



Apple Ripeness Identification Using Deep Learning

Bingjie Xiao^(✉), Minh Nguyen^(✉), and Wei Qi Yan^(✉)

Auckland University of Technology, Auckland 1010, New Zealand
vty0884@autuni.ac.nz, {mnguyen, wyan}@aut.ac.nz

Abstract. Deep learning models assist us in fruit classification, which allow us to use digital images from cameras to classify a fruit and find its class of ripeness automatically. Apple ripeness classification is a problem in computer vision and deep learning for pattern classification. In this paper, the ripeness of apples in digital images will be classified by using convolutional neural networks (CNN or ConvNets) in deep learning. The goal of this project is to verify the capability of deep learning models for fruit classification so as to lessen our human labor. Our experiments consist of four parts, namely, image preprocessing, apple detection, ripeness classification, and resultant evaluations. The contribution of this paper is that the classifiers are able to achieve the best result, i.e., the ripeness class of an apple from a given digital image is able to be precisely predicted. We have optimized the deep learning models and trained the classifiers so as to achieve the best outcome.

Keywords: Ground truth · Apple ripeness classification · YOLO

1 Introduction

Deep learning is a vital method for modern computer vision [24, 34, 38, 53]. Deep learning, namely, deep neural networks classify semantic objects and export the output of pattern classification [15, 19, 27–29, 51, 58, 59], which has shown its super capability to outperform than our human visual system.

There are existing software products for fruit classification which achieve very fast speed and high accuracy. Different from the existing classifiers, in our experiments, our attention is paid to the classification of a single fruit. However, the purpose of this paper is to identify the ripeness of apples. Apples are basically grouped into three categories, i.e., unripeness, ripeness, and overripeness. The key of this project is to detect visual objects and figure out the best methods for apple classification. Based on the requirement of this research question, the corresponding experiment firstly needs to locate the apples in an image, then classify them and label them.

The motivation of this project lies in the classification of fruit ripeness based on deep neural networks [1, 3, 9, 12, 66, 67]. A myriad of computational models has been exploited to classify the class of fruit ripeness [37]. The trends of fruit ripeness classification have been surveyed. Based on the obtained knowledge, a method with the

capability of classifying fruits owing to an effective neural network has been proffered. Non-maximum suppression is propounded to achieve object detection [2, 55]. All the detected bounding boxes were sorted by using scores. A predefined threshold was taken to suppress the overlapping with the detected boundary box that achieves the maximum score. The process is based on a regression for all bounding boxes.

In this project, a soft-NMS continuous function is adopted for calculating the scores of all visual objects. The soft-NMS function enhances the mean average precision (mAP) by modifying the NMS algorithm, no parameters is required to be adjusted.

In this paper, digital images captured by using mobile phones are primarily labelled in three classes (unripeness, ripeness, and overripeness). The core work of this project is to classify the visual objects and develop optimal methods. Given the requirement analysis, the corresponding implementations should be designed for the purposes of locating the apples in an image, extracting visual features, classifying the apple ripeness using the labelled data.

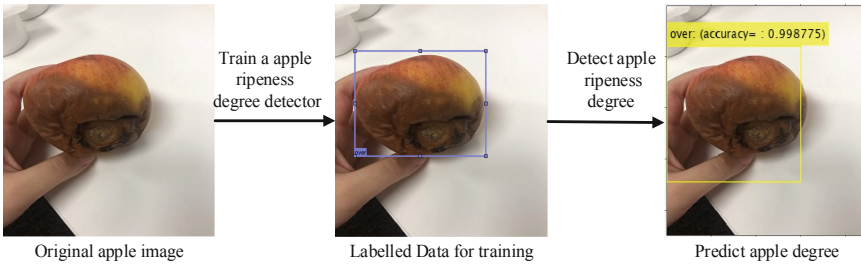


Fig. 1. Apple ripeness classification using supervised learning.

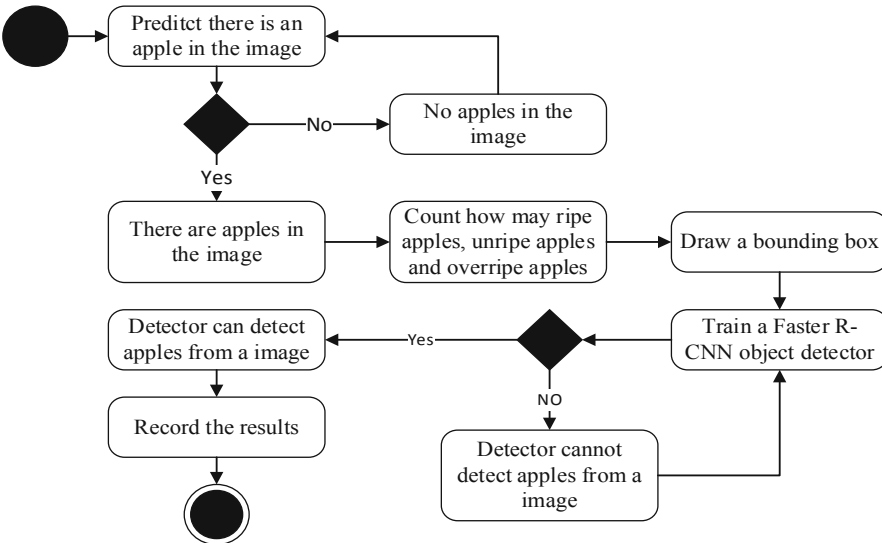


Fig. 2. The workflow of apple ripeness classification.

In Fig. 1, we detect an apple in a given image successfully, we thus draw a rectangle as the bounding box around it. This often involves two operations: (1) Predicting a class of apple ripeness (ripeness, unripeness, or overripeness); (2) Drawing a bounding box around the detected apple. After all apples are detected [46], we are use of a neural network as the classifier for apple classification.

The procedure is shown in Fig. 2. In Fig. 2, we manually label the location and ripeness class of the apples in each image as the labels and import the labelled data into the selected deep learning models for training. Then, we obtain a well-trained classifier associated with parameters (e.g., weights, network layout, etc.) that is able to classify the ripeness of the given apples [47].

The remaining parts of this paper are organized as follows. In Sect. 2, we review the literature. In Sect. 3, we elucidate our methodology. In Sect. 4, we explicit our experimental results. Our conclusion is drawn in Sect. 5, meanwhile the future work will be envisioned.

2 Related Work

A rich assortment of methods has been applied to apple classification. The first method was employed to find the edges, corners, colors, these visual features were extracted from digital images to classify visual objects. Although our human brain is proficient in classifying visual objects through our eyes, how the visual features take effect in the classification is still remaining unclear. Conventional machine learning methods have often been fulfilled directly by applying a broad spectrum of visual features from digital images [20, 25, 26, 30, 31, 41, 48, 49, 62, 64]. In fact, through long-time observations, experiments and evolutions, machine learning algorithms are subsequently employed to classify visual objects in the given images based on these visual features.

Feature extraction is the fundamental problem of pattern classification. Visual features usually encapsulate color, histogram, texture, edge, corners, motif, parts, etc. [10, 43, 50, 54, 62, 64, 65]. The extracted visual features will assist apple ripeness classification by exploiting deep neural networks [23, 65].

Besides feature extraction, artificial neural networks, especially deep neural networks, are trained by using the labelled images. A deep neural network has multiple layers of neurons in the end-to-end paradigm, in which each layer is connected to the next layer. The deep learning models are trained, tested, and validated by using the visual features extracted from the input images.

The existing work for visual object classification falls into CNN models and other machine learning methods [52, 62, 64, 65]. Deep learning methods are categorized into two groups: CNNs and RNNs [4, 5, 11, 13, 16, 20, 42]. In our surveys, R-CNN and Fast R-CNN are slower than YOLO and SSD in training, while the latter is proved to be with lower accuracy [6–8, 21, 22, 32, 33, 39, 40, 56]. Hence, training a classifier based on Faster R-CNN to conduct apple ripeness classification is regarded as an optimal method.

Based on Faster R-CNN, a hybrid detector was explicated for partially occluded object detection [21, 22]. The work emphasized on network depth. However, even if a network with the depth from sixteen to thirty is employed, the accuracy would not be further uplifted. Exploding gradient problem hampers the convergence at the beginning

of the training processing. Normalization can assist the convergence of deep neural networks by using stochastic gradient descent (SGD) for backpropagation. With the depth growth of the neural network, degradation problem occurs, the accuracy is down-regulated. Trimming the network layers cannot dramatically affect the accuracy of deep learning models. In contrast, excessive network layers lead to accuracy saturation.

A method using Fruits-360 dataset has been propounded for fruit recognition [18]. By taking advantage of the depth of convolution neural networks, the cost of computations is diminished, while the robustness is enhanced, the performance of deep learning models is prone to be better.

The low-quality depth map with multilevel features is originated from RGB images [61]. Color images may enclose abundant appearance information. A novel RGBD saliency model based on bottom-up module explores the color images and depth information. Taken the abundant information into consideration, the module embeds attention models associated with salient objects.

The collection of visual data and deep learning models are key parts of our experiments. In this paper, we choose ResNet-50 and GoogLeNet as our models for apple detection, Faster-RCNN for apple ripeness classification. We also apply YOLO, one of DarkNet models, to compare the accuracy and training time.

ResNet-50 is a residual neural network, which has 50 layers. There are four groups of blocks in ResNet-50, three convolutional layers in each block. Ideally, the accuracy raises with the growth of the number of the network layers.

GoogLeNet is a 22-layer network with the structure of inception network, average pooling layer replaces the fully connected layers in the end of the network. To avoid the gradient vanishing, the network additionally adds two auxiliary softmax layers for the forward gradient. The model is better for parallel computing.

GoogLeNet takes use of Inception model. Inception v3 model has two branches, each branch generates one output. We harness a branch of the Inception v3 model to calculate convolution features.

Faster R-CNN refers to region-based CNN model which exhibits high efficiency and accuracy [35]. Fast R-CNN associated with feature maps by using selective search method for object detection. The loss function exploits regression during the training process and then optimizes the gained results. Faster R-CNN method inherits the attributes of Fast R-CNN. The critical change is the mechanism of object position prediction because the selective search method of Fast R-CNN has too much computational redundancy.

Visual data collected from real world could be blurry, rotated, or jitter with noises [33, 68]. Noises and invalid data may affect the result of object detection. In this project, we create an image degradation model based on YOLO. A mathematical model is established to generate degraded images which are dominantly applied to train YOLO object detector.

Compared to R-CNN model, YOLO networks have lower accuracy for identifying the position of a visual object. The prediction of bounding boxes in each grid cuts down the times of the object detection. The region proposal method has curbed overlapping. Furthermore, YOLO model adjusts the network structure, employs multiscale features for object detection, and replaces softmax function with logistic function for object classification. In terms of visual feature extraction, YOLO net harnesses a network structure DarkNet-53 (i.e., 53 convolutional layers), which is better than the residual networks.

With competitive accuracy rates, YOLOv3 is faster than other models. Furthermore, YOLOv4 has the attainment in both speed and accuracy of computations.

3 Our Approaches

Different from the existing approaches, in this project, we will work for apple ripeness classification using deep learning. The workflow is shown in Fig. 3. Our work is different from the existing one, we implement YOLO models for the purpose of comparisons. The reason is that the focus of our experiments is on practical applications. Fig.3 intuitively assists us to understand the experimental process using Faster R-CNN. As one of object detection methods, Faster R-CNN firstly harnesses a set of basic layers including convolutional layers, revolution layers, and pooling layers to extract image feature maps. The feature maps are shared for the subsequent RPN layer and fully connected layer. The RPN network is used to generate region proposals. This layer judges that the anchors belong to the foreground or background through softmax layers, then takes use of regression algorithms to correct the anchor boxes so as to obtain the accurate proposals. This layer collects the input feature maps and proposals, extracts the proposal feature maps after integrating the information, and sends them to the subsequent fully connected layer to classify the target category. We take use of proposal feature maps to calculate the category of apples, bounding box regression to obtain the final precise position of the detected object.

In this project, we not only determine whether there is an apple in an image, but also mark the location of this apple. Locating an apple implies to determine the specific location of the apple in an image. There are three types of bounding boxes in Fig. 4, ground truth box (blue), anchor box (green), and predicted box (yellow) which represent the labelled location, the detected location, and the predicted location, respectively [57].

The ground truth boxes are labelled manually, which reveal the real locations of apples in the given image. Naturally, the images in the training dataset all have been labelled. An anchor box is the sliding window which traverses through the image. The anchor box describes how Faster R-CNN model extracts features from the given images [6, 7]. An image cell may belong to multiple anchor boxes. The input is the whole image and the output is the labels.

Usually, anchor box in an image is not generated by a training dataset, which usually is distributed across the image. The essence of this kind of anchor boxes is the sliding window which traverses the image. A cell will correspond to multiple anchor boxes. The quantity of anchor box is set at the initialization stage. For instance, if we cut an image into 3×3 grids and set 2 anchor boxes, those two boxes will go through all the grids and make predications.

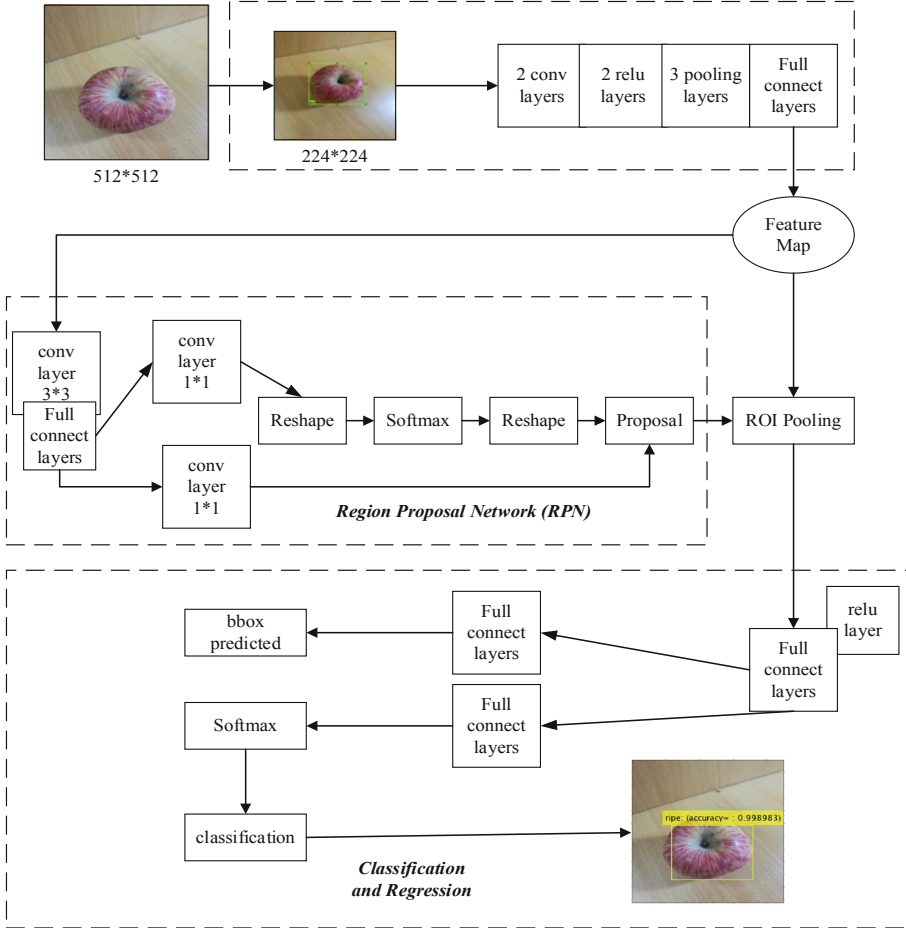


Fig. 3. Apple ripeness classification using Faster R-CNN

In our experiment, c_1 , c_2 , and c_3 represent three classes of apple ripeness. The bounding box is located by using four corner points. We denote the width of the bounding box as w , the height is h . In Fig. 5, i , j , and k represent the subscripts of bounding boxes, respectively. The vector $\mathbf{T}$ is

$$\mathbf{T} = (x, y, z, w, h, c_1, c_2, c_3). \quad (1)$$

The proposed model is to calculate the regression of bounding boxes $R_i(\cdot)$ that approaches to the ground truth $R_g(\cdot)$ iteratively,

$$R_1(A_1, B_1, C_1, D_1) \rightarrow \cdots \rightarrow R_i(A_i, B_i, C_i, D_i) \rightarrow \cdots \rightarrow R_g(A_g, B_g, C_g, D_g), i = 1, 2, \quad (2)$$

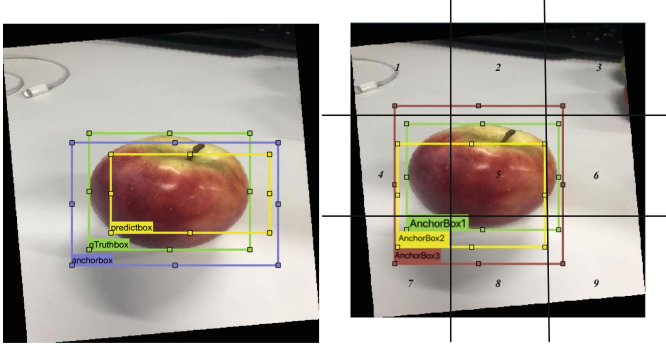


Fig. 4. The sample of bounding boxes.

The predicted bounding boxes are generated based on the regression of anchor boxes. Non-maximum suppression (NMS) allows each box to retain only one predicted bounding box and export all the probabilities from the highest to the lowest one. Each box only has the one with the maximum probability, the offsets are set as

$$\Delta x = (x_i - x_k)/w_k \quad (3)$$

$$\Delta y = (y_i - y_k)/h_k \quad (4)$$

$$\Delta w = \log(w_i/w_k) \quad (5)$$

$$\Delta h = \log(h_i/h_k) \quad (6)$$

where (x_k, y_k) denotes the center point of the predicted bounding box, h_k and w_k are the width and height, respectively. Δx , Δy , Δw , and Δh are the scaling or zooming factors of the bounding boxes calculated by using the region proposal regression.

In Fig. 5, A_1 , B_1 , C_1 and D_1 are applied to represent the predicted anchor box. x_k and y_k stand for the midpoint of the predicted apple location, h_k refers to the number of vertically divided anchors, w_k indicates the number of horizontally divided anchors, x_i , y_i , w_i and h_i refer to the coordinates, Δx , Δy , Δw and Δh should be the required values calculated by using region proposal. Actually, w_i and h_i may be larger or smaller than w_j and h_j . The loss function makes Δx , Δy , Δw and Δh get the minimum. In Fig. 5, the yellow boxes mark the actual apple position, the blue one is the position predicted.

In this paper, we design our experiments to compare the accuracy of apple classifications by using Faster R-CNN and YOLO models. ResNet-50 and GoogLeNet are chosen as training networks [14, 17]. In order to gain the best apple ripeness classification [6, 7, 9, 57, 61], mini-batch size, learning rate, epoch, quantity of data and quality of data are chosen as the optional parameters.

We train ResNet-50 model by using Dataset I. The mini-batch size was set as 1.000 in apple ripeness classification. Setting the learning rate as 0.001 is appropriate because a lower learning rate leads to an overloaded cost in the training. Dataset I and Dataset II

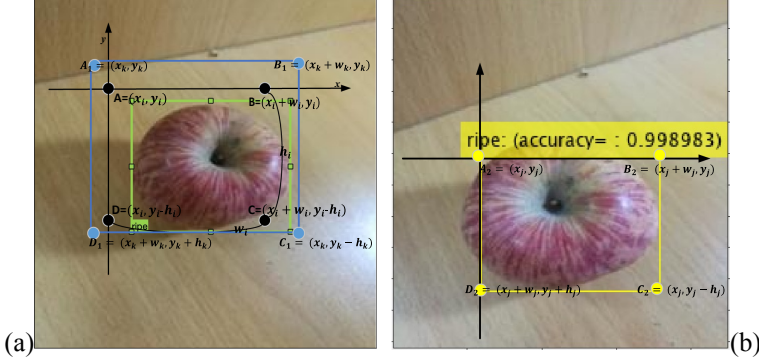


Fig. 5. The ground truth with labels, detected bounding boxes, and anchor boxes.

are selected as random datasets to test whether the number of training images will affect the results or not.

Ripe and overripe apple images in Dataset II have been applied to train ResNet-50 [36] and GoogLeNet. Due to the conditions (e.g., lighting and weather, etc.), the collection of training images have diversity.

The images in Dataset III were resized, no matter enlarged or shrunk. The dataset is scaled down, the image size is normalized to 224×224 from 512×512 . Each image was rotated anticlockwise every 10 degrees as the interval. Additionally, Gaussian noises and salt-and-pepper noises are added to achieve the robustness of deep learning model through data augmentation [44, 45, 60].

Data augmentation is to primarily solve the problem of insufficient number of samples and uplift the quality of classification. We have imported a wealth of data for the model training, we avoid overfitting by focusing on general patterns rather than the specific ones. We use data augmentation to conduct various operations offline, including rotating, cropping, and resizing so as to adjust the amount of input data. Our data collection does not require excessive costs, the data augmentation caters for the demand of raising the volume of visual data.

The size of the neural network usually refers to the number of layers and weight parameters of the network. In this paper, we take use of ResNet model, which contains eighteen blocks, we choose ResNet-50 model with 50 blocks for our experiments.

The idea of average precision (mAP) is considered to find the region of interest (ROI) under precision-recall (PR) graph [57]. AP is computed as the average of maximum precision

$$AP = \frac{1}{N} \sum_k^N P(k) \quad (7)$$

where $P(k)$ is the probability of each ripeness class of an apple. If we need to average the classification accuracy of all the apples in the image, mAP (mean average precision)

is calculated. For each class, we calculate AP and denote the average of all AP values as mAP , which is usually calculated based on one dataset, mAP is thought as a relatively objective and fair metric.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

$$Precision = \frac{TP}{TP + FN} \quad (9)$$

$$Recall = \frac{TP}{TP + FN} \quad (10)$$

where TP (i.e., true positive) is the hit, FP (i.e., false positive) refers to the false alarm, TN (i.e., true negative) means correct rejection, FN (i.e., false negative) stands for miss.

Classification accuracy, no matter which category, as long as the prediction is correct, its number is placed on the numerator, the denominator is the total number of data, which shows that the accuracy is a judgment on all data. The accuracy corresponds to a class in the classification, the numerator is the number of correct prediction of the class, the denominator is the number of all data predicted to be classified.

In our experiments, we use both precision and accuracy method to evaluate the model no matter whether the model can accurately classify apple ripeness or not. From the accuracy, we judge whether a deep learning model is effective and qualifies as a classifier. Besides, precision and recall are offered to assess the performance of the models.

4 Our Experiments

Our three datasets were created with more than 10,000 images as shown in Table 1, where the apples gradually turn to overripe from unripe. The ripe and overripe apples were shot with a mobile phone camera. All of these three datasets are randomly split into training set and test set. Dataset I and Dataset II are mainly to compare the influence of data quantity for our experiments, Dataset III is used to compare the impact of data quality on the experiments after data augmentation.

Our experiment did not use a standard dataset, but manually collected pictures of apples under multiple environments like indoors and outdoors. In the real training, we used MATLAB 2020b as our experimental tool. During the experimentations, we used ordinary CPU computers for training. It takes over 48 h to train the Faster R-CNN model based on the datasets.

We have trained our models for multiple times, the results are listed in Tables 2, 3, 4 and 5. In the first round of training, a dataset was labelled into three classes. The parameters minibatch and epoch were firstly selected for training, the learning rate was set as 0.001.

In Table 3, we clearly see the classification performance of each model. For the class of overripe apples with obvious features, the model shows better performance, which means that the model performs better for this kind of apples.

Table 1. The number of samples of our datasets for this project

Datasets	Ripeness	Overripeness	Unripeness
(a) Number of apple images			
I	144	111	92
II	1325	1083	1040
III	4149	3713	–
(b) Corresponding numbers of labels			
I	552	452	409
II	5416	4475	3976
III	12564	10812	–

Table 2. The results of average precisions of our proposed models

Faster R-CNN		
Epochs	Deep learning models	mAP
10	ResNet-50 (ripe, overripe, Unripeness)	39.33%
10	ResNet-50 (ripe, overripe)	49%
30	ResNet-50 (ripe, overripe, Unripeness)	35.33%
10	GoogLeNet (ripe, overripe)	46.5%
30	GoogLeNet (ripe, overripe)	35.67%

Table 3. The results of precision rates using our datasets

Faster R-CNN model				
Epochs	Deep learning models	Ripeness	Overripeness	Unripeness
10	ResNet-50	13%	93%	12%
10	ResNet-50	50%	48%	–
30	ResNet-50	36%	53%	17%
10	GoogLeNet	40%	53%	–
30	GoogLeNet	32%	54%	21%

Given current mAP listed in Table 2, the results reveal that our model performs better for single class. We take use of 22 layers of GoogLeNet and 50 layers of ResNet-50 for model testing. In the current dataset, the number of network layers does not have a significant impact on the experimental results.

As we expected, by using larger datasets, more rounds of model training are required. But the total steps for convergence should be the same and independent on the size of

Table 4. The accuracy rates by using YOLOv3 model

YOLOv3 model			
Epochs	Deep learning models	Ripeness	Overripeness
10	YOLOv3(DarkNet)	99.96%	99.87%
30	YOLOv3(DarkNet)	84.58%	88.77%

Table 5. The precision rates by using YOLOv4 model

YOLOv4 model				
Epochs	Deep learning models	Ripeness	Overripeness	Unripeness
10	ResNet-50	100%	99.98%	99.93%

the dataset. If the model is further adjusted, early stopping method is demanded or the learning rate needs to be dipped.

The depth of the proposed neural network for apple classification is critical. Nevertheless, if the network gets deeper, the performance cannot be further enriched. The degradation problem occurs whilst the depth of the layers of the network increases. If the network model is sufficiently deep, the precision will get saturated and then degraded. This problem indicates that the depth of deep network does not satisfy the requirement of our experiments. Because the network is too deep, the accuracy of the model cannot be enhanced.

As shown in Table 4, the model is still with high performance. The precision ensures high accuracy. Accuracy is calculated based on all the predicted samples. Compared to the Faster R-CNN model, YOLOv3 and YOLOv4 outperform in apple classification in the binary classification. The YOLOv4 model reflects the overall advantages of the classifier with very high accuracy.

5 Conclusion

In this paper, multiple deep learning models have been applied to apple ripeness classification. We strongly recommend the state-of-the-art YOLO model because it achieves the best performance for the sake of tradeoff between computing speed and accuracy as well as precision and recall. Our future work is to probe the parameters of optimized models. We also need to compare both of the precision and accuracy to diminish the random errors [61, 63].

References

1. Amara, J., Bouaziz, B., Algergawy, A.: A deep learning-based approach for banana leaf diseases classification. In: BTW (Workshops), pp. 79–88 (2017)

2. Bodla, N., Singh, B., Chellappa, R., Davis, L.S.: Soft-NMS-Improving object detection with one line of code. In: IEEE International Conference on Computer Vision, pp. 5561–5569 (2017)
3. Buzzelli, M., Belotti, F., Schettini, R.: Recognition of edible vegetables and fruits for smart home appliances. In: International Conference on Consumer Electronics, Berlin, pp. 1–4 (2018)
4. Byeon, Y.H., Kwak, K.C.: A performance comparison of pedestrian detection using Faster RCNN and ACF. In: International Congress on Advanced Applied Informatics, pp. 858–863 (2017)
5. Cai, Z., Vasconcelos, N.: Cascade R-CNN: delving into high quality object detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 6154–6162 (2018)
6. Cao, C., et al.: An improved faster R-CNN for small object detection. *IEEE Access* **7**, 106838–106846 (2019)
7. Chen, D., Wang, H.: Application on intersection classification algorithm based on clustering analysis. In: Annual Computer Software and Applications Conference, pp. 290–297 (2018)
8. De Rita, N., Aimar, A., Delbruck, T.: CNN-based object detection on low precision hardware: racing car case study. In: IEEE Intelligent Vehicles Symposium (IV), pp. 647–652 (2019)
9. Dias, P.A., Tabb, A., Medeiros, H.: Multispecies fruit flower detection using a refined semantic segmentation network. *IEEE Robot. Autom. Lett.* **3**(4), 3003–3010 (2018)
10. Dong, E., Lu, Y., Du, S.: An improved SSD algorithm and its mobile terminal implementation. In: IEEE International Conference on Mechatronics and Automation, pp. 2319–2324 (2019)
11. Eaton, A. T.: Fruit injury types recognized in annual new hampshire apple harvest evaluations. *Entomology*, **13** (2017)
12. Fachrurrozi, M., Fiqih, A., Saputra, B. R., Algani, R., Primanita, A.: Content based image retrieval for multi-objects fruits recognition using K-means and K-nearest neighbor. In: International Conference on Data and Software Engineering (ICoDSE), pp. 1–6 (2017)
13. Fachantidis, A., Partalas, I., Taylor, M.E., Vlahavas, I.: Transfer learning with probabilistic mapping selection. *Adapt. Behav.* **23**(1), 3–19 (2015)
14. Feng, J., Zeng, L., He, L.: Apple fruit recognition algorithm based on multi-spectral dynamic image analysis. *Sensors* **19**(4), 949 (2019)
15. Gongal, A., Amatya, S., Karkee, M., Zhang, Q., Lewis, K.: Sensors and systems for fruit detection and localization: a review. *Comput. Electron. Agric.* **116**(C), 8–19 (2015)
16. Guo, L., Lei, Y., Xing, S., Yan, T., Li, N.: Deep convolutional transfer learning network: a new method for intelligent fault diagnosis of machines with unlabelled data. *IEEE Trans. Industr. Electron.* **66**(9), 7316–7325 (2018)
17. Huang, Z., Cao, Y., Wang, T.: Transfer learning with efficient convolutional neural networks for fruit recognition. In: IEEE Information Technology, Networking, Electronic and Automation Control Conference, pp. 358–362 (2019)
18. Hou, L., Wu, Q., Sun, Q., Yang, H., Li, P.: Fruit recognition based on convolution neural network. In: The International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, pp.18–22 (2016)
19. Howlader, M.R., Habiba, U., Faisal, R.H., Rahman, M.M.: Automatic recognition of guava leaf diseases using deep convolution neural network. In: International Conference on Electrical, Computer and Communication Engineering, pp. 1–5 (2019)
20. Hsu, S.C., Huang, C.L., Chuang, C.H.: Vehicle detection using simplified Fast R-CNN. In: International Workshop on Advanced Image Technology, pp. 1–3 (2018)
21. Hsu, S.C., Wang, Y.W., Huang, C.L.: Human object identification for human-robot Interaction by using fast R-CNN. In: IEEE International Conference on Robotic Computing (IRC), pp. 201–204 (2018)

22. Jana, S., Basak, S., Parekh, R.: Automatic fruit recognition from natural images using color and texture features. In: *The Conference on Devices for Integrated Circuit (DevIC)*, pp. 620–624 (2017)
23. Kendall, A.G.: *Geometry and uncertainty in deep learning for computer vision* (Ph.D. thesis), The University of Cambridge, UK (2019)
24. Kim, J.Y., Vogl, M., Kim, S.D.: A code-based fruit recognition method via image conversion using multiple features. In: *International Conference on IT Convergence and Security*, pp. 1–4 (2014)
25. Islam, M., Dinh, A., Wahid, K., Bhowmik, K.: Detection of potato diseases using image segmentation and multiclass support vector machine. In: *Canadian Conference on Electrical and Computer Engineering*, pp. 1–4 (2017)
26. Ji, W., Zhao, D., Cheng, F., Xu, B., Zhang, Y., Wang, J.: Automatic recognition vision system guided for apple harvesting robot. *Comput. Electr. Eng.* **38**(5), 1186–1195 (2012)
27. Juhnevica-Radenkova, K., Radenkova, V., Seglina, D.: Microbiological changes and severity of decay in apples stored for a long-term under different storage conditions. *Zemdirbyste Agric.* **103**(4), 391–396 (2016)
28. Lal, S., Behera, S.K., Sathy, P.K., Rath, A.K.: Identification and counting of mature apple fruit based on BP feed forward neural network. In: *International Conference on Sensing, Signal Processing and Security*, pp. 361–368 (2017)
29. Li, G., Ma, Z., Wang, H.: Image recognition of grape downy mildew and grape powdery mildew based on support vector machine. In: Li, D., Chen, Y. (eds.) *CCTA 2011. IAICT*, vol. 370, pp. 151–162. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-27275-2_17
30. Liu, B., Zhang, Y., He, D., Li, Y.: Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry* **10**(1), 11 (2017)
31. Liu, B., Zhao, W., Sun, Q.: Study of object detection based on faster R-CNN. In: *Chinese Automation Congress*, pp. 6233–6236 (2017)
32. Liu, C., Tao, Y., Liang, J., Li, K., Chen, Y.: Object detection based on YOLO network. In: *IEEE Information Technology and Mechatronics Engineering Conference* (2018)
33. Lu, Y., Yi, S., Zeng, N., Liu, Y., Zhang, Y.: Identification of rice diseases using deep convolutional neural networks. *Neurocomputing* **267**, 378–384 (2017)
34. Manana, M., Tu, C., Owolawi, P.A.: Preprocessed faster RCNN for vehicle detection. In: *International Conference on Intelligent and Innovative Computing Applications (ICONIC)*, pp. 1–4 (2018)
35. Marathe, A., et al.: Performance modeling under resource constraints using deep transfer learning. In: *International Conference on High Performance Computing, Networking, Storage and Analysis*, p. 31 (2017)
36. Mohamud, A.H., Gopalakrishnan, A.K.: Fruit feature recognition based on unsupervised competitive learning and backpropagation algorithms. In: *International Conference on Engineering, Applied Sciences, and Technology*, pp. 29–32 (2018)
37. Mohanty, S.P., Hughes, D.P., Salathe, M.: Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **7**, 1419 (2016)
38. Murugan, V., Vijaykumar, V.R., Nidhila, A.: A deep learning RCNN approach for vehicle recognition in traffic surveillance system. In: *International Conference on Communication and Signal Processing*, pp. 0157–0160 (2019)
39. Nourmohammadi-Khiarak, J., Mazaheri, S., Moosavi-Tayebi, R., Noorbakhsh-Devlagh, H.: Object detection utilizing modified auto encoder and convolutional neural networks. In: *Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, pp. 43–49 (2018)

40. Nyarko, E.K., Vidovic, I., Radocia, K., Cupec, R.: A nearest neighbor approach for fruit recognition in RGB-D images based on detection of convex surfaces. *Expert Syst. Appl.* **114**, 454–466 (2018)
41. Nguyen, D., Nguyen, T., Kim, H., Lee, H.-J.: A high-throughput and power-efficient FPGA implementation of YOLO CNN for object detection. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **27**(8), 1861–1873 (2019). <https://doi.org/10.1109/TVLSI.2019.2905242>
42. Rachmawati, E., Supriana, I., Khodra, M.L.: Toward a new approach in fruit recognition using hybrid RGBD features and fruit hierarchy property. In: *International Conference on Electrical Engineering, Computer Science and Informatics*, pp. 1–6 (2017)
43. Rochac, J.F., Zhang, N., Thompson, L., Oladunni, T.: A data augmentation-assisted deep learning model for high dimensional and highly imbalanced hyperspectral imaging data. In: *IEEE International Conference on Information Science and Technology* (2019)
44. Rochac, J.F., Zhang, N., Xiong, J., Zhong, J., Oladunni, T.: Data augmentation for mixed spectral signatures coupled with convolutional neural networks. In: *IEEE International Conference on Information Science and Technology* (2019)
45. Rzanny, M., Seeland, M., Waldchen, J., Mader, P.: Acquiring and preprocessing leaf images for automated plant identification: understanding the tradeoff between effort and information gain. In: *BMC*, vol. 7674 (2017)
46. Scheffler, O., Coetzee, C., Opara, U.: A discrete element model (DEM) for predicting apple damage during handling. *Biosys. Eng.* **172**, 29–48 (2018)
47. Song, W., Wang, H., Maguire, P., Nibouche, O.: Differentiation of organic and non-organic apples using near infrared reflectance spectroscopy—A pattern recognition approach. In: *IEEE Sensors*, pp. 1–3 (2016)
48. Sun, S., Wu, Q., Jiao, L., Long, Y., He, D., Song, H.: Recognition of green apples based on fuzzy set theory and manifold ranking algorithm. *Optik* **165**, 395–407 (2018)
49. Sun, W., He, Y.: Spatial-chromatic clustering for color image compression. In: *IEEE World Congress on Computation Intelligence*, pp. 1601–1604 (1998)
50. Shukla, D., Desai, A.: Recognition of fruits using hybrid features and machine learning. In: *International Conference on Computing, Analytics and Security Trends (CAST)*, pp. 572–577 (2016)
51. Szegedy, S., et al.: Going deeper with convolutions. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9 (2015)
52. Thilagavathi, M., Abirami, S.: Application of image processing in diagnosing guava leaf diseases. *Int. J. Sci. Res. Manag.* **5**(7), 5927–5933 (2017)
53. Tu, S., Xue, Y., Zheng, C., Qi, Y., Wan, H., Mao, L.: Detection of passion fruits and maturity classification using red-green-blue depth images. *Biosys. Eng.* **175**, 156–167 (2018)
54. Wang, S., Chen, Z., Ding, Z.: The unified object detection framework with arbitrary angle. In: *International Conference on Big Data and Information Analytics (BigDIA)*, pp. 103–107 (2019)
55. Wang, X., Ma, H., Chen, X.: Salient object detection via fast R-CNN and low-level cues. In: *IEEE International Conference on Image Processing (ICIP)*, pp. 1042–1046 (2016)
56. Yan, J., Wang, H., Yan, M., Diao, W., Sun, X., Li, H.: IOU-adaptive deformable R-CNN: make full use of IOU for multiclass object detection in remote sensing imagery. *Remote Sens.* **11**(3), 286 (2019)
57. Yang, C., Hu, Y., Lin, H., Sa, L., Liu, Y.: Overlapped fruit recognition for citrus harvesting robot in natural scenes. In: *International Conference on Robotics and Automation Engineering* (2017)
58. Zhang, X., Qiao, Y., Meng, F., Fan, C., Zhang, M.: Identification of maize leaf diseases using improved deep convolutional neural networks. *IEEE Access* **6**, 30370–30377 (2018)

59. Zhang, X., Wang, Z., Liu, D., Ling, Q.: DADA: deep adversarial data augmentation for extremely low data regime classification. In: International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2807–2811 (2019)
60. Zhou, X., Li, G., Gong, C., Liu, Z., Zhang, J.: Attention-guided RGBD saliency detection using appearance information. *Image Vis. Comput.* **95**, 103888 (2020)
61. Yan, W.Q.: *Computational Methods for Deep Learning: Theoretic, Practice and Applications*. Springer, Cham (2021). <https://doi.org/10.1007/978-3-030-61081-4>
62. Pan, C., Yan, W.Q.: Object detection based on saturation of visual perception. *Multimed. Tools Appl.* **79**(27–28), 19925–19944 (2020). <https://doi.org/10.1007/s11042-020-08866-x>
63. Yan, W.Q.: *Introduction to Intelligent Surveillance: Surveillance Data Capture, Transmission, and Analytics*. Springer, Cham (2019). <https://doi.org/10.1007/978-3-030-10713-0>
64. Pan, C., Li, X., Yan, W.: A learning-based positive feedback approach in salient object detection. In: IEEE IVCNZ (2018)
65. Al-Sarayreh, M., Reis, M., Yan, W., Klette, R.: Deep spectral-spatial features of snapshot hyperspectral images for red-meat classification. In: IEEE IVCNZ (2018)
66. Zhao, K., Yan, W.: Fruit detection from digital images using CenterNet. In: ISGV (2021)
67. Fu, Y.: *Fruit freshness grading using deep learning*. Master's Thesis, Auckland University of Technology, New Zealand (2020)
68. Liu, Z., Yan, W.Q., Yang, B.: Image denoising based on a CNN model. In: IEEE ICCAR, pp. 389–393 (2018)