted basically to reduce the number of network parameters due to detecting features from changing scale and direction (Convolutional Neural Networks Tutorial in TensorFlow, 2020).

3.3. Fully Connected Layer

Fully connected layers solve object classification from the output of the accumulating layers. In summary, it can be stated that this is a standard neural network classifier connected to the end of a high-level feature extractor. The output of the final (terminal) pooling layer is a large number of $X*Y$ matrix channels. The output must reach a flat $1*N$ tensor to connect the pooling layer to the fully connected layer (Convolutional Neural Networks Tutorial in TensorFlow, 2020).

3.4. Training Process

The process of deep learning is based on the training stage. The purpose of training is to enable the model to learn the task in question by looking at given samples of the data set. After training, we shall confirm that the model has learned something by testing itself on the unobserved validation data before learning. The ability to predict an accurate class of an invisible sample indicates the fitness of the model.

3.5. Learning Rate

Learning rate is an essential parameter in the training process. The higher the learning rate, the larger the changes in weights, meaning that bigger steps are taken. A high learning rate enables the model to quickly converge in optimum sets of weights. If the learning rate is very high, steps can be substantial but not precise enough to find the optimal point.

4.6. Optimizer

An optimizer is utilized to minimize or maximize the loss performance. Given $y = f(x)$, its derivative is $\frac{dy}{dx}$. The derivative of the function gives the gradient of the function at x. The derivative determines how to measure a slight change in the input to set the output. The derivative is beneficial in minimizing a function because it demonstrates how to change x to improve the interval of y. This method is called gradient descent. The first derivative is used instead of the second derivative due to complexity problems. It would take a long time to compute the second derivatives and would contain a heavy computational load.

4.7. Loss Performance

In deep learning, the purpose of training is to increase the number of certain performance criteria defined according to the test set. It can be difficult to measure performance; thus, the loss performance will be reduced by the likely improvement of the indirect performance. Many loss functions can be applied to the main target variables. Cross-entropy is the most popular method in this regard. The utilized model often defined the distribution of $p(y|x;\theta)$. This means that the principle of maximum probability can be employed by cross-intersection of the training data and the distribution of the model as the loss function.

In this project, categorical_crossentropy is used as the loss function. Categorical cross-entropy turns the cross-intersection between an approximate distribution and an actual distribution. The actual distribution in this project refers to the empirical distribution of the training data. The intersection of two probability distributions determines the average number of bits required for identifying an event from a set of facilities. Cross-entropy measures the divergence of these two distributions, something that we are intended to minimize during the training.

$$H(p,q) = -\sum_x p(x)\log(q(x)) \tag{3}$$

where p is the empirical distribution, and q is the given probability distribution.

On the other hand, entropy is a function of distribution P, indicating the amount of anticipated information about an event sampled from P. It can be expressed as follows:

$$H(X) = -\sum_{i=1}^{n} P(x_i) log_b(P(x_i)) \tag{4}$$

where *b* is the utilized logarithm base, which is equal to 2.

The loss function takes the network predictions and the real target and calculates the score. The resulting score demonstrates how the network has classified this specific sample. A main feature in deep learning is using the score as a feedback signal to change the values of weights.

In a neural network, information is propagated on the network. The input gives the basic information, which is then transmitted to hidden layers and then the output. This is called "forward propagation" forward propagation continues until the loss of criterion $J(\theta)$ is reached. Reproagation utilizes partial derivatives to update the coefficients backwards. The gradient of the final layer of the weights is initially calculated, and the gradient of the first layer of the wrights is calculated at the end. This backpropagation improves gradient computation efficiency because the partial gradient computation of a layer is used again in the computation of the previous layer. In repropagation, the derivation is initiated using the chain law to the partial derivative of the error function, as follows:

$$\frac{\partial E}{\omega_{i.j}^{k}} = \frac{\partial E}{\alpha_j^k} \frac{\partial \alpha_j^k}{\omega_{i.j}^{k}} \tag{5}$$

where α_j^k represents the activation of a node *j* in a layer *k* before it is transformed into a nonlinear activation function (in this case, ReLU) to generate output.

$$\delta_j^k = \frac{\partial E}{\omega_{i.j}^{k}} \tag{6}$$

$$\frac{\partial \alpha_j^k}{\omega_{i.j}^{k}} = \frac{\partial}{\omega_{i.j}^{k}} \left(\sum_{l=0}^{r_{i-1}} \omega_{l.j}^{k} o_l^{k-1} \right) = o_i^{k-1} \tag{7}$$

The partial derivative of error function *E*, given the weight $W_{i.j}^{k}$, is as follows:

$$\frac{\partial E}{\omega_{i.j}^{k}} = \partial_j^k o_i^{k-1} \tag{8}$$

Thus, the partial derivative of the weight is obtained from error δ_j^k in node *j* in layer *k* and output o_i^{k-1} of node *i* in layer *k-1*.

Initially, random values are assigned to the network weight when training the network, indicating simply a series of random transformations. The output rate of the initial network is normally distant from the accurate and expected output, and the loss score is high. However, after further processing of samples, the weights are accurately corrected, thereby reducing the loss score. The gradient is corrected by the optimizer. The weight values are set after several repetitions of the exercise loop, leading to minimized loss performance. The weight values are typically local optima for the loss performance rather than global optima required for actual minimization. The general objective is to obtain a network with minimum loss.

4.   SIMULATION

Given the problem statement and literature review, Section 3 discusses the method of research. This section also describes the process of the implementation of the proposed method. The obtained results will be compared with those of other methods as well as previous research introduced in the preceding sections. This study uses the Python 3.0 programming language to implement the proposed method, and the obtained results are compared.

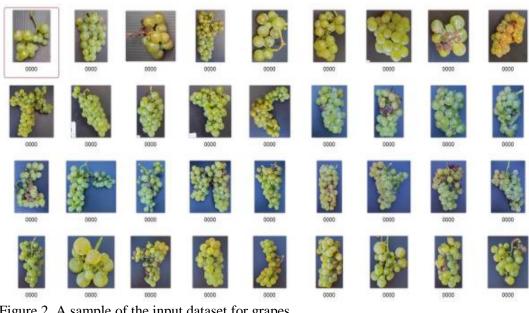### 4.1. Characterization of the utilized dataset

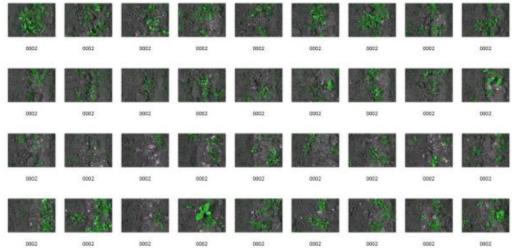Figure 2. A sample of the input dataset for grapes

Figure 3. A sample of the input dataset for weeds

4.2. Model Construction

The images are initially converted to grayscale because not all the images have high resolution. While some images may have a higher contrast, others may lack proper lighting. There are generally three classes. In the next step, the available classes are defined. Then, they are assigned to the available classes to increase the accuracy of training. Then, the data are loaded and converted to grayscale images. Afterward, the pixel values are rewritten to make them fall in the [255.0, 0.0] range. The size of the images is then modified to 32*32. Finally, they are encoded using one-hot labeling. Therefore, following label coding, each image will have three labels, all of which are 0 except for the label of the corresponding image. After labeling operation and its testing, the images are divided into three parts: validation, test, and training. We will use 20% of the images for validation and test and the rest for training. The VGG16 is used in this study to ensure standard performance. VGG16 is a prepared model, which has already been trained by large enterprises using supercomputers. In this model, there is a single-channel 32*32 image. There are convolution and pooling in each step. The convolution of each layer is given to the subsequent layer. The initial 1*32*32 image is initially made passing through the filter 32 times. This contributes to the creation of the first network output. It is then given to and pass through the next layer with 64 filters. There is a Max_Pooling layer in which the image is shrunk for two reasons: 1) filter size and 2) filter step. Adam optimizer, with a learning rate of 0.001, will be employed. Categorical_Crossentropy will be used as the loss function, and the criteria are accurate. This

model will be executed for a number of epochs to stabilize the learning rate. In this study, after 30 epochs, the learning rate on the data was stabilized, and a learning accuracy of 97% was obtained. After implementation, the validation rate was calculated to be 95% at best, which is acceptable.

5.   RESULTS AND ANALYSIS
In this section, the results of the simulation are discussed and analysis.

5.1. Results obtained for vineyard
After 45 epochs, the learning rate on the data was stabilized, and a learning accuracy of 99% was obtained as shown in Figure 4.
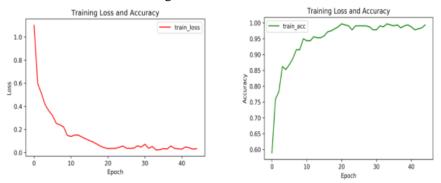

Figure 4. Accuracy and loss in the learning process

According to the diagram above, the learning operation has been stabilized, and the results of the validation of the designed machine are reliable. The results are as shown in Figure 5.
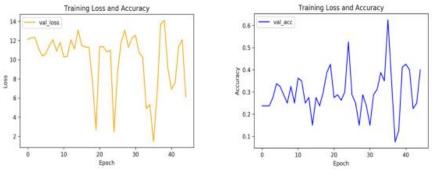

Figure 5. Accuracy and loss in the validation process

As indicated in the diagrams above, after 45 training and test epochs, the highest reliability rate of this model is 63%, and the lowest information loss rate is 1.45. Thus, it can be concluded that the designed machine had yielded a good performance, and this model has an accuracy of 63% using the VGG16 standard. Based on the obtained results and using the data and comparing them with the KNN, decision tree, random forest, and neural network algorithms, the following results were obtained in Figure 6.
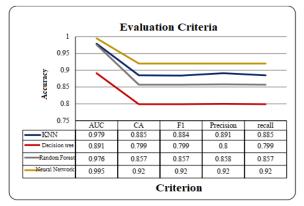
Figure 6. Evaluation Criteria

According to the obtained results, it can be concluded that the neural network had outperformed other algorithms. AUC refers to the area under the curve. The more the value of AUC is related to a larger classifier, the higher its efficiency. CA is the ratio of accurate responses (solutions) to the whole correct and incorrect responses. The higher the CA, the more the number of correctly guessed responses by the established network. Precision also refers to the ratio of the number of correct responses to the whole correct responses. F1-measure (F1-score) is an appropriate criterion for evaluating the accuracy of a test. In light of the foregoing, the neural network has higher accuracy than the other algorithms. This indicates that the use of CNNs with the VGG16 standard has outperformed the other algorithms.

5.2. Weed detection results

After 30 epochs, the learning rate on the data was stabilized, and a learning accuracy of 99% was obtained as shown in Figure 7.
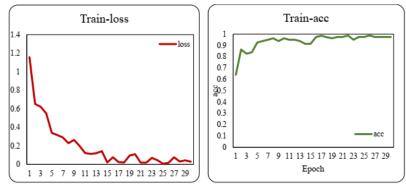

Figure 7. Accuracy and loss in the learning process

According to the diagram above, the learning operation has been stabilized, and the results of the validation of the designed machine are reliable. The results are as Figure 8.
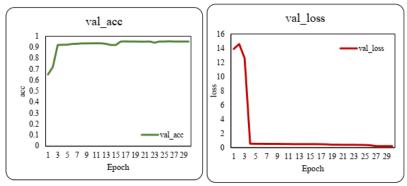

Figure 8. Accuracy and loss in the validation process

As indicated in the diagrams above, after 45 training and test epochs, the highest reliability rate of this model is 95%, and the lowest information loss rate is 0.21. Thus, it can be concluded that the designed machine had yielded a good performance, and this model has an accuracy of 95% using the VGG16 standard. Based on the obtained results and using the data and comparing them with the KNN, decision tree, random forest, and neural network algorithms, the following results were obtained in Figure 9.
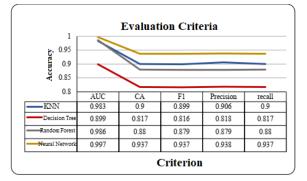

Figure 9. Evaluation Criteria

According to the obtained results, it can be concluded that the neural network had outperformed other algorithms. AUC refers to the area under the curve. The more the value of AUC is related to a larger classifier, the higher its efficiency. CA is the ratio of accurate responses (solutions) to the whole correct and incorrect responses. The higher the CA, the more the number of correctly guessed responses by the established network. Precision also refers to the ratio of the number of correct responses to the whole correct responses. Besides, Recall refers to the ratio of the number of correct responses to the whole correct responses. F1-measure (F1-score) is an appropriate criterion for evaluating the accuracy of a test. In light of the foregoing, the neural network has higher accuracy than the other algorithms. This indicates that the use of CNNs with the VGG16 standard has outperformed the other algorithms.

6. CONCLUSION

Given the problems raised above, this study was conducted on grape and weed detection. This study utilized deep learning and image processing to detect grapes in vineyards and weeds in potato fields. For this purpose, a machine was designed based on deep learning. It was structured based on the Keras class and the VGG16 standard. Due to the structure of this machine, the learning operation was conducted in 45 epochs, yielding a learning rate of 99%. The data utilized in this study included the data collected from the images of grapes in vineyards and weeds in potato fields, which is a complete dataset. The data were divided into three parts: training, test, and validation. After the learning operation, the training and test rates were stabilized. Given stabilized training, the obtained results are reliable. In 45 epochs during the validation stage, the best performance rate yielded by the designed machine was obtained to be 63%, which is acceptable.

REFERENCES

[1] J. Melorose, R. Perroy, and S. J. U. N. Careas, "World population prospects," vol. 1, no. 6042, pp. 587-92, 2015.

[2] M. G. Lampridi, C. G. Sørensen, and D. J. S. Bochtis, "Agricultural sustainability: A review of concepts and methods," vol. 11, no. 18, p. 5120, 2019.

[3] K. P. J. C. Ferentinos and E. i. Agriculture, "Deep learning models for plant disease detection and diagnosis," vol. 145, pp. 311-318, 2018.

[4] M. F. Hansen *et al.*, "Towards on-farm pig face recognition using convolutional neural networks," vol. 98, pp. 145-152, 2018.

[5] S. Dieleman, K. W. Willett, and J. J. M. n. o. t. r. a. s. Dambre, "Rotation-invariant convolutional neural networks for galaxy morphology prediction," vol. 450, no. 2, pp. 1441-1459, 2015.

[6]   L. Luo, Y. Tang, X. Zou, M. Ye, W. Feng, and G. J. B. E. Li, "Vision-based extraction of spatial information in grape clusters for harvesting robots," vol. 151, pp. 90-104, 2016.

[7]   L. Luo, Y. Tang, X. Zou, C. Wang, P. Zhang, and W. J. S. Feng, "Robust grape cluster detection in a vineyard by combining the AdaBoost framework and multiple color components," vol. 16, no. 12, p. 2098, 2016.

[8]   C. Muppala and V. J. J. o. P. Guruviah, "Machine vision detection of pests, diseases and weeds: A review," vol. 12, pp. 09-19, 2020.

[9]   *Vista   News   Hub*.   Available   (2020): https://vista.ir/content/27786/%D8%B9%D9%84%D9%81%E2%80%8C%D9%87%D8%A7 %DB%8C-%D9%87%D8%B1%D8%B2-%D8%AF%D8%B1- %D8%B3%DB%8C%D8%A8-%D8%B2%D9%85%DB%8C%D9%86%DB%8C/

[10]   Y.-Y. Zheng, J.-L. Kong, X.-B. Jin, X.-Y. Wang, T.-L. Su, and M. J. S. Zuo, "CropDeep: The crop vision dataset for deep-learning-based classification and detection in precision agriculture," vol. 19, no. 5, p. 1058, 2019.

[11]   S. Abouzahir, M. Sadik, and E. Sabir, "Enhanced approach for weeds species detection using machine vision," in *2018 International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, 2018, pp. 1-6: IEEE.

[12]   Q. Feng, X. Wang, G. Wang, and Z. Li, "Design and test of tomatoes harvesting robot," in *2015 IEEE International Conference on Information and Automation*, 2015, pp. 949-952: IEEE.

[13]   J. Hughes, L. Scimeca, I. Ifrim, P. Maiolino, F. J. I. R. Iida, and A. Letters, "Achieving robotically peeled lettuce," vol. 3, no. 4, pp. 4337-4342, 2018.

[14]   M. Islam, A. Dinh, K. Wahid, and P. Bhowmik, "Detection of potato diseases using image segmentation and multiclass support vector machine," in *2017 IEEE 30th canadian conference on electrical and computer engineering (CCECE)*, 2017, pp. 1-4: IEEE.

[15]   P. Jiang, Y. Chen, B. Liu, D. He, and C. J. I. A. Liang, "Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks," vol. 7, pp. 59069-59080, 2019.

[16]   R. Kamath, M. Balachandra, and S. J. I. A. Prabhu, "Raspberry pi as visual sensor nodes in precision agriculture: A study," vol. 7, pp. 45110-45122, 2019.

[17]   M. Kaur and C.-H. Min, "Automatic Crop Furrow Detection for Precision Agriculture," in *2018 IEEE 61st International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2018, pp. 520-523: IEEE.

[18]   V. Alchanatis, L. Ridel, A. Hetzroni, L. J. C. Yaroslavsky, and E. i. Agriculture, "Weed detection in multi-spectral images of cotton fields," vol. 47, no. 3, pp. 243-260, 2005.

[19]   J.-J. Q. Armstrong, R. D. Dirks, and K. D. J. W. T. Gibson, "The use of early season multispectral images for weed detection in corn," vol. 21, no. 4, pp. 857-862, 2007.

[20]   T. U. Rehman, Q. U. Zaman, Y. K. Chang, A. W. Schumann, K. W. J. C. Corscadden, and E. i. Agriculture, "Development and field evaluation of a machine vision based in-season weed detection system for wild blueberry," vol. 162, pp. 1-13, 2019.

[21]   J. Yu, A. W. Schumann, Z. Cao, S. M. Sharpe, and N. S. J. f. i. P. S. Boyd, "Weed detection in perennial ryegrass with deep learning convolutional neural network," vol. 10, p. 1422, 2019.

[22]   S. Sabzi, Y. Abbaspour-Gilandeh, and J. I. J. H. Arribas, "An automatic visible-range video weed detection, segmentation and classification prototype in potato field," vol. 6, no. 5, p. e03685, 2020.

[23]   A. Wang, W. Zhang, X. J. C. Wei, and e. i. agriculture, "A review on weed detection using ground-based machine vision and image processing techniques," vol. 158, pp. 226-240, 2019.

[24]   L. N. Smith, A. Byrne, M. F. Hansen, W. Zhang, and M. L. Smith, "Weed classification in grasslands using convolutional neural networks," in *Applications of Machine Learning*, 2019, vol. 11139, p. 1113919: International Society for Optics and Photonics.

[25]   J. Yu, S. M. Sharpe, A. W. Schumann, and N. S. J. E. j. o. a. Boyd, "Deep learning for image-based weed detection in turfgrass," vol. 104, pp. 78-84, 2019.