A Mixture Model for Fruit Ripeness Identification in Deep Learning

Bingjie Xiao, Minh Nguyen, Wei Qi Yan
Auckland University of Technology, 1010 New Zealand

ABSTRACT

Visual object detection is a foundation in the field of computer vision. Since the size of visual objects in an images is various, the speed and accuracy of object detection are the focus of current research projects in computer vision. In this book chapter, our datasets consist of fruit images with various maturity. Different types of fruit are divided into the classes "ripe" and "overripe" according to the degree of skin folds. Then the object detection model is employed to automatically classify different ripeness of fruits. A family of YOLO models are representative algorithms for visual object detection. We make use of ConvNeXt and YOLOv7, which belong to the CNN network, to locate and detect fruits, respectively. YOLOv7 employs the bag-of-freebies training method to achieve its objectives, which reduces training costs and enhances detection accuracy. An extended E-ELAN module, based on the original ELAN, is proposed within YOLOv7 to increase group convolution and improve visual feature extraction. In contrast, ConvNeXt makes use of a standard neural network architecture, with ResNet-50 serving as the baseline. We compare the proposed models, which result in an optimal classification model with best precision of 98.9%.

Keywords: Visual Object Detection, YOLOv7, ConvNeXt

INTRODUCTION

In the field of computer vision (Gowdra, 2021), digital cameras are utilized to emulate biological vision, enabling computers to process the contents of images or videos in a manner akin to human perception (Pan, & Yan, 2020). The object detection (Qi, Nguyen, & Yan, 2022) task (Zhang, Wang, Liu, & Xiong 2022). in computer vision primarily focuses on identifying visual objects within entire images, which includes detecting both the object and its location (Zhao, & Yan, 2021). In the realm of visual object detection (Shi, Li, & Yamaguchi 2020)., models such as CNN (Liu, Yan,&Yang, 2018), R-CNN, Fast R-CNN, Faster R-CNN(Al-Sarayreh,et. al., 2019), and the YOLO (Zhijun, et. al., 2021) series have successfully located and classified (Liu, Nouaze, Touko Mbouembe, & Kim 2020) fruit images (Gowdra, et. al., 2021). Building upon this foundation, the YOLOv7 (Liu, & Yan, 2023) model has improved the speed and accuracy of visual object detection (Yao et. al., 2021).

In recent years, artificial intelligence has been widely employed in various fields (Wang, & Yan, 2021). In view of the lack of labor in fruit picking and subsequent fruit quality classification (Xia, Nguyen, & Yan, 2022) that requires a lot of human labors (Xia, Nguyen, & Yan, 2023). In this book chapter, we propose an automatic fruit recognition algorithm based on YOLOv7 and ConvNext (Tian, 2022) models. The application of the above is mainly to build a deep learning model that can distinguish different fruit categories (Bazame, 2021 (apples and pears) for the same kind of fruit to distinguish the category level according to the degree of skin folds (Kang, & Chen, 2020). The high-precision fruit (apple, pear) detection and recognition (Wang, &Yan, 2021) system based on deep learning can be harnessed in daily life or in the wild to detect and locate fruit targets (Fu, Nguyen, & Yan, 2022). Using deep learning algorithms, it can realize fruit target

detection and recognition in the form of pictures, videos, cameras, etc. In addition, it supports results visualization and export of image or video inspection results (Bhargava, & Bansal, 2021).

Visual object detection is characterized by using location and classification (Liu, Sun, Gu, & Deng, 2022). In a two-dimensional image, target detection can locate the position of an apple in the picture, and distinguish the current apple type as "ripe apple". Firstly, we preprocess the dataset, then input the backbone network to extract features, and take use of ELAN attention. The module acts on the corresponding channel of the feature map to obtain effective features for fruit recognition; then the model performs feature fusion to obtain semantic information and locate the feature map of the information. Finally, accurate detection results are obtained through classification and prediction frame regression calculations (Gokhale, Chavan, & Sonawane, 2023).

In this book chapter, we employ anchor boxes to label fruits and their maturity levels (Xiao, Nguyen, Yan, 2023). We leverage ConvNeXt (Qi, Nguyen, & Yan, 2022) and YOLOv7 models to obtain an optimal model for fruit ripeness classification. ConvNeXt (Hassanien, Singh, Puig, & Abdel-Nasser, 2022) optimizes the technology and parameters of the original CNN to achieve state-of-the-art performance. A characteristic of ConvNeXt is that it does not consider the visual features; it simply inputs the image as a patch and sends it to the deep learning network model for training and testing (Feng, Tan, Li, & Xie, 2022). Conversely, YOLOv7 focuses on optimizing modules and methods without increasing training costs. YOLOv7 serializes or parallelizes network layers into a convolutional group to reduce computations and enhance training speed (Junos, Mohd Khairuddin, Thannirmalai, & Dahari, 2022).

Agricultural harvesting is a labor-intensive process. Utilizing a visual object detection model to classify fruits is the motivation of this book chapter. The visual object detection pipeline is illustrated in Figure 1. The dataset is input into the model for training, and a predicted bounding box is subsequently output. As demonstrated in Figure 1, YOLO model (KIVRAK, & GÜRBÜZ, 2022) is use of the entire image as input, employs a CNN network for end-to-end design, and effectively returns the position and class label of the bounding box at the output layer (Zhang et al., 2022). In our experiments, the YOLOv7 model accurately detects and classifies fruits.

This study demonstrates the use of YOLOv7 model (Kuznetsova, Maleva, & Soloviev, 2021), ConvNeXt, and their transfer learning to detect fruits, which can accurately classify fruit types and their maturity levels (Lee & Kim, 2020). Simultaneously, we also created our own datasets using mobile phones to increase the influence of the environment on experimental results.

The contribution of this book chapter is that we created our own dataset. We take advantage of YOLOv7, ConvNext, and improved models to locate and classify fruits. The model can realize the fruit detection task and achieve high precision.

In the second section of this book chapter, we will discuss the development of the proposed YOLOv7 model. The third section will include the experimental details and outcomes. In the fourth section, we will showcase the training results of the YOLOv7 model and summarize the advantages and disadvantages of the proposed model. In this book chapter, following the section

on related work, the proposed methods will be elaborated upon. The result analysis will be explained, leading to the final conclusion of this book chapter.

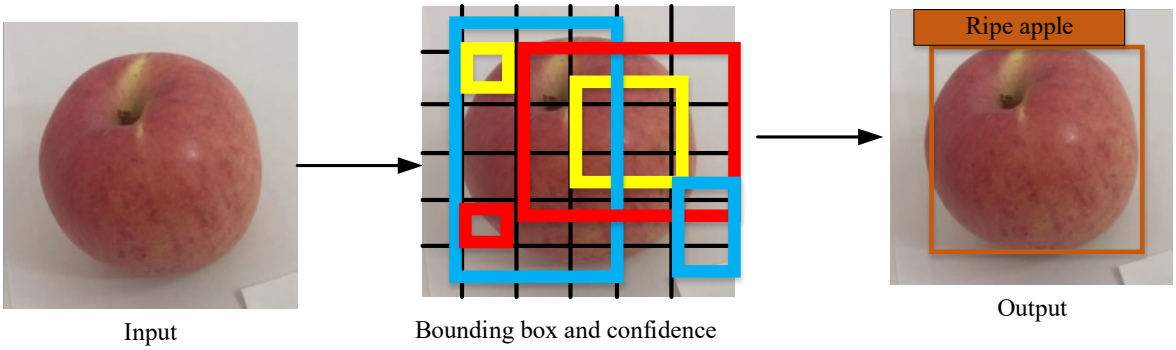


Figure 1. The pipeline of YOLO model in visual object detection.

RELATED WORK

A model with YOLOv4-based convolution block was proposed to add an attention mechanism, which judges the maturity of apples by distinguishing colors. For the actual situation in the orchard, the size of fruit trees in the orchard, the influence of branches and leaves on fruits, actual size and color of the apples (Gongal, Karkee, & Amatya, 2018) are all issues that need to be considered in the real object detection (Pan, Liu, & Yan, Zhou, 2021). The YOLOv4 model proposed by Lu et. al. achieved an accuracy 86.2% for a number of fruits, which is about 3% higher than the original YOLOv4 model (Lu et. al., 2022).

Ou et. al. proposed an improved FSOne-YOLOv7 model for the detection of passion fruit (Ou et. al., 2023). ShuffleOne as a new backbone network and slim-neck as an improved YOLOv7 network of the neck network are employed for passion fruit detection in complex natural environments. The FSOne-YOLOv7 model takes advantage of gradient weighted class activation mapping to enhance the feature extraction and fusion capabilities. Ou et. al. achieved an average accuracy 94.5%. The improved model can better extract features, thereby improving detection speed.

Zhou, et al. also studied how visual inspection can replace manual picking of dragon fruits (Zhou, Zhang, Wang, 2023). Zhou et. al. proposed a PSP-Ellipse method based on YOLOv7 to implement classification. The PSP-Ellipse method detects the endpoints of dragon fruit in the picture by segmenting the detection target and using an ellipse fitting algorithm, and then takes use of ResNet to implement the classification task. In the PSP-Ellipse endpoint detection task, the model achieved an accuracy of 92% for dragon fruit.

Another experiment that has great significance is studied. (Wu, et. al., 2022). Previous agricultural-related detections were based on fruit color and shape classification. However, traditional agricultural detection is prone to false detection in complex natural environments, and the model lacks robustness. Wu et. al. studied target detection based on complex environments, made use of the module characteristics of YOLOv7 data enhancement, and established an improved DA-YOLOv7 model. The DA-YOLOv7 model strengthens the generalization ability of the model in

complex environments, which is adopted for the detection of Camellia oleifera under the interference of side light, backlight, slight occlusion and heavy occlusion.

A visual object detection method was proposed to solve fruit counting problem. SSD was employed with MobileNet and Faster R-CNN with Inception V2 for multi-fruit object tracking based on Gaussian estimation. Vasconez et. al. achieved 90% accuracy using SSD model and 93% accuracy using Faster R-CNN (Vasconez et. al., 2020).

A nighttime dataset was collected which demonstrated that YOLOv4 model achieved F1 score 0.968 and average precision 0.983 with images from various orchards, varieties and lighting conditions for real-time mango detection (Koirala et al., 2019).

YOLOv2 was designed for visual object location prediction (Sozzi, et. al, 2022). YOLOv3 continues the idea of YOLOv2. The FPN structure is adopted in YOLOv3 to improve the accuracy of corresponding multi-scale target detection (Liu, et. al., 2022). YOLOv5 (Wang et. al., 2022) was basically modified based on the structure of YOLOv3(Wang, Jin, Wang, & Xu, 2022). YOLOv5 was use of CSPDarknet (Cross Stage Partial Networks) as the backbone to extract visual features from the input image (Wang, & He, 2021).

The difficulty of multi-target tracking in model training lies in the fact that real-life targets are occluded, blurred and deformed. In this project, fruit recognition from digital images is also affected by the natural environment, such as lighting, overlapping, scales change and other factors that affect the final results (Yang et. al., 2022). The E-ELAN module of YOLOv7 can enhance the ability of network and guide different modules. E-ELAN can enhance the net ability without changing the gradient.

Hussain et al. also proposed YOLOv7 for visual object detection (Hussain et. al., 2022). YOLOv7 NAS can implement iterative search by mining the optimal scale factor according to the resolution, width, and depth, as well as the number of feature pyramids. Reparameterization can assist the gradient propagation path to reintegrate the parameters of the model, so that the head module can be applied to fruit detection.

Swin Transformer (Ruiz, et. al., 2022) takes advantage of hierarchical feature maps which are similar to convolutional neural networks (Gowdra, 2021). After the image is downsampled by 4 times, 8 times, or 16 times in the size of the feature map, the backbone builds tasks such as target detection and instance segmentation on this basis. The concept of Windows Multi-Head Self-Attention (W-MSA) was employed in Swin Transformer. For example, in the 4 times downsampling or 8 times downsampling, the feature map is segmented into multiple disjoint regions, and each self-attention is only performed within each window. Transformer can effectively reduce the amount of calculations if the shallow feature map is large. ConvNeXt is based on ResNet, the process of transforming ResNet into ConvNet is similar to the construction process of Transformer. ConvNeXt maintains the simplicity of CNN neural networks (An, &Yan, 2021) while following the structure of Swin Transformer model.

METHODOLOGY

YOLO series models are a one-stage network structure. There is only one neural network in the whole process, which is able to achieve end-to-end structure. After an image is input into YOLO model, the image is segmented into s×s grids, each grid can be processed to obtain bounding boxes and the confidence score of each box. In Figure 2, the blue, yellow, and red boxes are the bounding boxes. Each grid predicts the conditional class probabilities, the confidence of each bounding box is multiplied by using probability. The result contains class information and accuracy of the bounding box prediction. Finally, we set the threshold, filter out the low scores, and cast the rest to non-maximum suppression, and then get the prediction box.

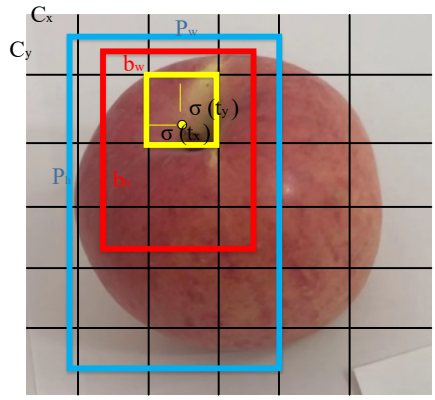


Figure 2. Prediction of bounding boxes.

In Figure 2, $b_x, b_y, b_w,$ and b_h show the value of the predicted bounding box, C_x and C_y represent the distance from the upper left corner of the current grid to the upper left corner of the image. P_w and P_h are the width and height of anchor box, respectively; σ is the sigmoid function. $t_x, t_y, t_w, t_{h,}$ and t_0 are the parameters which are employed to calculate the bounding box and confidence. The centre of the predicted box is the point inside the yellow box as shown in Figure 2, then a group of equations for calculating is listed as follows:

$$b_x = \sigma(t_x) + C_x \quad (1)$$

$$b_y = \sigma(t_y) + C_y \quad (2)$$

$$b_x = P_w e^{t^w} \quad (3)$$

$$b_x = P_h e^{t_h} \quad (4)$$

The confidence and maximum suppression values are:

$$Confidence\ score = P(Object) \times IoU_{truth\_pred} \quad (5)$$

$$Class - specific\ confidence\ scores = Confidence \times P(Class|Object) \quad (6)$$

Neck in YOLO is mainly applied to generate feature pyramids. The feature pyramid will enhance the object detection at hierarchical scales, the same object having different sizes and scales will be recognized. CSPNet backbone solves the gradient duplication problem of network optimization in the backbone, large-scale convolutional neural network frameworks integrate the gradient changes into the feature map from beginning to end, thus reduce the parameter amount and FLOPS value of the model, and ensure the inference speed and accuracy, and decrease the size of the proposed model. Head is mainly employed to the final part which applies anchor boxes to the

feature map and generate the final output vector with class probabilities, object scores and bounding boxes.

In YOLOv7, a decoupled training-time and inference-time architecture was proposed for training a multi-branch model, converting it into a single-channel model and deploying it. In this way, the high performance of the multi-branch architecture and the fast inference advantage of the single-branch model are realized.

In Figure 3, YOLOv7 merges all Conv and BN layers, and converts the fused Conv layer into a 3×3 Conv layer. A 1×1 Conv layer is converted into a 3×3 Conv layer by using the center weight that equals to the 1×1 Conv layer which merges the branch 3×3 Conv layer. Finally, the weights and bias of the convolution kernels of all branches are added to form a new 3×3 Conv layer. YOLOv7 finally constitutes an identity mapping branch, that is, a RepVGG block. The mapping network of YOLOv7 is similar to the residual network of ResNet, that is, adding a branch at a specific layer.

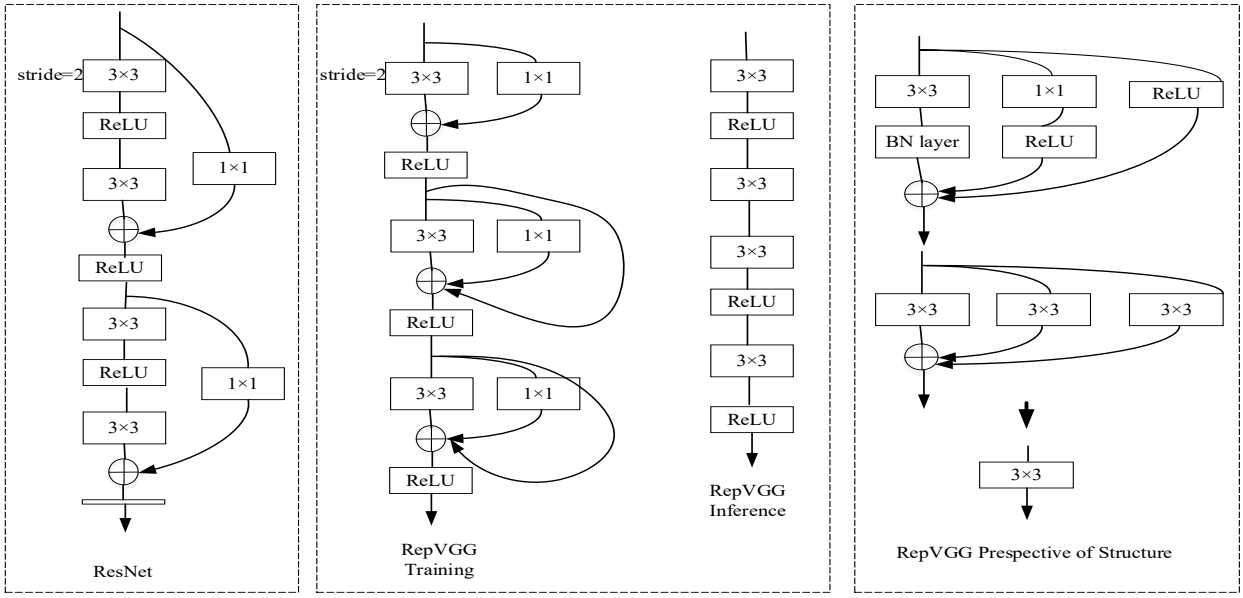


Figure 3. Reparameterization process of RepVGG

The parameter fusion process is:

$$W'_{i,:,:,:} = \frac{\gamma_i}{\sigma_i} W_{i,:,:,:} \tag{7}$$

$$b'_i = -\frac{\mu_i \gamma_i}{\sigma_i} + \beta_i \tag{8}$$

where μ_i, σ_i, γ_i, $\beta_i$are the mean, variance, scale factor and offset factor of BN, respectively, W_i is the original convolution weight. Eq.(9) for each calculation in Conv is,

$$Conv(x) = W \times X \tag{9}$$

$$BN(x) = \gamma_i \left(\frac{x - \mu_i}{\sqrt{\sigma_i^2 + \varepsilon}}\right) \tag{10}$$

where ε is equal to the minimum. The fused result of Conv and BN is,

$$\left(\frac{W\times x-\mu_i}{\sqrt{\sigma_i^2+\varepsilon}}\right)+\beta_i=\frac{\gamma_i}{\sqrt{\sigma_i^2+\varepsilon}}W_iX-\frac{\mu_i\gamma_i}{\sqrt{\sigma_i^2+\varepsilon}}+\beta_i \quad (11)$$

Ignored the minimum value ε, the new convolution is,

$$Conv(x)=W'x+\beta' \quad (12)$$

Compared with the calculations after fusion, it is essentially a linear operation of convolution.

In Figure 4, scale has always been one of the characteristics of the YOLO model. YOLOv7 adopts the composite model scaling method, modifies the depth factor and calculates the proportion of the corresponding change in the transfer layer. The optimal state of the model can be maintained by scaling the model and the width corresponding to the depth scaling factor [20, 38].

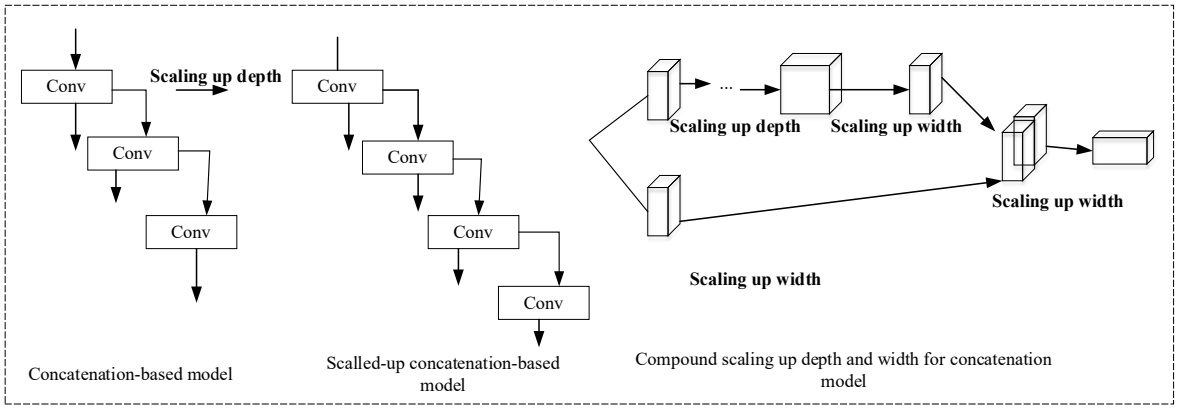


Figure 4. Stitch-based model scaling

YOLOv7 model scales the stack in the Neck module, which is use of a composite scaling method to scale the depth and width of the entire model to obtain the weight YOLOv7-X. E-ELAN is applied to the weights YOLOv7-E6. The weights of YOLOv7, YOLOv7-X and YOLOv7-E6 are use of SiLu as the activation function,

$$SiLu(x)=x\times sigmoid(x) \quad (13)$$

The SiLU function is the abbreviation of sigmoid weighted linear unit, which is adopted as the activation function. Unlike other activation functions (e.g., sigmoid, tanh), the activation function SiLU is not monotonically increasing. The SiLU function is self-stabilizing and acting as an implicit regulariser on the weights at the global minimum with zero derivative, inhibiting the learning of a large number of weights. The weight YOLOv7-tiny is an edge GPU-oriented architecture. Leaky tunes the zero-gradient problem for negative values by giving the negative input x a tiny linear component.

In Figure 5, YOLOv7 separates the auxiliary and dominant heads, and performs label assignment with the respective predictions and ground truths. Deep supervision information is employed for adding additional supervision information to the model so as to improve the performance of the

model. The leading head represents the feature map responsible for the final output, and the auxiliary head represents the additional training branch added for auxiliary training.

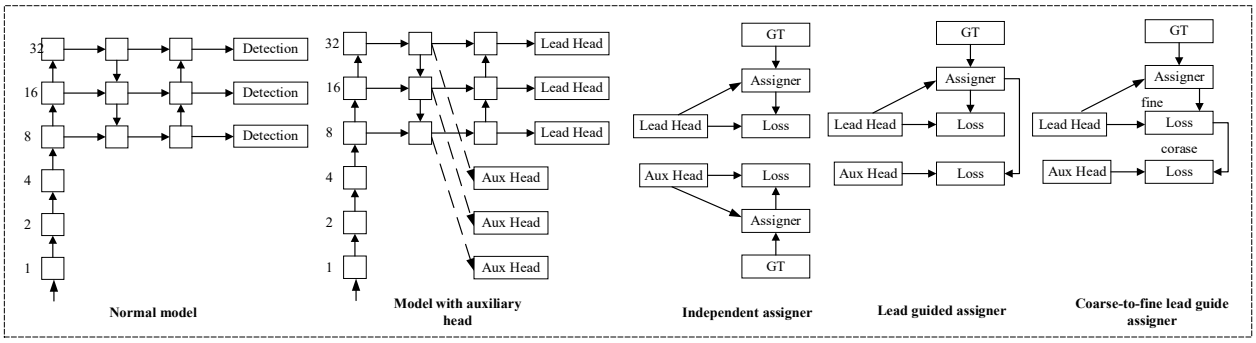


Figure 5: Auxiliary head and leading head perform label assignment process using prediction results and ground-truth values.

ELAN is an efficient long-range attention network that takes use of convolutions to extract image local structure information in complex cases, which is use of a grouped multi-scale self-attention (GMSA) module to compute on non-overlapping feature sets at different window sizes, and speed up the operation of this module through a shared attention mechanism. The E-ELAN designed by YOLOv7 based on ELAN has the ability to expand, shuffle, and merge cardinality to achieve the ability and continuously enhance the network learning ability without destroying the original gradient path.

All ConvNeXt models are the existing structures and methods, the transformation process is similar to Transformer's construction process. The starting point of ConvNeXt is ResNet, which takes use of enhanced training methods to improve the performance of the ResNet-50 model. The ConvNeXt network structure is composed of macro design, ResNeXt, inverted bottleneck, large kernel size, and various micro designs with layers as the smallest granularity.

The ConvNeXt network adjusts the stacking times of each stage of ResNet from (3, 4, 6, 3) to (3, 3, 9, 3), which increases the accuracy with the cost of increasing calculation scales. The stem layer in Swin Transformer network is a convolutional layer with a convolution kernel size of 4 and a stride of 4. The stem layer of ResNet50 consists of a convolutional layer with a kernel size of 7 and a stride of 2 plus a maximum pooling layer with a kernel size of 3 and a stride of 2. As a combination of Transformer networks and ResNet models, ConvNeXt replaces the stem layer with the same convolution layer as the Swin Transformer network with a convolution kernel size of 4 and a step size of 4, and its accuracy has a small improvement.

Compared with classical ResNet network, the ResNeXt network has achieved a balance between FLOPs and accuracy. ResNeXt takes use of group-wise convolution in the middle of the convolution block to make the convolution block form a parallel structure, while the volume of the ResNet network increases, the block is similar to the structure of bottleneck “thick at both ends and thin in the middle”. In Figure 6, compared with ResNeX and ResNet, the ConvNeXt network makes use of depth-wise convolution to form a convolution block, which greatly reduces the parameter scale of the network while sacrificing a part of the accuracy.

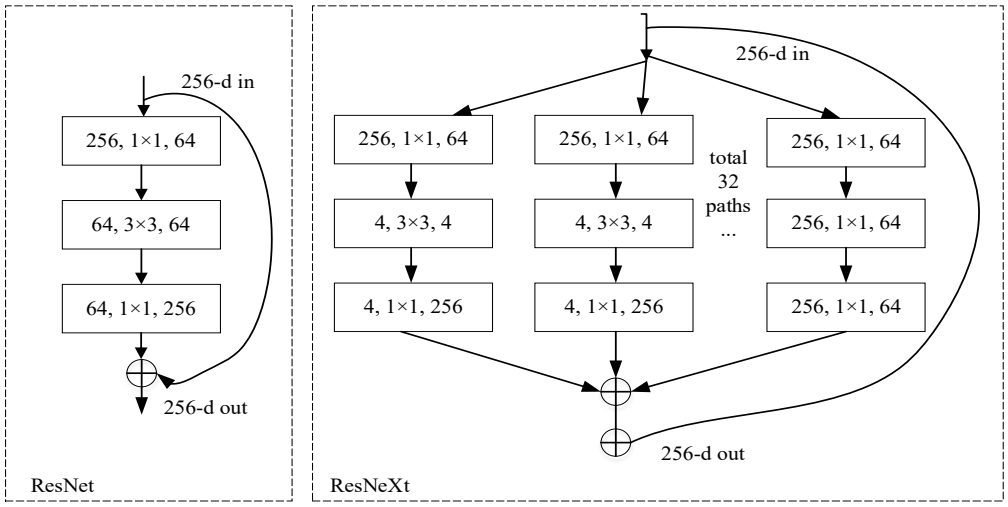


Figure 6: ResNet and ResNeXt blocks.

The number of output feature channels of the stem layer in the Swin Transformer network is 96, while the output of the stem layer of the ResNet network is only 64 dimensions. In order to be consistent with the Swin Transformer network, the ConvNeXt network increases the number of output dimensions to make it the same as the Swin-T network, which greatly improves the accuracy of the network, but at the same time inevitably increases the parameter scale of the model.

The ConvNeXt network was designed with a similar inverted bottleneck structure, which can partially reduce the parameter size of the model and improve the overall performance of the model while slightly improving the accuracy.

In the ConvNeXt network, the convolution kernel size of depthwise conv is changed from 3×3 to 7×7 like Swin Transformer, which saturates the accuracy. The current mainstream convolutional neural network has a 3×3 window size. However, a 3×3 window will result in a smaller receptive field, and ConvNeXt can increase the receptive field by using a large convolution kernel, and to a certain extent, more information can be obtained.

ConvNeX replaces the regular activation function ReLU with GELU with fewer activation functions. In a convolutional neural network, an activation function is generally connected after each convolutional layer or full connection. It is not every module in ConvNeXt which is followed by an activation function. At the same time, ConvNeXt is use of less Normalization. The normalization layer in the ConvNeXt block only retains the normalization layer after the depthwise convolution. Batch Normalization (BN) can speed up the convergence of network and reduce overfitting in the convolutional neural network. The downsampling operation of ResNet is completed at the beginning of each stage using a 3×3 convolution with a step size of 2 and a 1×1 convolution with a direct step size of 2. ConvNext performs independent downsampling between different stages, using 2 × 2 convolution with a step size of 2 for spatial downsampling. This change will lead to unstable training, so a layer-based normalization is added before the downsampling operation, after the Stem operation and the global pooling layer to stabilize the training.

RESULTS

Visual object detection is characterized by location and classification. In a two-dimensional image, target detection can locate the position of the apple in the given image, and distinguish the current apple type as “ripe apple”. Firstly, we preprocess the dataset, then input the backbone network to extract features by using ELAN attention. The module acts on the corresponding channel of the feature map to obtain effective features for fruit recognition; then the model performs feature fusion to obtain semantic information and locate the feature map of the information. Finally, accurate detection results are obtained through category classification and prediction frame regression calculations.

In this chapter, we are use of the object detection model with training dataset and the Pytorch library to build the page display. The functions supported by this chapter include the import and initialization of the fruit training model; the adjustment of confidence score and IOU threshold, image uploading, object detection, visual result display, result export and end detection, etc.

YOLOv7 attention mechanism automatically learns the importance of each feature channel, and then strengthens the features useful for fruit recognition task and suppresses the useless features according to the importance. Aiming at the problem that GIOU cannot accurately express the overlap relationship between fruit recognition frames when the prediction frame overlaps with the target frame. In this book chapter, the original frame regression loss function GIOU is replaced with CIOU, taking into account the height-to-width ratio and the center point of the target frame and prediction frame. relationship, thereby making the fruit prediction frame closer to the real frame and improving the prediction accuracy. Therefore, mean average precision (mAP) is shown as an indicator to evaluate the model.

The IOU threshold is the degree of overlap between the predicted frame and the ground-truth frame. mAP@.5 means that if IoU is set to 0.5, the AP of all pictures in each category is calculated and averaged. mAP@.5:.95 indicates different IoU thresholds, from 0.5 to 0.95, with a step length 0.05. The larger the IOU, the smaller the number of preselected boxes, which leads to a corresponding increase in the ratio. We observed Table 1~ Table 5 that mAP@.5 results are better than mAP@.5:.95. We set the IOU higher to filter out boxes with low confidence scores. Therefore, the identified frame is basically around the target and counted as a positive sample. The object detection is to select the closest positive sample based on a group of positive samples in Figure 7.

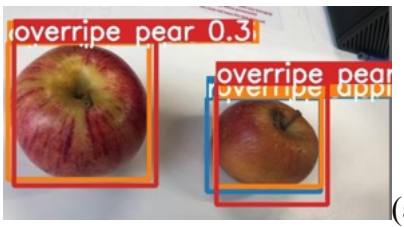
In this book chapter, fruit images taken by mobile phones are employed for fruit detection. We took use of LabelMe to annotate the dataset, the dataset has four classes: “Ripe apple”, “overripe apple”, “ripe pear”, “overripe pear”, with the bounding boxes. Given the IOU threshold 0.7, we calculate the average precision as the evaluation. We adjusted the weights of YOLOv7 model. Under the same weights, the number of iterations is taken into account on with the accuracy. We trained the model with batch size 64, Adam optimization with an initial learning rate 0.0002. Our dataset has a total of two thousand fruits and their maturity labels.

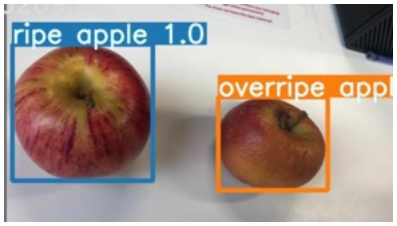
We chose four weights YOLOv7, YOLOv7-X, YOLOv7-E6 and YOLOv7-tiny for our experiments, and compared the model performance. We loaded the pretrained weights, compared the backbone network with the network parameters including the pretrained weights, and see how

many layers are the same. The training process will only load the same number of layers, we observe how the model is trained by adjusting the number of iterations. At the same time, we are use of the ConvNeXt model for training, and take advantage of ConvNeXt pre-training model for transfer learning.

The bag-of-freebies in YOLOv7 model improved the accuracy of fruit detection. The scaling method of YOLOv7 model reduces the loss of visual information. Adaptive image scaling can deepen the model with visual features, ensure that the overall image transformation is consistent, the information of receptive field can be effectively utilized. The replacement of the reparametrized module and the assignment of dynamic label assignments compute the prediction results and ground truth values, which enable the dominant leader to have strong learning ability through the optimization process.
In order to improve the recognition accuracy of fruits with only different local features and similar global features, the ELAN module of YOLOv7 aims at the problem of poor model performance in model scaling. YOLO v7 borrows from ResNeXt, takes use of 1×1 conv for dimensionality reduction, then convolutes separately, and finally adds YOLOv7 re-parameterization method in residual structure and the problem of dynamic label assignment in multiple output layers.

In Tables 1, 2 and 3, YOLOv7 can achieve better fruit positioning. Figure 7(a) shows that though the model cannot accurately determine the category of the fruit, which can still precisely locate the location of the fruit. After the model has learned enough features, Figure 9(b) shows the detection results of the model.



(a)


(b)

Figure 6: (a) and (b) are the images including fruits and the predicted boxes.

Table 1. The precisions after trained YOLOv7 model

Model	**Weights**	**Epoch**	**Class Synchronous**	**AP@.5**	**AP@.5:.95**
YOLO v7	yolov7	10	Ripe apple	0.468	0.427
			Over apple	0.885	0.764
			Ripe pear	0.169	0.115
			Overripe pear	0.209	0.151
		20	Ripe apple	0.427	0.419
			Over apple	0.995	0.932
			Ripe pear	0.673	0.622
			Overripe pear	0.967	0.944
		30	Ripe apple	0.993	0.974
			Over apple	0.996	0.948
			Ripe pear	0.542	0.534

			Overripe pear	0.995	0.932
		50	Ripe apple	0.996	0.992
			Over apple	0.996	0.979
			Ripe pear	0.996	0.981
			Overripe pear	0.996	0.991
		100	Ripe apple	0.996	0.992
			Over apple	0.996	0.988
			Ripe pear	0.996	0.995
			Over apple	0.996	0.990

Table 2. The precisions after trainied YOLOv7-tiny model.

Model	**Weights**	**Epoch**	**Class**	**AP@.5**	**AP@.5:.95**
YOLO v7	yolov7-tiny	10	Ripe apple	0.660	0.259
			Over apple	0.144	0.041
			Ripe pear	0.075	0.031
			Overripe pear	0.056	0.018
		20	Ripe apple	0.470	0.336
			Over apple	0.730	0.526
			Ripe pear	0.859	0.697
			Overripe pear	0.165	0.113
		30	Ripe apple	0.665	0.608
			Over apple	0.651	0.593
			Ripe pear	0.778	0.691
			Overripe pear	0.492	0.370
		50	Ripe apple	0.995	0.935
			Over apple	0.995	0.905
			Ripe pear	0.995	0.898
			Overripe pear	0.880	0.757
		100	Ripe apple	0.995	0.958
			Over apple	0.995	0.963
			Ripe pear	0.995	0.930
			Over apple	0.995	0.953

Table 3. The precisions after trained YOLOv7-X model.

Model	**Weights**	**Epoch**	**Class**	**AP@.5**	**AP@.5:.95**
YOLO v7	yolov7-X	10	Ripe apple	0.439	0.393
			Over apple	0.843	0.718
			Ripe pear	0.344	0.224
			Overripe pear	0.436	0.396
		20	Ripe apple	0.918	0.906
			Over apple	0.996	0.949
			Ripe pear	0.390	0.352
			Overripe pear	0.517	0.470

		30	Ripe apple	0.996	0.978
			Over apple	0.996	0.960
			Ripe pear	0.996	0.919
			Overripe pear	0.996	0.959
		50	Ripe apple	0.996	0.991
			Over apple	0.997	0.983
			Ripe pear	0.996	0.994
			Overripe pear	0.996	0.989
		100	Ripe apple	0.996	0.992
			Over apple	0.996	0.989
			Ripe pear	0.996	0.996
			Over apple	0.996	0.993

In Table 4, we observed that E-ELAN module with YOLO-E6 weights enables the deep network to converge efficiently by controlling the shortest and longest gradient paths with the same number of iterations. The weight YOLO-tiny takes use of LeakyReLU function to resolve the problem that the parameters cannot be updated after the neural network accepts the input of the outlier range. During the backpropagation process, a large gradient will be generated because the derivatives are multiplied continuously, so the parameters cannot be updated. This leads to the vanishing gradient problem. For the input of LeakyReLU less than 0, the value is negative, so there is a small gradient, which avoids the problem of aliasing in the gradient direction. As the number of epochs increases in Table 5, the number of iterations for weight updating increases, the curve goes from the initial unfitting state to the optimal fitting state.

Table 4. The precisions after trained YOLOv7-E6.

Model	**Weights**	**Epoch**	**Class**	**AP@.5**	**AP@.5:.95**
YOLO v7	yolov7-E6	10	Ripe apple	0.334	0.277
			Over apple	0.437	0.358
			Ripe pear	0.156	0.088
			Overripe pear	0.714	0.538
		20	Ripe apple	0.377	0.342
			Over apple	0.365	0.319
			Ripe pear	0.234	0.204
			Overripe pear	0.589	0.511
		30	Ripe apple	0.993	0.897
			Over apple	0.995	0.921
			Ripe pear	0.227	0.209
			Overripe pear	0.995	0.971
		50	Ripe apple	0.994	0.988
			Over apple	0.996	0.969
			Ripe pear	0.995	0.930
			Overripe pear	0.995	0.923
		100	Ripe apple	0.995	0.991
			Over apple	0.996	0.986
			Ripe pear	0.995	0.995

			Over apple	0.995	0.992

Table 5. The mean average precisions (mAP).

Model	**Weights**	**Epoch**	**AP@.5**	**AP@.5:.95**	**Average inference time(millisecond)**
YOLOv7	YOLOv7	10	0.433	0.364	6
		20	0.766	0.730	6
		30	0.882	0.847	6
		50	0.996	0.986	6
		100	0.996	0.991	6
	YOLOv7-tiny	10	0.234	0.087	8
		20	0.556	0.418	8
		30	0.644	0.565	7
		50	0.966	0.874	7
		100	0.995	0.951	6
	YOLOv7-X	10	0.515	0.433	9
		20	0.705	0.669	9
		30	0.996	0.954	9
		50	0.996	0.989	9
		100	0.996	0.993	10
	YOLOv7-E6	10	0.410	0.315	13
		20	0.391	0.344	13
		30	0.803	0.749	13
		50	0.995	0.952	13
		100	0.995	0.991	12

Table 6. The results of ConvNeXt model for fruit detection

Model	**Weights**	**Epoch**	**AP@.5**	**AP@.5:.95**	**Average inference time(millisecond)**
ConvNeXt	ConvNext + Mask R-CNN	10	0.848	0.719	5
		20	0.948	0.678	13
		30	0.926	0.669	37
		50	0.844	0.617	58
	ConvNext + Mask R-CNN Transfer Learning	10	0.500	0.701	14
		20	0.487	0.695	4
		30	0.483	0.694	5
		50	0.483	0.695	5

As a traditional CNN model, ConvNeXt shows better training results. Under the same training parameters, the transfer learning model does not have much advantage. But in Figure 9, transfer learning saves much time if pre-training parameters are frozen. The ConvNeXt model cannot fully capture fruit features. In Tables 5 and Table 6, the ConvNeXt transfer learning model has a slight advantage in detection speed. But in terms of accuracy, the YOLO model is still better.

CONCLUSION

In conclusion, our comparative study of ConvNeXt and YOLOv7, which exemplifies CNN and YOLO architectures respectively, has demonstrated remarkable performance in the domain of fruit detection. The ConvNeXt model builds upon the residual structure of ResNet, thereby significantly enhances the speed of detection. Taking into account the primary objective of our research, which is the realization of automated fruit harvesting, we conclude that the lightweight YOLOv7 model presents a more favorable balance between detection accuracy and computational efficiency.

Fruit detection from digital images still encounters many complex problems, and the impact of the environment on image quality can easily cause errors in detection. In our experiments, in order to increase the possibility of the detection target, we screened a part of the data that was greatly affected by the environment when making the dataset. Our follow-up experiments will make up for this shortcoming, and study how to use the deep learning model to realize the target detection of fruits in environments such as light and rain.

REFERENCES

Al-Sarayreh, M., Reis, M., Yan, W., Klette, R. (2019) A sequential CNN approach for foreign object detection in hyperspectral images. International Conference on Information, Communications and Signal.

Al-Sarayreha, M. (2020) Hyperspectral Imaging and Deep Learning for Food Safety. PhD Thesis, Auckland University of Technology, New Zealand

An, N., Yan, W. (2021) Multitarget tracking using Siamese neural networks. ACM Transactions on Multimedia Computing, Communications and Applications.

An, N. (2020) Anomalies Detection and Tracking Using Siamese Neural Networks. Master's Thesis, Auckland University of Technology, New Zealand.

Bazame, H. C., Molin, J. P., Althoff, D., & Martello, M. (2021). Detection, classification, and mapping of coffee fruits during harvest with computer vision. Computers and Electronics in Agriculture, 183, 106066.

Bhargava, A., & Bansal, A. (2021). Fruits and vegetables quality evaluation using computer vision: A review. Journal of King Saud University-Computer and Information Sciences, 33(3), 243-257.

Feng, J., Tan, H., Li, W., & Xie, M. (2022). Conv2NeXt: Reconsidering Conv NeXt Network Design for Image Recognition. International Conference on Computers and Artificial Intelligence Technologies (CAIT) (pp. 53-60). IEEE.

Fu, Y., Nguyen, M., Yan, W. (2022) Grading methods for fruit freshness based on deep learning. Springer Nature Computer Science.

Fu, Y. (2020) Fruit Freshness Grading Using Deep Learning. Master's Thesis, Auckland University of Technology, New Zealand.

Gokhale, A., Chavan, A., & Sonawane, S. (2023). Leveraging ML techniques for image-based freshness index prediction of fruits and vegetables. International Conference on Emerging Smart Computing and Informatics (ESCI) (pp. 1-6). IEEE.

Gongal, A., Karkee, M., & Amatya, S. (2018). Apple fruit size estimation using a 3D machine vision system. Information Processing in Agriculture, 5(4), 498-503.

Gowdra, N., Sinha, R., MacDonell, S., Yan, W. (2021) Maximum Categorical Cross Entropy (MCCE): A noise-robust alternative loss function to mitigate racial bias in Convolutional Neural Networks (CNNs) by reducing overfitting. Pattern Recognition.

Gowdra, N. (2021) Entropy-Based Optimization Strategies for Convolutional Neural Networks. PhD Thesis, Auckland University of Technology, New Zealand.

Hussain, M., Al-Aqrabi, H., Munawar, M., Hill, R., & Alsboui, T. (2022). Domain feature mapping with YOLOv7 for automated edge-based pallet racking inspections. Sensors, 22(18), 6927.

Hassanien, M. A., Singh, V. K., Puig, D., & Abdel-Nasser, M. (2022). Predicting breast tumor malignancy using deep ConvNeXt radiomics and quality-based score pooling in ultrasound sequences. Diagnostics, 12(5), 1053.

Junos, M. H., Mohd Khairuddin, A. S., Thannirmalai, S., & Dahari, M. (2022). Automatic detection of oil palm fruits from UAV images using an improved YOLO model. The Visual Computer, 38(7), 2341-2355.

Kang, H., & Chen, C. (2020). Fast implementation of real-time fruit detection in apple orchards using deep learning. Computers and Electronics in Agriculture, 168, 105108.

KIVRAK, O., & GÜRBÜZ, M. Z. (2022). Performance comparison of YOLOv3, YOLOv4 and YOLOv5 algorithms: A case study for poultry recognition. Avrupa Bilim ve Teknoloji Dergisi, (38), 392-397.

Koirala, A., Walsh, K. B., Wang, Z., & McCarthy, C. (2019). Deep learning for real-time fruit detection and orchard fruit load estimation: Benchmarking of 'MangoYOLO'. Precision Agriculture, 20(6), 1107-1135.

Kuznetsova, A., Maleva, T., & Soloviev, V. (2021). YOLOv5 versus YOLOv3 for apple detection. Cyber-Physical Systems: Modelling and Intelligent Control (pp. 349-358). Springer, Cham.

Lee, Y. H., & Kim, Y. (2020). Comparison of CNN and YOLO for object detection. Journal of the Semiconductor & Display Technology, 19(1), 85-92.

Lu, S., Chen, W., Zhang, X., & Karkee, M. (2022). Canopy-attention-YOLOv4-based immature/mature apple fruit detection on dense-foliage tree architectures for early crop load estimation. Computers and Electronics in Agriculture, 193, 106696.

Liu, G., Nouaze, J. C., Touko Mbouembe, P. L., & Kim, J. H. (2020). YOLO-Tomato: A robust algorithm for tomato detection based on YOLOv3. Sensors, 20(7), 2145.

Liu, H., Sun, F., Gu, J., & Deng, L. (2022). SF-YOLOv5: A lightweight small object detection algorithm based on improved feature fusion mode. Sensors, 22(15), 5817.

Liu, X., Li, G., Chen, W., Liu, B., Chen, M., & Lu, S. (2022). Detection of dense citrus fruits by combining coordinated attention and cross-scale connection with weighted feature fusion. Applied Sciences, 12(13), 6600.

Liu, X., & Yan, W. Q. (2023). Vehicle-related distance estimation using customized YOLOv7. IVCNZ 2022, Auckland, New Zealand (pp. 91-103). Cham: Springer Nature Switzerland.

Liu, Z., Yan, W., Yang, B. (2018) Image denoising based on a CNN model. International Conference on Control, Automation and Robotics.

Pan, C., Yan, W. (2018) A learning-based positive feedback in salient object detection. International Conference on Image and Vision Computing New Zealand.

Pan, C., Yan, W. (2020) Object detection based on saturation of visual perception. Multimedia Tools and Applications, 79 (27-28), 19925-19944.

Pan, C., Liu, J., Yan, W., Zhou, Y. (2021) Salient object detection based on visual perceptual saturation and two-stream hybrid networks. IEEE Transactions on Image Processing.

Ou, J., Zhang, R., Li, X., & Lin, G. (2023). Research and explainable analysis of a real-time passion fruit detection model based on FSOne-YOLOv7. Agronomy, 13(8), 1993.

Qi, J., Nguyen, M., & Yan, W. (2022). Waste classification from digital images using ConvNeXt. Pacific-Rim Symposium on Image and Video Technology.

Qi, J., Nguyen, M., Yan, W. (2022) Small visual object detection in smart waste classification using Transformers with deep learning. International Conference on Image and Vision Computing New Zealand (IVCNZ).

Ruiz, N., Bargal, S., Xie, C., Saenko, K., & Sclaroff, S. (2022). Finding differences between Transformers and ConvNets using counterfactual simulation testing. Advances in Neural Information Processing Systems, 35, 14403-14418.

Tian, G., Wang, Z., Wang, C., Chen, J., Liu, G., Xu, H., ... & Peng, L. (2022). A deep ensemble learning-based automated detection of COVID-19 using lung CT images and Vision Transformer and ConvNeXt. Frontiers in Microbiology, 13.

Shi, R., Li, T., & Yamaguchi, Y. (2020). An attribution-based pruning method for real-time mango detection with YOLO network. Computers and Electronics in Agriculture, 169, 105214.

Sozzi, M., Cantalamessa, S., Cogato, A., Kayad, A., & Marinello, F. (2022). Automatic bunch detection in white grape varieties using YOLOv3, YOLOv4, and YOLOv5 deep learning algorithms. Agronomy, 12(2), 319.

Vasconez, J. P., Delpiano, J., Vougioukas, S., & Cheein, F. A. (2020). Comparison of convolutional neural networks in fruit detection and counting: A comprehensive evaluation. Computers and Electronics in Agriculture, 173, 105348.

Wang, D., & He, D. (2021). Channel pruned YOLO V5s-based deep learning approach for rapid and accurate apple fruitlet detection before fruit thinning. Biosystems Engineering, 210, 271-281.

Wang, Z., Jin, L., Wang, S., & Xu, H. (2022). Apple stem/calyx real-time recognition using YOLO-v5 algorithm for fruit automatic loading system. Postharvest Biology and Technology, 185, 111808.

Wang, L., Zhao, Y., Xiong, Z., Wang, S., Li, Y., & Lan, Y. (2022). Fast and precise detection of litchi fruits for yield estimation based on the improved YOLOv5 model. Frontiers in Plant Science, 13.

Wang, L., Yan, W. (2021) Tree leaves detection based on deep learning. International Symposium on Geometry and Vision.

Wu, D., Jiang, S., Zhao, E., Liu, Y., Zhu, H., Wang, W., & Wang, R. (2022). Detection of Camellia oleifera fruit in complex scenes by using YOLOv7 and data augmentation. Applied Sciences, 12(22), 11318.

Xia, Y., Nguyen, M., Yan, W. (2022) A real-time Kiwifruit detection based on improved YOLOv7. International Conference on Image and Vision Computing New Zealand (IVCNZ)

Xia, Y., Nguyen, M., Yan, W. (2023) Kiwifruit counting using KiwiDetector and KiwiTracker. IntelliSys.

Xiao, B., Nguyen, M., Yan, W. (2023) Apple ripeness identification from digital images using transformers. Multimedia Tools and Applications, Springer.
Xiao, B., Nguyen, M., Yan, W. (2023) Fruit ripeness identification using transformers. Applied Intelligence, Springer Science and Business Media LLC.
Xiao, B., Nguyen, M., Yan, W. (2021) Apple ripeness identification using deep learning. International Symposium on Geometry and Vision.
Yan, B., Fan, P., Lei, X., Liu, Z., & Yang, F. (2021). A real-time apple targets detection method for picking robot based on improved YOLOv5. Remote Sensing, 13(9), 1619.
Yang, R., Hu, Y., Yao, Y., Gao, M., & Liu, R. (2022). Fruit target detection based on BCo-YOLOv5 model. Mobile Information Systems, 2022.
Yao, J., Qi, J., Zhang, J., Shao, H., Yang, J., & Li, X. (2021). A real-time detection algorithm for Kiwifruit defects based on YOLOv5. Electronics, 10(14), 1711.
Zhang, H., Wang, Y., Liu, Y., & Xiong, N. (2022). IFD: An intelligent fast detection for real-time image information in industrial IoT. Applied Sciences, 12(15), 7847.
Zhang, Y., Zhang, Y., & Zhang, Y. (2022). Fruit and vegetable disease identification based on updating the activation function for the ConvNeXt model. International Conference on Electronic Information Technology and Computer Engineering (pp. 1045-1049).
Zhao, K. (2021) Fruit Detection Using CenterNet. Master's Thesis, Auckland University of Technology, New Zealand.
Zhao, K., Yan, W. (2021) Fruit detection from digital images using CenterNet. International Symposium on Geometry and Vision.
Zhijun, L. I., Shenghui, Y. A. N. G., Deshuai, S. H. I., Xingxing, L. I. U., & Yongjun, Z. H. E. N. G. (2021). Yield estimation method of apple tree based on improved lightweight YOLOv5. Smart Agriculture, 3(2), 100.
Zhou, J., Zhang, Y., & Wang, J. (2023). A dragon fruit picking detection method based on YOLOv7 and PSP-Ellipse. Sensors, 23(8), 3803.