S AKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
VOL. 3, NO. 3, DECEMBER 2020
DOI: 10.35377/saucis.03.03.755269

Automatic Olive Peacock Spot Disease Recognition System Development by Using Single Shot Detector

Sinan Uğuz[1]
[1]Department of Computer Engineering, Isparta University of Applied Sciences, Isparta, Turkey;
sinanuguz@isparta.edu.tr; Phone Number: +90 505 499 58 81

Received 19 June 2020; Revised 25 June 2020; Accepted 9 September 2020; Published online 30 December 2020

Abstract

Among the artificial intelligence based studies conducted in the field of agriculture, disease recognition methods founded on deep learning are observed to become widespread. Due to the diversity and regional specificity of many plant species, studies performed in this field are not at the desired level. Olive peacock spot disease of the olive plant which grows only in certain regions in the world is a widely encountered disease particularly in Turkey. The aim of this research is to develop an olive peacock spot disease detection system using a Single Shot Detector (SSD) which is one the popular deep learning architectures to support olive farmers. This study presents a data set consisting of 1460 olive leaves samples for the detection of olive peacock spot disease. All of the images of the olive leaves which produced under controlled conditions were collected from Aegean region of Turkey during spring and summer. The data set was trained with different intersection over union (IoU) threshold values using SSD architecture. A 96% average precision (AP) value was obtained with IoU=0.5. As IOU value goes up from 0.5, erroneously classified olive peacock spot disease symptoms growed larger as well. The AP curve becomes flat when between 0.1 and 0.5, and it decreases when greater than 0.5. This analysis showed that the IoU significantly influenced the performance of SSD based model in detection of olive peacock spot disease. In addition to, trainings were performed by employing Pytorch library and a GUI was developed for the SSD based application using PyQt5 which is one of Pyhton's libraries. Results showed that the SSD was a robust tool for recognizing the olive peacock spot disease.

Keywords: olive peacock spot, single shot detector, deep learning, object detection

Single Shot Detector Kullanarak Otomatik Zeytin Halkalı Leke Hastalığı Tanıma Sistemi Geliştirilmesi

Öz

Tarım alanında gerçekleştirilen yapay zekâ temelli çalışmalar arasında, derin öğrenmeye dayanan hastalık tespiti uygulamalarının giderek yaygınlaştığı görülmektedir. Bitki türleri arasındaki çeşitlilik ve çoğu bitki türünün belirli coğrafyalarda yetişmesi bu alanda gerçekleştirilen çalışmaların sayısının istenen düzeyde olmadığını göstermektedir. Dünyada sadece belirli bölgelerde yetişen zeytin bitkisine ait halkalı leke hastalığı özellikle Türkiye'de yaygın olarak görülmektedir. Bu çalışmanın amacı, zeytin çiftçilerini desteklemek için popüler derin öğrenme mimarilerinden birisi olan Single Shot Detector (SSD) kullanarak zeytin halkalı leke hastalığını tespit sistemi geliştirmektir. Bu çalışmada zeytin halkalı leke hastalığının tespiti için 1460 adet zeytin yaprağı örneğini içeren veri seti oluşturulmuştur. Kontrollü koşullar altında üretilen tüm zeytin yaprak görüntüleri ilkbahar ve yaz dönemlerinde Türkiye'nin Ege bölgesinden toplanmıştır. Veri seti, SSD mimarisi üzerinde farklı IoU treshold değerleri ile eğitilmiştir. IoU=0.5 için %96 düzeyinde Average Precision (AP) değeri elde edilmiştir. IOU değeri 0.5'den yukarı doğru gittikçe, düşüş hatalı olarak sınıflandırılan olive peacock spot semptomu sayısının arttığı görülmüştür. AP eğrisi 0.1 ile 0.5 arasındayken düz hale gelir ve 0.5'den büyük olduğunda azalır. Bu analiz IoU'nun zeytin halkalı leke hastalığının tespitinde SSD temelli modelin performansını önemli şekilde etkilediğini göstermektedir. Ayrıca eğitimler Pytorch kütüphanesi kullanılarak gerçekleştirildi ve Python kütüphanelerinden biri olan PyQt5 kullanılarak SSD tabanlı uygulama için bir GUI geliştirildi. Sonuçlar, SSD'nin olive peacock spot hastalığının tanınması için güçlü bir araç olduğunu göstermiştir.

Anahtar Kelimeler: zeytin halkalı leke, single shot detector, derin öğrenme, nesne tespiti

1. Introduction

In recent years, significant improvements in dealing with problems such as classification, object detection and object segmentation were accomplished using architectures developed in the area of deep learning. As a consequence of these accomplishments, current literature surveys indicate an increase in the number of the studies directed at solving these problems through agricultural applications. Agricultural object detection studies can be said to be concentrating on problems such as visual fruit detection [1], disease detection [2], yield estimation via UAV photography [3] and regional disease identification [4]. Timely and accurate recognition of the disease is of great importance in the fight against plant diseases. Research studies found in the literature show that in recent years, Convolutional Neural Network (CNN) techniques have outperformed in the detection of the diseases of various plants. In the study carried out by Ozguven and Adem [5], automatic recognition of the leaf spot disease found in sugar beet was performed at three different significance levels using Faster R-CNN architecture. An accuracy rating of up to 95 % was obtained in the classfication and detection of the disease by training a data set comprising of 155 images. In their study, Selvaraj et al. [6] formed a data set of pests and diseases belonging to banana plant collecting images from various regions in Africa and India. Object detection of 18 different classes of diseases and pests was performed using three different CNN architectures consisting of Faster R-CNN InceptionV2, Faster R-CNN ResNet50 and SSD MobileNetV1. As a result of the study, accuracy scores exceeding 90% were obtained in most of the models employed. Li, et al. [7], on the other hand, achieved a mean Accuracy Precision (mAP) score of 0.885 as a result of trainings for the detection of plant pests utilizing ZFNet, AlexNet and ResNet architectures. Fuentes et al. [8] carried out a study on the detection of the diseases and pests of the tomato plant using Faster R-CNN, R-FCN and Single Shot Detector (SSD) architectures. The data set contained approximately 5000 images belonging to 10 different disease and pest classes. Data augmentation was implemented in the study and training results obtained from unaugmented data sets were compared. Consequently, success rate of the disease and pest detection was found to be higher for the augmented data set. Polder et al. [9] compared classical machine learning techniques with deep learning techniques based on Faster R-CNN architecture in order to detect lesions caused by a virus affecting tulip plants. According to the findings of this study, the utilized techniques were found to be almost the same in terms of their performances. Rong et al. [10] developed a system based on UNET architecture that separated walnut and foreigns objects from each other in real-time by a conveyor mechanism set-up by themselves. Findings with regards to accuracy and processing time were obtained in the segmentation and detection of foreign objects. Accordingly, an accuracy performance score of 95% was obtained in the object segmentation and detection processes carried out in durations less than 50 ms. Bhatt et al. [11] developed a YOLO architecture based disease and pest detection application for a data set of tea plants created under uncontrolled conditions. As a result of the study, a mAP score of 0.86 was obtained.

While diseases of some plants are detected, difficulties in accessing field experts and clinics are experienced due to the challenging conditions of the geographical regions where these plants are grown [12]. For this reason, popularization of computer vision applications for the detection of the diseases of some certain plants growing in certain geographical regions are of great importance. Turkey is among the leading countries in the world in the production of olive plant grown in certain geographies with an annual output of 183 thousand tonnes [13]. As is the case with all of the plants, olive diseases specific to the geography where it is grown can be observed. Olivepeacock spot is among the widespread olive diseases seen in Turkey. The disease leading to premature leaf abscission and branch death causes significant yield drops [14]. Cruz et al. [15] conducted a study on the classification of healthy and diseased olive leafs affected by Olive Quick Decline Syndrome which was explored in Italy and caused great damage. An accuracy score of 98.6% was obtained in the study that employed transfer learning method. In another study [16], a classfication success score of 99% was obtained by a model developed for the detection of 14 different diseases found on the trunks, leaves and fruits of olive plant using AlexNet architecture.

The requirement to conduct more studies on the olive plant has become obvious due to the large number of olive plant diseases and observation of different diseases varying in accordance with geographical regions. The limited number of studies in the literature on olive plant are found to be directed on the solution of classification problem. This study focused on the solution of an object detection problem directed on the identification of olive peacock spot disease widely seen in Turkey on the olive plant leaves. In line with this objective, a data set of 1460 olive leaves having olive peacock spot disease and collected under controlled conditions was created. Trainings were performed using SSD architecture and a program was developed visualizing detection of the diseased areas via a GUI. Object classification processes were also performed for the olive peacock spot disease even though in limited numbers. Nevertheless, to our knowledge, this study has the property of being the first object detection study conducted for the olive peacock spot disease.

2. Structure of SSD Architecture

R-CNN [17] and other similar architectures used in the CNN based deep learning applications are the architectures that run dependent on region proposal. High computational cost associated with region proposal process particularly leads to unsatisfactory results with respect to time in the object detection with real time applications. SSD architecture [18] produces more successful results in terms of time compared to R-CNN and similar architectures especially for the real time applications. VGG16 architecture should first be mentioned when the SSD architecture is examined. Because, the SSD architecture is based on the VGG16 architecture which provides less feature maps. In recent years, architectures that have been successful in solving object detection and object classification problems by being trained with images reaching millons in number have become prominent in Image Net challenges. One of these architectures is the VGG net architecture [19] that has two versions as VGG166 and VGG19. The VGG16 architecture seen in Figure 1 encompasses 13 convolution layers. These layers are divided into five blocks each having either two or three layers. Each block is complemented with a pooling layer that simplifies image outputs generated as a result of the convolution process. The last three layers are comprised of fully connected layers where classification tasks are perfomed and class scores are identified using neural network processes.

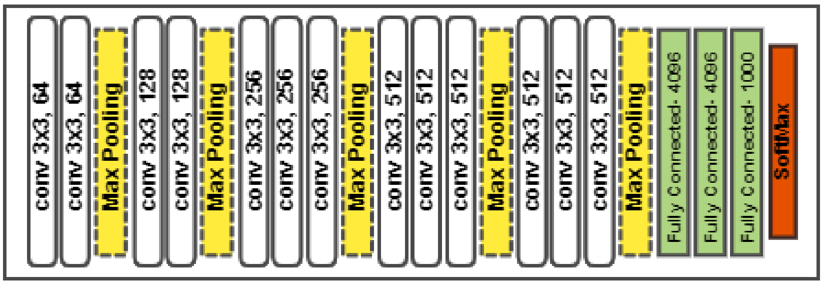


Figure 1. VGG16 architecture

Architectures successful in image classification such as VGG16 have currently learned at a good level smaller features such as lines and curves as the basic components of an image. For this reason, using the features of a model already proven to be succesful in the newly designed model instead of training it from scratch has a significant effect to increase the model's success rate [20]. An approach similar to this situation expressed as transfer learning was used in the SSD architecture and the structure in the VGG16 architecture has formed the backbone of the SSD architecture.

As seen in Figure 2, since the fullyconected layers used in VGG16 architecture have become redundant in the SSD architecture, last fully connected layer has totally not been included into the model and the first two fully connected layers have been reworked in the conv6 and conv7 layers.

Convolution layers outside VGG16 are expressed as extra feature layers in the SSD architecture. Stride was decreased from 2 to 1 during the pooling process residing at the 5th layer of the VGG16 architecture and filter size was determined as 3x3 instead of 2x2. With this process, prevention of the sizes of the featuremaps to be decreased by half at the previous convolution layer was aimed.

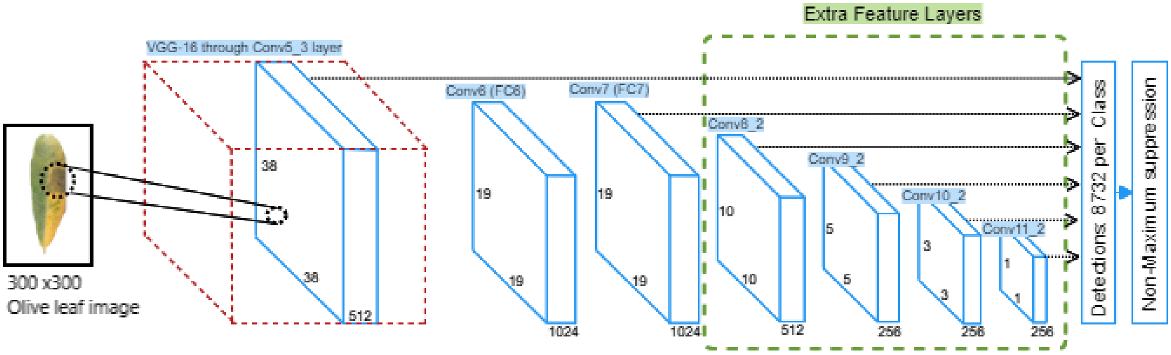


Figure 2. SSD architecture

3. Material and Method

Data set creation constitutes the first stage of the processes of the recognition of olive peacock spot disease found in olive leaf using the SSD architecture. Samples obtained were prepared being labelled under controlled conditions with assistance received from the experts. Then the process of coding some important stages in the SSD model where samples in the data set were trained was initiated. In the first place, default boxes were created on images. Then the functions required to match ground-truth objects with the default boxes were created. In the study, the model that was set up was trained with threshold values ranging between 0.1 and 0.9 for IoU and best AP values were recorded pursuing the trainings. At the last stage, a desk top application of the study was developed in order to facilitiate access of the trained final model for different users. All of the stages of the study were examined in detail in the following subsections.

3.1.Dataset Description

In this study, an open source data set created by Uğuz and Uysal [21] was used. In their study, a total of 1460 images were collected from olive groves in Aegean region of Turkey, in order to deal with the object detection problem directed at identification of the olive peacock spot disease, a common olive disease in Turkey, on the olive leaves. All of the images of the olive leaves were produced under controlled conditions by being photographed on a mono color. In this way, model's processing speed and accuracy rating were aimed to be increased by making the model not to include objects irrelevant with the study into the feature map. Later, olive peacock spot areas on olive leaves were labeled with a program called "labeling" with the assistance of an agricultural engineer expert in the field. Some of the leaf images in the data set are depicted in Figure 3.

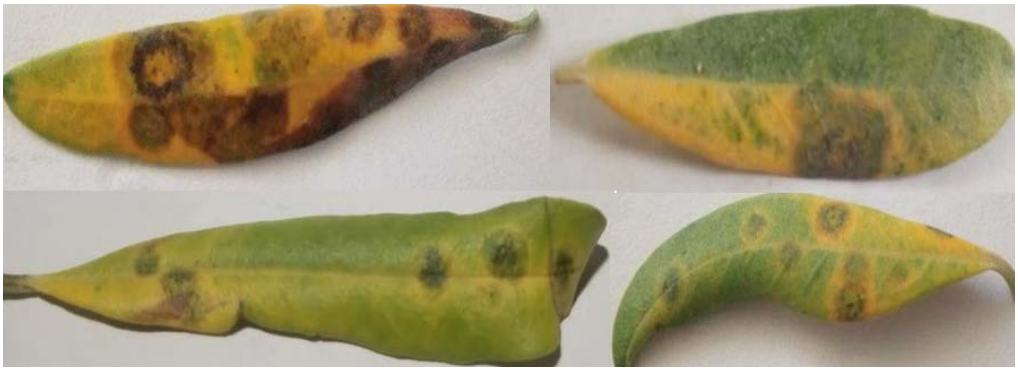


Figure 3. Sample images in the data set

3.2. Creation of the Default Boxes on Images

In the classification problems, which class an image belongs to is determined in accordance with the result predicted by the network. For example, if the images in the data set are divided into two groups as those having olive peacock spot disease and those not having olive peacock spot disease, then the result predicted by the network will fall into one of these two classes. But object detection problems entail a more complex task to be fulfilled. Because, in these problems, there could be multiple objects residing on an image. Along with the determination of the locations of these objects with the help of bounding boxes, which class the object lying inside the bounding box belongs to should also be predicted.

Each convolution layer in SSD architecture consists of different feature map dimensions the largest of which is 38x38 belonging to conv4_3. Default boxes with different aspect ratios and sizes were created at each convolution layer in order to ensure best selection of the objects residing on an image. In Figure 4, conv8_2 layer that has 10x10 feauture map dimension is depicted. For each of the 100 locations, 6 default boxes having different aspect ratios and sizes were created. Thus, a total of 600 default boxes are produced for only Conv8_2. Center coordinates of the default box are denoted by c_x and c_y while w and h represent the width and height of the default box respectively. The dimensions of the default boxes in each location can be calculated using the expressions in Figure 4 where $k = 1, \ldots, K$, S_k and a_r denote kth feature layer, scale of the default boxes and aspect ratio respectively. The number of default boxes produced on all of the convolution layers for SSD 300 turns out to be 8732 as stated in Figure 2.

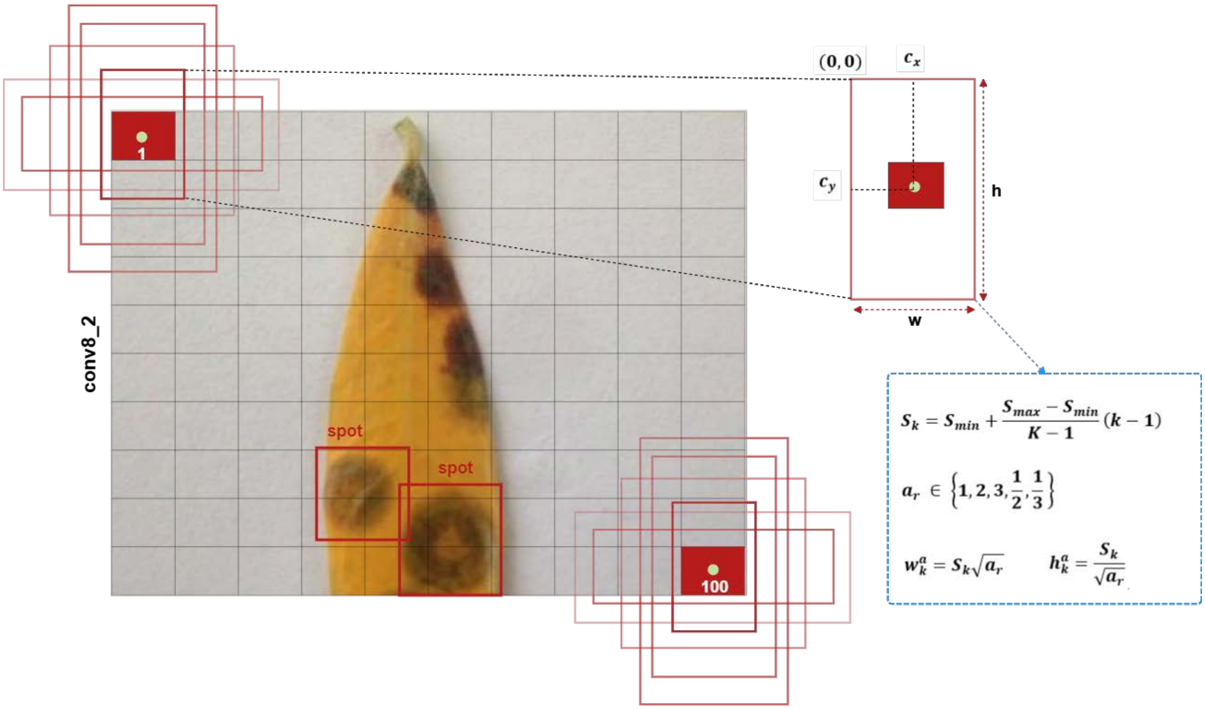

1
conv8_2
spot
spot
100
$(0,0)$
c_x
c_y
h
w
$S_k = S_{min} + \frac{S_{max} - S_{min}}{K - 1}(k - 1)$
$a_r \in \left\{1, 2, 3, \frac{1}{2}, \frac{1}{3}\right\}$
$w_k^a = S_k\sqrt{a_r}$
$h_k^a = \frac{S_k}{\sqrt{a_r}}$

Figure 4. Creation of the default boxes

3.3. Matching Default Boxes with Ground-Truth Object

At this stage of the study, matching of the default boxes at each location of the convolution layers to the ground-truth object is carried out. Matching process is done in accordance with the result of the operation defined as IoU (intersection over union) or Jaccard Index. As observed in the expression in Figure 5, IoU is calculated as the ratio of the intersection of the ground truth and default box to their union. IoU value ranges between 0 and 1 and IoU=1 means a perfect overlap of the boxes [7]. The ground truth object expressed with A in Figure 5 is the labeled form of the olive peacock spot symptom. B, C, and D on the other hand represent 3 defaultboxes generated in the convolution layers.

For instance, while the IoU ratio to be calculated for A and B is seen to take a value close to 1, the IoU ratio to be calculated for A and C or for A and D is seen to take a value close to zero. While object recognition problems are dealt with in SSD, default boxes lower than the 0.5 threshold value are expressed as negative matching and ignored in the classfication predictions to decrease computational cost. Therefore, only the default boxes lying above the 0.5 threshold value and hence expressed as positive matching are used in classification predictions. In this case, only the default box B is going to be used in classification prediction.

In this study, classification performances were compared not only with respect to the 0.5 threshold value but also by conducting experiments with other threshold values as well. This comparison is examined in detal in Section 4.

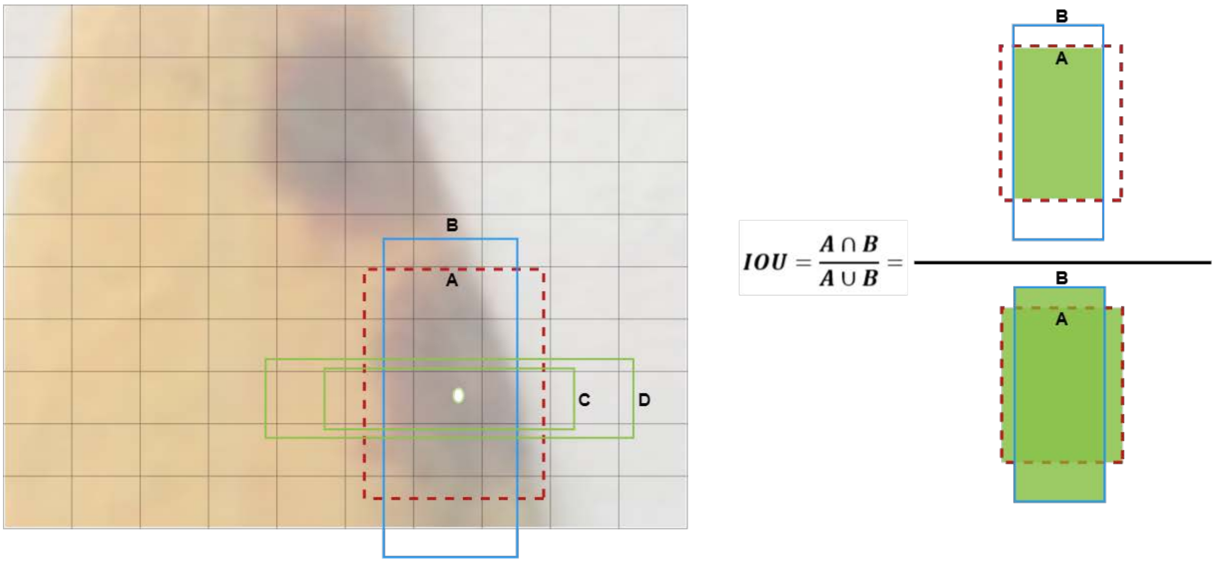


Figure 5. Matching of default boxes with ground-truth

3.4.Training Objective

In the object detection with SSD, localization loss between ground truth and default box and confidence loss belonging to the object lying inside default box take an important place. While the objective function is written considering only the positive matchings, weighted sum of these two loss criteria are taken into account. Expression x_{ij}^{p} denotes jth ground truth that matches ith default box of class p . $x_{ij}^{p} = 1$ indicates a match , while $x_{ij}^{p} = 0$ on the other hand indicates that there is no match. As observed in the localization loss function in Equation 1, score of false match between ground-truth (g) and default box (l) is computed using smooth L1 loss function. Thus, the offsets are regressed to find the center (cx, cy) of the default box (d) along with its width (w) and height (h) [22].

$$L_{loc}(x,l,g) = \sum_{i \in Pos}^{N} \sum_{m \in \{cx,cy,w,h\}} x_{ij}^{k} Smooth_{L1}\left(l_i^m - \hat{g}_j^m\right) \tag{1}$$

$$where \begin{cases} \hat{g}_j^{cx} = \dfrac{\left(g_j^{cx} - d_i^{cx}\right)}{d_i^w}, \hat{g}_j^{cy} = \dfrac{\left(g_j^{cy} - d_i^{cy}\right)}{d_i^h} \\ \hat{g}_j^{w} = \log\left(\dfrac{g_j^w}{d_i^w}\right) \\ \hat{g}_j^{h} = \log\left(\dfrac{g_j^h}{d_i^h}\right) \end{cases}$$

$$and\ Smooth_{L1} = \begin{cases} 0.5\,x^2, & if\ |x| < 1 \\ |x| - 0.5, & otherwise \end{cases}$$

With N denoting the number of default boxes that matched, confidence loss expression which is used in the class prediction and calculated as soft max loss can be written as in Equation 2.

$$L_{conf}(x,c) = -\sum_{i\in Pos}^{N} x_{ij}^{p} \log\left(\hat{c}_i^p\right) \tag{2}$$

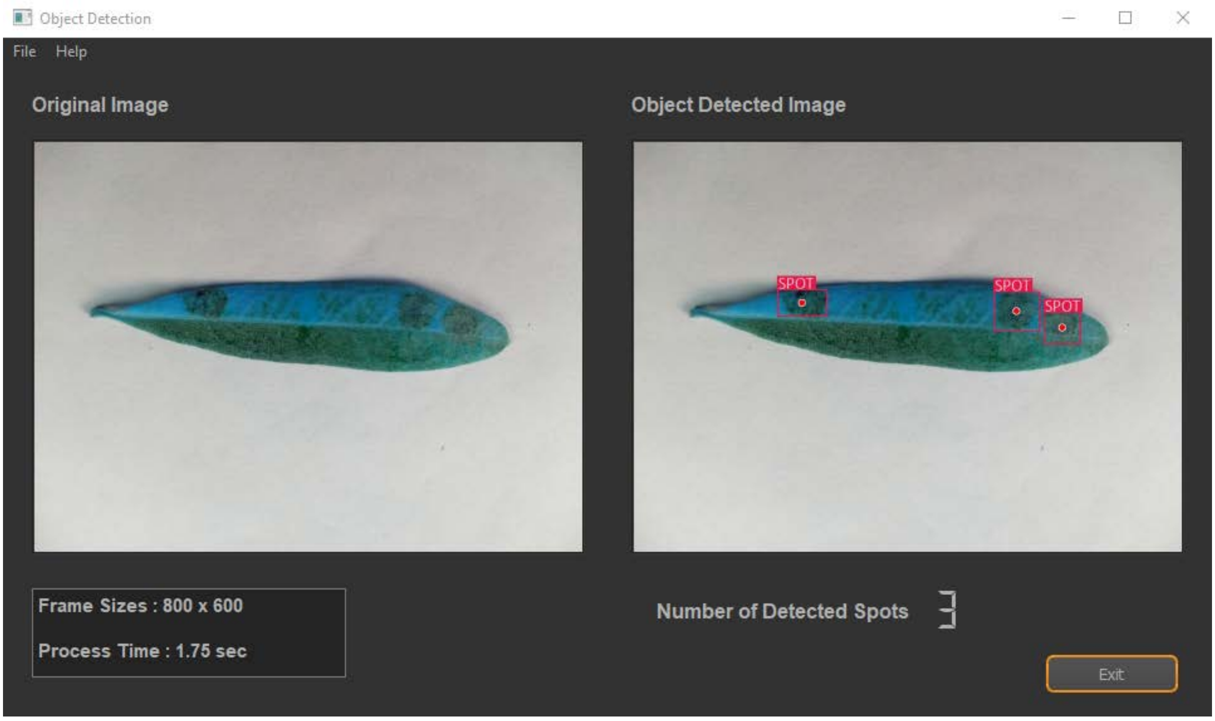
$$where\ \ \hat{c}_i^p = \frac{exp\left(c_i^p\right)}{\sum_p exp\left(c_i^p\right)}$$

After L_{loc} and L_{conf} are defined final objective function for SSD can be expressed as in Equation 3. The parameter α in the equation is the weight value used in balancing the contribution of localization loss.

$$L(x,c,l,g) = \frac{1}{N}\left(L_{conf}(x,c) + \alpha\, L_{loc}(x,l,g)\right) \tag{3}$$

4. Findings and Discussion

Trainings in the automatic disease identification system developed using the SSD architecture were performed by employing Pytorch 1.3 library. With respect to hardware, Google Colab cloud service was utilized during the model trainings. Model trainings were performed using Tesla K80 GPUs. In the studies focusing on plant diseases, the number of applications realized either developing desktop, mobile or web based applications are observed to be quite inadeqaute. Dissemination of the realized application as a packaged software would undoubtedly facilitate a much better utilization for the olive producers. Subsequently, in this study, a GUI was developed for the SSD based application using PyQt5 which is one of Pyhton's libraries. In the software screen seen in Figure 6, when the images registered in the system are selected from the File menu, Original Image is opened. Simultaneously, in the Object Detected Image section, peacock spots detected by the model trained on the image are displayed. Process time elapsed and the number of detected symptoms can be seen on the screen as well.



Frame Sizes : 800 x 600

Process Time : 1.75 sec

Number of Detected Spots 3

Figure 6. GUI software developed for the detection olive peacock spot disease

For the performance evaluation of object detection problems, it is observed that different performance evaluation criteria are used in challenges such as PASCAL VOC, ImageNet and COCO. In this study, AP (average precision) score which is commonly used in the literature [6,11,23] as a performance evaluation criterion in the detection of plant diseases has been employed.

The probability of an object to be inside of a default box is expressed by confidence score. Each default box on the image is marked as TP or FP. Then, the precision and recall values of each are calculated as ordered according to the confidence scores. The percentage of correctly predicted ones among the detected objects is obtained with the precision criterion seen in Equation 4. Recall in Equation 5 gives the percentage of correctly detected ground truths among all of the ground truths [24].

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

The behavior of a model can be adjusted trying different threshold values for the confidence score. While Precision-Recall curve provides information on a single model's performance, performances of multiple object detectors can also be compared by creating many precision call curves on the same graph. However, in this case, since the model curves can mesh on each other, AP (Average Precision) score which helps performance to be expressed quantitatively could be used.

Maximum precision scores obtained for each $r' \geq r$ where r represents 11 points between 0 and 1 were chosen as recall (r) values of the precision recall curve in PASCAL VOC 2008 challenge. Then, average of these presicion scores are taken to calculate AP score (Equation 6).

$$p_{interp}(r) = \max_{r' \geq r} p(r') \tag{6}$$

For PASCAL VOC 2010 and later challenge, just as in the previous formulation, maximum precision scores obtained for each $r' \geq r$ were chosen for the precision recall curve. However, the distinction is that all unique r values are treated in the formulation.

$$AP = \sum_{i=1}^{n-1} (r_{i+1} - r_i)\, p_{interp}(r_{i+1}) \tag{7}$$

$$where \;\; p_{interp}(r_{i+1}) = \max_{r' \geq r_{i+1}} p(r')$$

AP value provides performance information for only one class. Therefore, performance can be measured by calculating mAP which is the average of all AP values for all of the classes in the model. Nevertheless, since object detection for only one disease type was carried out in this study, performance evaluation was not performed using only AP score. IoU threshold values in the study were determined so as to remain between 0.1 and 0.9 and experiments encompassing the trainings for each value were performed.

In Figure 7, IoU values obtained for each threshold value used are seen. When the graph is examined, it is observed that overlapping ratios of manually labelled symptoms (ground truths) of the olive peacock spot disease and the default boxes and accuracy ratio for the correct localization of the symptoms are observed to increase as the chosen IoU value increases. A downward trend is observed in the matchings of groundtruths and defaultboxes as the IoU assumes values higher than 0.5. This decline also indicates that the number of erroneously classified olive peacock spot disease syptoms grow larger as well. AP score reaches 96 % when IoU = 0.5. It is believed that the study has made a significant contribution to literature as the model trained with SSD architecture detected syptoms of the olive peacock spot disease at sufficiently high precision scores.

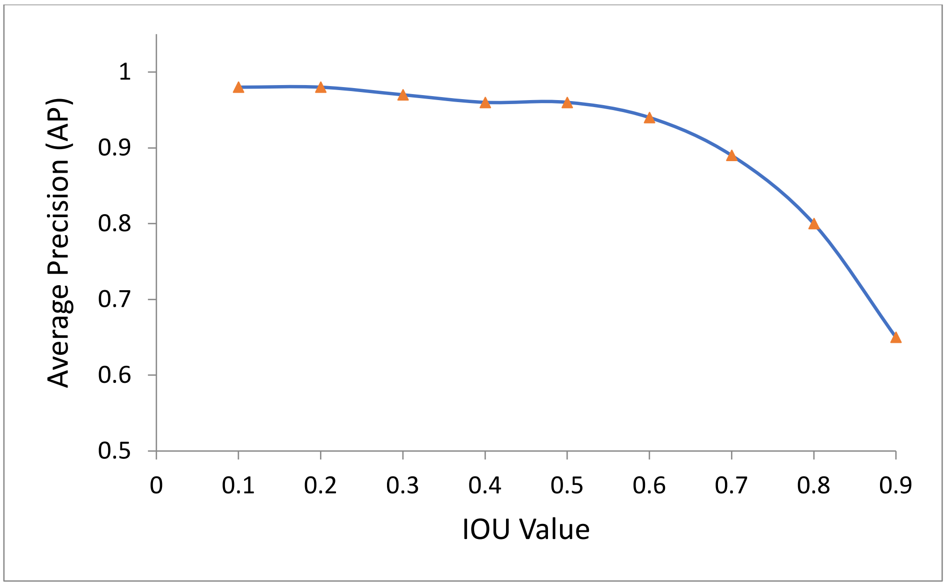


Figure 7. AP results obtained for different IoU values

It is observed that there is quite limited number of studies on the detection of olive plants diseases dealt with in the study. This study differs from the studies of Cruz, et al. [15] and Alruwaili, et al. [16] with regards to the type of the disease detected and the CNN architectures employed. On of the diseases Alruwaili, et al. [16] endeavoured to detect was olive peacock spot. But their study was on object classification not on object detection. In the studies of the authors, a 90.2% classification precision score was obtained. Another study carried out by Cruz, et al. [15] on olive plant diseases also focused on the subject of classification. Their study diverges from this study concerning the type of the detected disease and the development of a web-based application instead of a desktop application.

5. Results and Proposals

In this study an application was developed for the detection of the symptoms of the olive peacock spot, a disease widely seen in Turkey on olive plants, using Single Shot Detector which is one of the popular deep learning architectures. A new data set was created on our own means by collecting 1460 images under controlled conditions from the olive groves of the Aegean region in Turkey. In the study directed at object detection, trainings were performed for different IoU values in Single Shot Detector architecture which produces fast detection results especially for real time applications and the results were discussed using the average precision evaluation criterion. Accordingly, a 96 % AP value was obtained for IoU=0.5. A downward trend was observed for values higher than this threshold value.

In order to make the study available for the stakeholders as well, a desk top application was developed. In this way, people who selected a leaf image on the application had the chance of seeing both the disease symptoms and also the process time.

At later stages, alternatives to turn the developed software into a web-based or a mobile application can be considered. In this study, images were obtained under controlled conditions to get better results in terms of speed. However, especially those researchers interested in developing a real-time mobile application have to conduct their works on uncontrolled images. Moreover, in this case, the number of images that should be processed in the trainings should be much higher.

The study was conducted on the detection of only one disease. Among the disease types of the olive plant, various diseases seen on the fruit, leaves or branches such as Anthracnose, Canker, Lepra Fruit Rot, Parlatoria Oleae and Aspidiotus Nerii can also examined in the studies. However, since some diseases can be specific to some regions, obtaining them might not be an easy task. Hence, the crucial point is to be able to obtain the images belonging to those diseases. For this reason, every study on the detection of plant diseases and every original data set are considered valuable.

Success rate can be expected to decrease as the number of the diseases wanted to be detected in the experiments conducted for the same type of plant is increased. Because, it is possible for the diseases of the same type of plant to exhibit symptoms similar to each other. Due to lower probabilities of succes, future studies to be conducted on this area are considered valuable.

Just as in human health, early diagnosis of the diseases of the agricultural products during their growing stage is very important. Detection of a disease at an early stage may help the farmers take the appropriate measures and thus reduse the associated costs. In the literature survey conducted, it is seen that applications directed at detection, as in this study, were developed after the disease occurred. It is considered that it would be more effective if the new research studies were designed according to the stages of the disease.

References

[1] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "Deepfruits: A fruit detection system using deep neural networks," *Sensors,* vol. 16, p. 1222, 2016.

[2] J. Lu, J. Hu, G. Zhao, F. Mei, and C. Zhang, "An in-field automatic wheat disease diagnosis system," *Computers and electronics in agriculture,* vol. 142, pp. 369-379, 2017.

[3] O. Apolo-Apolo, J. Martínez-Guanter, G. Egea, P. Raja, and M. Pérez-Ruiz, "Deep learning techniques for estimation of the yield and size of citrus fruits using a UAV," *European Journal of Agronomy,* vol. 115, p. 126030, 2020.

[4] M. Kerkech, A. Hafiane, and R. Canals, "Deep leaning approach with colorimetric spaces and vegetation indices for vine diseases detection in UAV images," *Computers and electronics in agriculture,* vol. 155, pp. 237-243, 2018.

[5] M. M. Ozguven and K. Adem, "Automatic detection and classification of leaf spot disease in sugar beet using deep learning algorithms," *Physica A: Statistical Mechanics and its Applications,* vol. 535, p. 122537, 2019.

[6] M. G. Selvaraj, A. Vergara, H. Ruiz, N. Safari, S. Elayabalan, W. Ocimati*, et al.*, "AI-powered banana diseases and pest detection," *Plant Methods,* vol. 15, p. 92, 2019.

[7] W. Li, P. Chen, B. Wang, and C. Xie, "Automatic localization and count of agricultural crop pests based on an improved deep learning pipeline," *Scientific reports,* vol. 9, pp. 1-11, 2019.

[8] A. Fuentes, S. Yoon, S. C. Kim, and D. S. Park, "A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition," *Sensors,* vol. 17, p. 2022, 2017.

[9] G. Polder, N. van de Westeringh, J. Kool, H. A. Khan, G. Kootstra, and A. Nieuwenhuizen, "Automatic Detection of Tulip Breaking Virus (TBV) Using a Deep Convolutional Neural Network," *IFAC-PapersOnLine,* vol. 52, pp. 12-17, 2019.

[10] D. Rong, L. Xie, and Y. Ying, "Computer vision detection of foreign objects in walnuts using deep learning," *Computers and Electronics in Agriculture,* vol. 162, pp. 1001-1010, 2019.

[11] P. V. Bhatt, S. Sarangi, and S. Pappula, "Detection of diseases and pests on images captured in uncontrolled conditions from tea plantations," in *Autonomous Air and Ground Sensing Systems for Agricultural Optimization and Phenotyping IV,* 2019, p. 1100808.

[12] B. A. Ashqar and S. S. Abu-Naser, "Image-based tomato leaves diseases detection using deep learning," *International Journal of Academic Engineering Research,* vol. 2, pp. 10-16, 2018.

[13] C. IO. (2018, 13 December 2019). *World oil production*. Available: https://www.internationaloliveoil.org/what-we-do/economic-affairs-promotion-unit/

[14] F. O. Obanor, M. V. Jaspers, E. E. Jones, and M. Walter, "Greenhouse and field evaluation of fungicides for control of olive leaf spot in New Zealand," *Crop Protection,* vol. 27, pp. 1335-1342, 2008.

[15] A. C. Cruz, A. Luvisi, L. De Bellis, and Y. Ampatzidis, "X-FIDO: An effective application for detecting olive quick decline syndrome with deep learning and data fusion," *Frontiers in plant science,* vol. 8, p. 1741, 2017.

[16] M. Alruwaili, S. Alanazi, S. A. El-Ghany, and A. Shehab, "An Efficient Deep Learning Model for Olive Diseases Detection," *International Journal of Advanced Computer Science and Applications,* vol. 10, 2019.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91-99.

[18] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, *et al.*, "Ssd: Single shot multibox detector," in *European conference on computer vision*, 2016, pp. 21-37.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556,* 2014.

[20] A. Kaya, A. S. Keceli, C. Catal, H. Y. Yalic, H. Temucin, and B. Tekinerdogan, "Analysis of transfer learning for deep neural network based plant classification models," *Computers and electronics in agriculture,* vol. 158, pp. 20-29, 2019.

[21] S. Uğuz and N. Uysal, "Classification of olive leaf diseases using deep convolutional neural networks", *Neural Computing and Applications*, https://doi.org/10.1007/s00521-020-05235-5.

[22] S. Cao, D. Zhao, X. Liu, and Y. Sun, "Real-time robust detector for underwater live crabs based on deep learning," *Computers and Electronics in Agriculture,* vol. 172, p. 105339, 2020.

[23] A. Ramcharan, P. McCloskey, K. Baranowski, N. Mbilinyi, L. Mrisho, M. Ndalahwa, *et al.,* "A mobile-based deep learning model for cassava disease diagnosis," *Frontiers in plant science,* vol. 10, p. 272, 2019.

[24] S. Uğuz, *Makine Öğrenmesi Teorik Yönleri ve Python Uygulamaları ile Bir Yapay Zeka Ekolü*: Nobel Akademik Yayıncılık, 2019.