agronomy

MDPI

Review

Fruit Detection and Recognition Based on Deep Learning for Automatic Harvesting: An Overview and Review

Feng Xiao, Haibin Wang *, Yueqin Xu and Ruiqing Zhang

College of Mechanical and Electrical Engineering, Northeast Forestry University, Harbin 150040, China; xiaofeng@nefu.edu.cn (F.X.)
* Correspondence: whb_nefu@nefu.edu.cn

Abstract: Continuing progress in machine learning (ML) has led to significant advancements in agricultural tasks. Due to its strong ability to extract high-dimensional features from fruit images, deep learning (DL) is widely used in fruit detection and automatic harvesting. Convolutional neural networks (CNN) in particular have demonstrated the ability to attain accuracy and speed levels comparable to those of humans in some fruit detection and automatic harvesting fields. This paper presents a comprehensive overview and review of fruit detection and recognition based on DL for automatic harvesting from 2018 up to now. We focus on the current challenges affecting fruit detection performance for automatic harvesting: the scarcity of high-quality fruit datasets, fruit detection of small targets, fruit detection in occluded and dense scenarios, fruit detection of multiple scales and multiple species, and lightweight fruit detection models. In response to these challenges, we propose feasible solutions and prospective future development trends. Future research should prioritize addressing these current challenges and improving the accuracy, speed, robustness, and generalization of fruit vision detection systems, while reducing the overall complexity and cost. This paper hopes to provide a reference for follow-up research in the field of fruit detection and recognition based on DL for automatic harvesting.

Keywords: computer vision; deep learning; fruit detection; fruit recognition; automatic harvesting; current challenge; development trend; research review

check for updates

Citation: Xiao, F.; Wang, H.; Xu, Y.; Zhang, R. Fruit Detection and Recognition Based on Deep Learning for Automatic Harvesting: An Overview and Review. *Agronomy* **2023**, *13*, 1625. https://doi.org/10.3390/agronomy13061625

Academic Editor: Shubo Wang

Received: 11 May 2023
Revised: 13 June 2023
Accepted: 14 June 2023
Published: 16 June 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In recent years, the application of artificial intelligence (AI) techniques and robotic systems to automate agricultural processes has garnered significant interest (as shown in Figure 1). Fruits usually grow in complex environments with many uncertainties. Powerful fruit vision detection systems are necessary for intelligent agriculture and automatic harvesting. Fruit vision detection systems' characteristics mainly include imaging sensors and visual information about fruits. Fruit vision detection systems generally operate through the five stages (as shown in Figure 2): fruit image acquisition, fruit image preprocessing, fruit feature extraction, fruit image segmentation, and fruit image recognition. Black and white cameras, red–green–blue (RGB) cameras, spectral cameras, thermal cameras, and RGB-depth map (RGB-D) cameras (as shown in Figure 3) are commonly used for fruit vision detection systems to obtain color, shape, texture, and size information of fruits in specific operational areas. A comparison of different types of imaging sensors is shown in Table 1. Fruit images acquired through different imaging methods are shown in Figure 4. The main research processes of fruit detection and recognition methods are shown in Figure 5. Since DL has a strong ability to extract high-dimensional features from fruit images, researchers have proposed many fruit detection and recognition methods based on DL (you only look once (YOLO), single shot multibox detector (SSD), Alex Krizhevsky networks (AlexNet), visual geometry group networks (VGGNet), residual networks (ResNet), faster region-convolutional neural networks (Faster R-CNN), fully convolutional networks (FCN), SegNet, and mask region-convolutional

neural networks (Mask R-CNN)) for automatic harvesting (as shown in Table 2). Despite much research, many challenges need to be overcome to build an effective fruit vision detection and harvesting system.

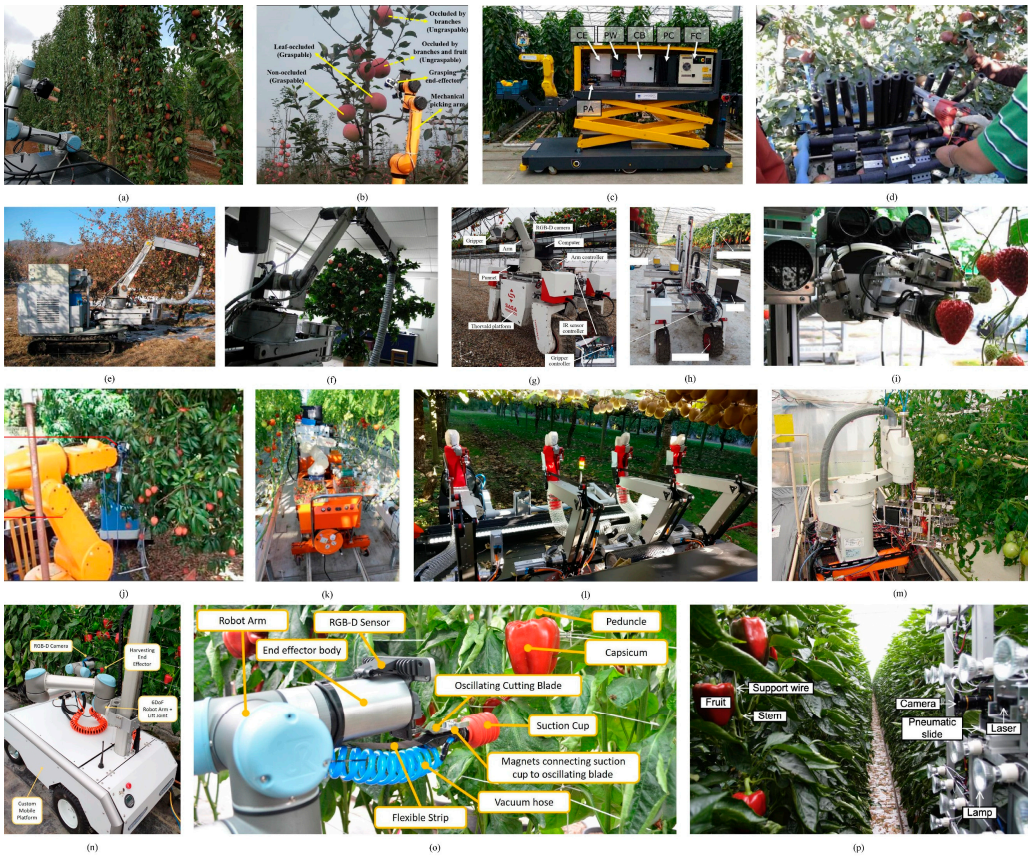


Figure 1. Typical harvesting robots. (**a**) A plum-harvesting robot (photo reprinted with permission from ref. [1]. 2021, Brown, J.); (**b**,**d**–**f**) Apple-harvesting robots (photo reprinted with permission from ref. [2]. 2021, Yan, B.; ref. [3]. 2017, He, L.; ref. [4]. 2012, Ji, W.; ref. [5]. 2011, Zhao, D.); (**c**,**n**–**p**) Sweet pepper-harvesting robots (photo reprinted with permission from ref. [6]. 2020, Arad, B.; ref. [7]. 2017, Lehnert, C.; ref. [8]. 2014, Bac, C.W.); (**g**–**i**) Strawberry-harvesting robots (photo reprinted with permission from ref. [9]. 2020, Xiong, Y.; ref. [10]. 2019, Xiong, Y.; ref. [11]. 2010, Hayashi, S.); (**j**) A lychee-harvesting robot (photo reprinted with permission from ref. [12]. 2018, Xiong, J.); (**k**,**m**) Tomato-harvesting robots (photo reprinted with permission from ref. [13]. 2018, Feng, Q.; ref. [14]. 2010, Kondo, N.); (**l**) A kiwifruit-harvesting robot (photo reprinted with permission from ref. [15]. 2019, Williams, H.A.M.).

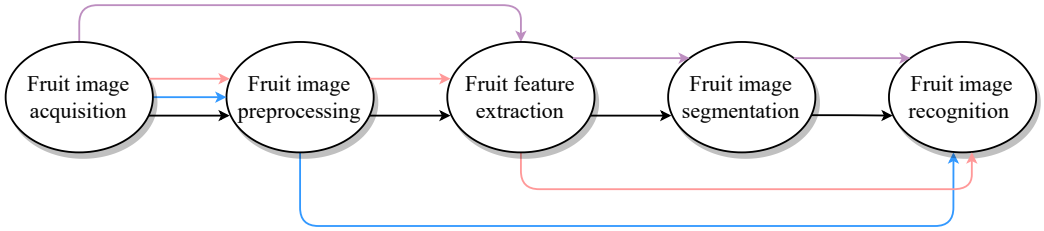


Figure 2. Different processes of fruit detection and recognition based on DL (image reprinted with permission from ref. [16]. 2023, Xiao F.).

Figure 3. Different types of imaging sensors commonly used for fruit vision detection systems (accessed on 5 January 2023).

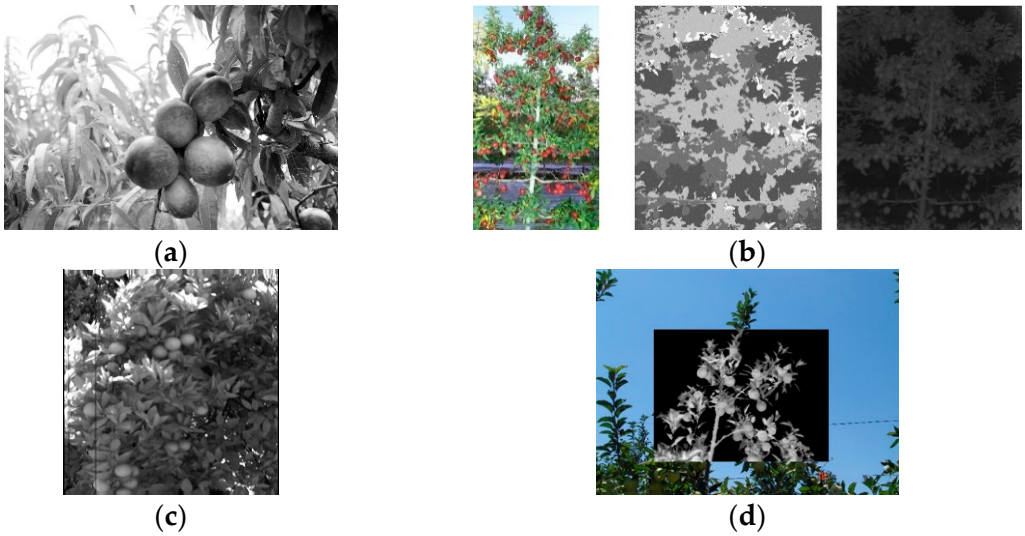


Figure 4. Fruit images acquired using different imaging methods. (**a**) Black and white image; (**b**) RGB, depth, and infrared images (photos reprinted with permission from ref. [17]. 2020, Fu L.); (**c**) spectral image (photo reprinted with permission from ref. [18]. 2009, Okamoto H.); (**d**) color and thermal-registered image (photo reprinted with permission from ref. [19]. 2010, Wachs J.P.).

Some review articles have been published encompassing diverse agricultural applications, such as crop recognition, fruit counting, weed discrimination, and plant disease detection, with or without a robotic system, by considering AI/computer vision (CV)/other advanced vision control techniques. For example, Rehman, T.U. et al., (including researchers based in America and Canada) (2019) [20] provided a comprehensive summary of ML algorithms that have been utilized in diverse agricultural operations. Brazilian researchers Patrício, D.I. and Rieder, R. (2018) [21] investigated potential applications of machine vision (MV) for diverse agricultural tasks, such as crop disease/pest detection, grain quality evaluation, and automatic plant phenotyping. Narvaez, F.Y. et al., (including researchers based in Chile, Italy, and America) (2017) [22] summarized various sensing

techniques, along with their limitations, to categorize fruits/plants. Indian researchers Jha, K. et al., (2019) [23] outlined the latest smart methodologies, such as the Internet of Things (IoT), for agricultural purposes. Dutch researchers Wolfert, S. et al., (2017) [24] reviewed the application of big data in agriculture. There are also some review articles that have been published incorporating only a particular type of agricultural application or scenario. For example, we reviewed fruit detection and recognition techniques based on digital image processing and traditional ML for fruit harvesters in [16]. New Zealand researchers Saleem, M.H. et al., (2019) [25] summarized and explained DL models for the identification and classification of plant diseases, along with the application of DL with advanced imaging techniques, including hyperspectral/multispectral imaging. Wang, D. et al., (including American researchers and a researcher based in Israel) (2019) [26] and Chinese researchers Wang, A. et al., (2019) [27] reviewed procedures for weed detection using various classification methods, including ML and DL. The review literature on AI/ML/DL/MV/CV/other advanced vision control techniques for intelligent agriculture and automatic harvesting also includes [28–41]. However, unlike the articles mentioned above, our work focuses on providing an overview and review of the use of DL applied to fruit image recognition (mainly in the areas of detection and classification) for automatic harvesting. In order to further define the study areas of our paper, we identify fruit detection and classification tasks such as the determination of classes based on their specific types.

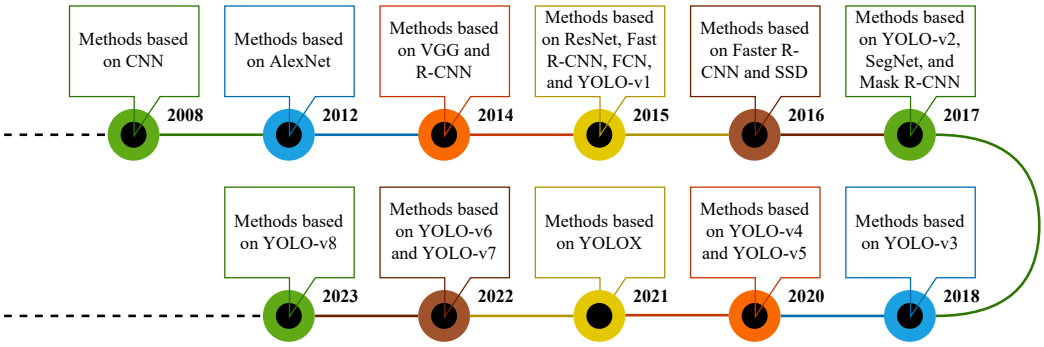


Figure 5. Main research processes of fruit detection and recognition methods based on DL.

Table 1. Comparison of different types of imaging sensors commonly used in fruit vision detection systems.

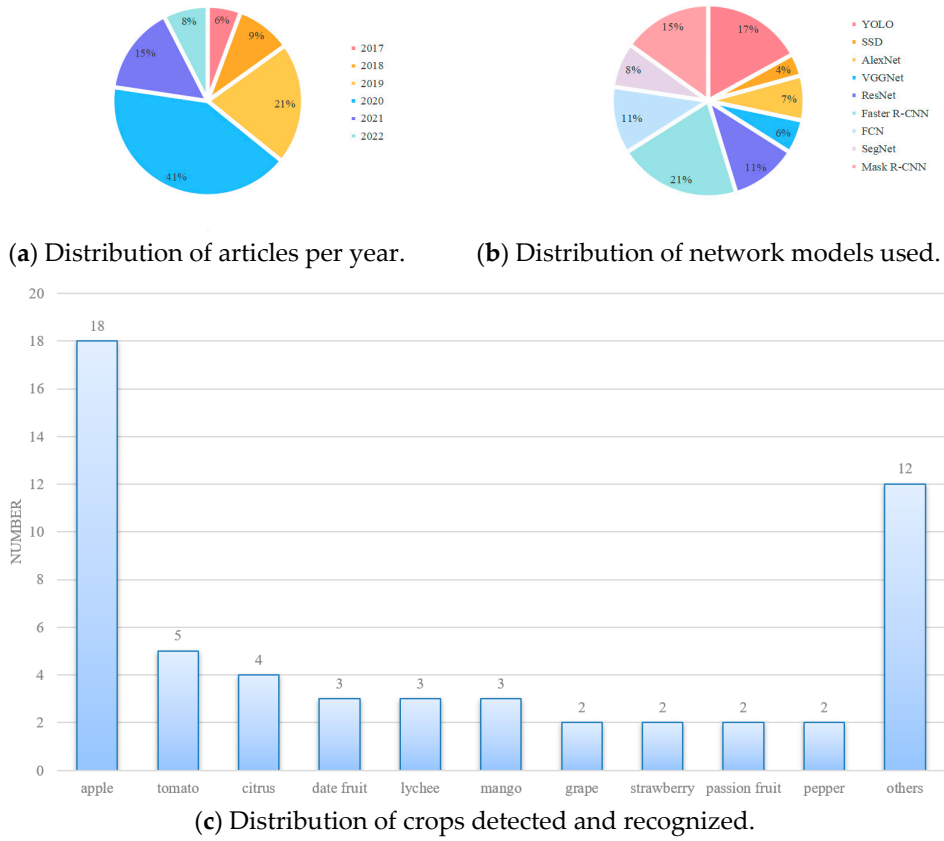
Fruit Imaging Sensors	Types	Information	Advantages	Limitations
RGB-D camera and LSS (Lift, Splat, Shoot)	Active	RGB and depth images	Complete fruit scene characteristics	Lack of feature descriptors
Black and white camera	Passive	Shape and texture features	Little effect of changes in lighting conditions	Lack of color information
RGB camera		Color, shape, and texture features	Exploiting all the basic features of target fruits	Highly sensitive to changing lighting conditions
Spectral camera		Color features and spectral information	Providing more information about reflectance	Computationally expensive for complete spectrum analysis
Thermal camera		Thermal signatures	Color-invariant	Dependency on minute thermal difference

Table 2. Fruit detection and recognition methods based on DL.

Types	Accuracy	Applied Crops	Advantages	Disadvantages
YOLO	84–98%	cabbage, citrus, lychee, mango, tomato	High fruit detection speed; it can meet real-time requirements well for automatic harvesting	Fruit detection accuracy under severe occlusion, low resolution, and changing lighting conditions is low
SSD	75–92%	apple, mango, pear, sour lemon	High detection accuracy and speed; good robustness and generalization	Fruit images need to be preprocessed; detection accuracy for small targets is low
AlexNet	86–96%	apple, strawberry, sugar beet, tomato	Using dropout to avoid overfitting; good generalization ability	Network convergence takes a little longer
VGGNet	92–99%	jujube, potato, sugar beet, tomato	Simple structure of fruit vision detection models	Network convergence takes a little longer; using more network parameters
ResNet	90–95%	apple, banana	Using residual blocks to deepen network layers and reduce network parameters	Too deep network layers may result in vanishing gradients, poor training effectiveness, and low detection accuracy
Faster R-CNN	90–99%	apple, mango, orange	High detection accuracy	Fruit detection speed is slow, and it cannot meet real-time requirements well
FCN	89–98%	cotton, grape, guava, kiwifruit	Accepting fruit image inputs with arbitrary sizes; high efficiency and low computational effort	Insensitive to the details of fruits in fruit images; fruit classification does not consider inter-pixel relationships
SegNet	83–95%	apple, tomato	Obtaining edge contours and maintaining the integrity of high-frequency details in segmentation	Neighboring information may be ignored when fruit feature maps with low resolution are unpooled
Mask R-CNN	80–94%	apple, strawberry, tomato	Combining semantic segmentation with fruit detection by outputting mask images	Fruit detection speed is slow, and it cannot meet real-time requirements well

The contributions of this work are as follows: (1) systematically summarizes and explains all kinds of fruit detection and recognition methods based on DL for automatic harvesting from 2018 up to now; (2) systematically compares and analyzes the advantages, disadvantages, and applicability of various fruit detection and recognition methods based on DL for automatic harvesting; (3) systematically demonstrates the current challenges affecting fruit detection performance for automatic harvesting and proposes feasible solutions and prospective future potential developments. Through this clearer and more comprehensive overview and review, we aim to provide a reference for follow-up research in the field of fruit detection and recognition based on DL for automatic harvesting.

According to Martín-Martín, A. et al., (including Spanish researchers and a researcher based in the UK) (2018) [42], Google Scholar citation data encompass a larger set of publications than Web of Science and Scopus. In order to comprehensively survey the literature relevant to the scope of this article, the Google Scholar database has been selected as the source. In the first step, combinations of keywords such as "fruit detection", "fruit recognition", "deep learning", "computer vision", and "fruit harvesting" were utilized in the initial search process. All retrieved papers were subsequently evaluated for their relevance to the subject matter. The second step included the examination of the references from step one for a more thorough review. In the final step, to ensure that our study focuses on the most current research, all papers published before 2018 were excluded. Only the recent literature from 2018 to the present was considered. The final set of papers regarding fruit detection and recognition based on DL for automatic harvesting included 53 research articles. Figure 6 displays the distribution of articles per year, network models used, and crops detected.



(**a**) Distribution of articles per year. (**b**) Distribution of network models used.

(**c**) Distribution of crops detected and recognized.

Figure 6. Summary of literature search.

As shown in Figure 6, in recent years, the application of DL techniques and robotic systems to automate agricultural processes has garnered significant interest. Improvement and application research based on Faster R-CNN (21%) is currently a hotspot. The recognition accuracy of fruit detection methods based on Faster R-CNN is high, but recognition speed is limited by complex anchor frame mechanisms. When there are mobile deployment and high recognition speed requirements, fruit detection methods based on YOLO (17%) are used most frequently. Their recognition speed is fast, but the recognition effect for small target fruits is not very good. In addition, ResNet (11%) is the most popular backbone network, followed by AlexNet (7%).

Most of the research focuses on apples (32.14%), followed by tomatoes (8.93%), and citrus (7.14%). These three kinds of fruits are in high demand and yield globally. There are some reasons that make them ideal candidates for automatic harvesting. Firstly, they individually hang from plants, making them easily detectable based on their distinctive features. Secondly, they have no extreme variations in size or weight. Lastly, they are relatively hard and not easily damaged in mechanical operations. However, in terms of fruit dimensions and peduncle length, different cultivars may exhibit different characteristics, which can affect fruit detection and recognition performance. This poses challenges for adapting fruit detection and recognition methods for different cultivars. Future work could aim to identify cultivars that are more suitable for automatic harvesting.

The outline of this article is shown in Figure 7. The organization of the rest of the paper is as follows: Section 2 summarizes and explains previous research articles about DL applied to fruit detection and recognition for automatic harvesting. We compare and analyze the advantages, disadvantages, and applicability of various fruit detection and recognition methods based on DL (YOLO, SSD, AlexNet, VGGNet, ResNet, Faster R-CNN, FCN, SegNet, and Mask R-CNN) for automatic harvesting; Section 3 discusses the current challenges affecting fruit detection and recognition performance for automatic harvesting (scarcity of high-quality fruit datasets, fruit detection of small targets, fruit detection in

occluded and dense scenarios, fruit detection of multiple scales and multiple species, and lightweight fruit detection models) and proposes feasible solutions and prospective future development trends; Section 4 concludes this article.

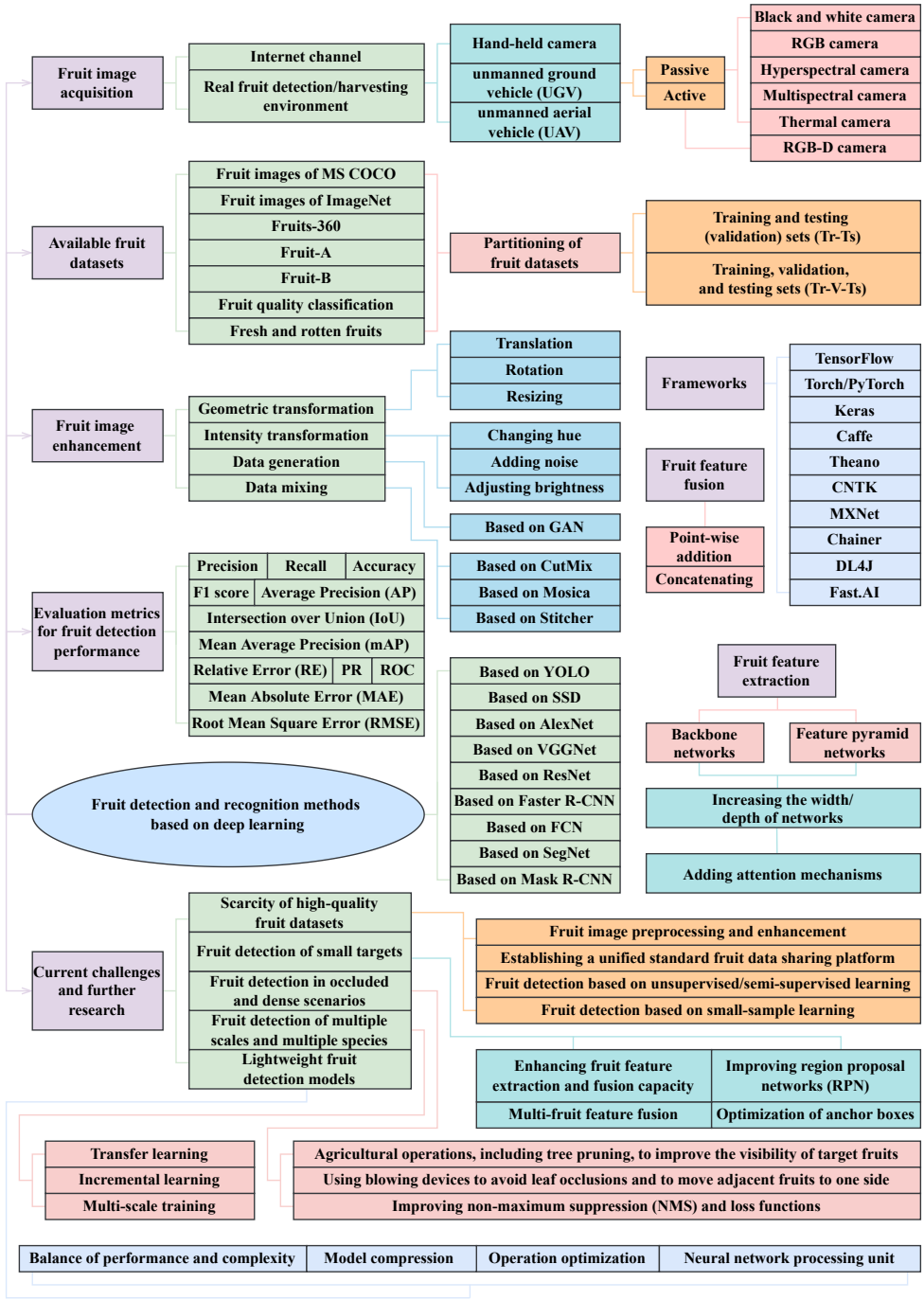


Figure 7. Outline of the article.

2. Fruit Detection and Recognition Based on DL

The concept of DL originated from research on artificial neural networks (ANN), proposed by Canadian researchers Hinton, G.E. and Salakhutdinov, R.R. in 2006 [43]. Since DL has a strong ability to extract high-dimensional features from fruit images, many researchers have conducted extensive and in-depth research on fruit detection and recognition based on DL for automatic harvesting. The basic architecture of DL-based ANN for fruit detection and recognition is shown in Figure 8.

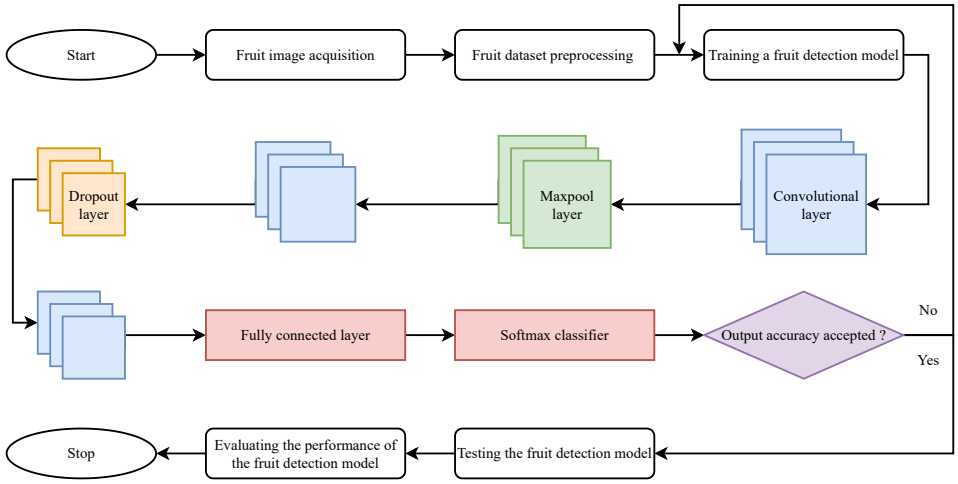


Figure 8. Basic architecture of DL-based ANN for fruit detection and recognition.

CNNs were proposed by American researchers LeCun, Y. et al. in the 1980s [44,45]. They can efficiently capture patterns in multidimensional space. A typical CNN framework for fruit detection and recognition is shown in Figure 9. It includes the convolutional layer (Conv), pooling layer (Pool), nonlinear activation function, and fully connected layer (FC). The convolutional layer is the core of the CNN for fruit feature extraction. Depending on the designed convolution kernel, convolution operations capture fruit image contours and generate corresponding fruit feature maps. In order to reduce the spatial size of the fruit feature maps, the pooling layer performs down-sampling operations by sampling the maximum or average value in a neighborhood range. The nonlinear activation function uses activation functions to process the input data. Neurons in the fully connected layer are connected to all activated neurons in the layer above it. When training the CNN, the model scores categories of predicted images, calculates training loss using selected loss functions, and updates weights through backpropagation functions and gradient descent. The cross-entropy loss function is one of the most widely used loss functions, and the stochastic gradient descent method is the most popular method to address gradient descent.

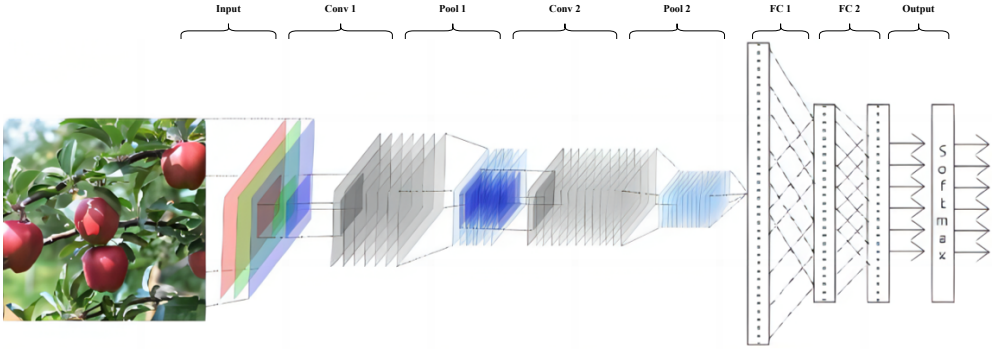


Figure 9. Typical CNN framework for fruit detection and recognition.

Compared with digital image processing and traditional ML techniques, fruit detection and recognition methods based on CNN have great advantages in terms of accuracy. Jahanbakhshi, A. et al., (including Iranian researchers and a researcher based in the UK) (2020) [46] proposed an improved CNN (15, 16, and 18 layers) to detect apparent defects in sour lemons. In comparison to traditional fruit feature extraction methods, such as histogram of oriented gradient (HOG), local binary pattern (LBP), support vector machine (SVM), k-nearest neighbor (KNN), decision tree, and fuzzy classification, the improved CNN was found to outperform these methods, achieving an accuracy of 100%. Bangladeshi researchers Sakib, S. et al., (2019) [47] proposed a fruit detection system using CNN. The Fruits-360 dataset was utilized to evaluate the proposed system. The training accuracy

and testing accuracy are 99.79% and 100%, respectively. In general, fruit detection and recognition methods based on CNN can achieve state-of-the-art (SOTA) accuracy for detecting and recognizing any type of fruit on any background.

Current fruit detection and recognition methods based on DL for automatic harvesting can be classified into two categories: single-stage fruit detection and recognition methods (such as YOLO and SSD) based on regression, and two-stage fruit detection and recognition methods (AlexNet, VGGNet, ResNet, Faster R-CNN, FCN, SegNet, and Mask R-CNN) based on candidate regions. Single-stage methods define fruit detection tasks as regression problems of class confidence and bounding box locations (as shown in Figure 10). They divide input fruit images into a grid of cells, extract fruit feature information through the convolutional layer, and predict object class probabilities and bounding box coordinates for each cell. In contrast, as shown in Figure 11, for two-stage methods, in the first stage, a set of target fruit proposals is generated by the RPN on fruit feature maps produced by the convolutional layer. The RPN generates region of interest (RoI) proposals for each location on the fruit feature maps. Each proposal consists of a fixed-size bounding box and a probability score of containing a target fruit. Based on the scores assigned to these proposals, the top *N* highest-scoring regions are selected as final RoI proposals. To generate RoI proposals, the RPN applies sliding windows of different scales and aspect ratios to fruit feature maps. In the second stage, each final RoI proposal is cropped into a fixed-size feature map using RoI pooling. The maps are then fed into a separate CNN for fruit classification and bounding box regression.

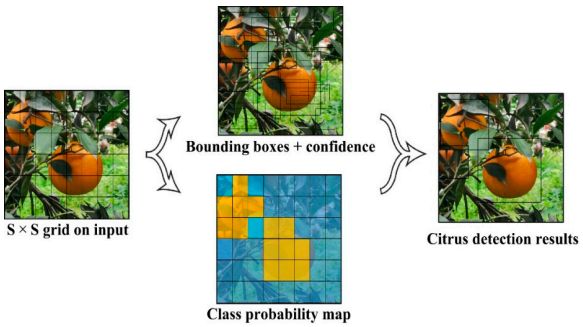


Figure 10. Modeling fruit detection as a regression problem (photos reprinted with permission from ref. [48]. 2022, Chen J.).

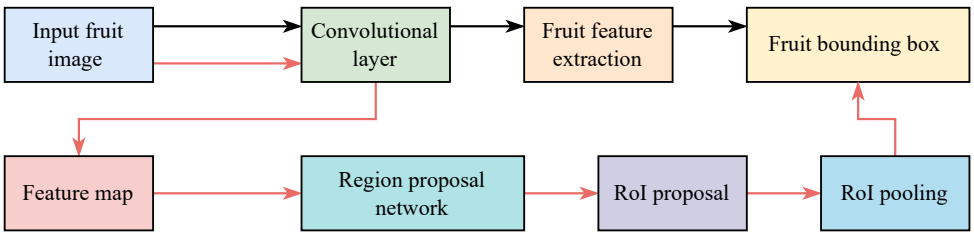


Figure 11. Comparison of one-stage and two-stage fruit detection and recognition methods.

Table 3 compares and analyzes different fruit detection and recognition methods used by various researchers. In the section on "crop, description, and merit", we explain the innovation. In the section on "improvement", we identify the weaknesses and potential improvements. In general, two-stage fruit detection and recognition methods have been shown to achieve higher accuracy than single-stage fruit detection and recognition methods due to their ability to propose more accurate fruit locations. However, they are slower and computationally more intensive than single-stage fruit detection and recognition methods. On the other hand, while single-stage fruit detection and recognition methods are faster and simpler than two-stage fruit detection and recognition methods, they may be less accurate, especially for small target fruits.

Table 3. Comparison of different fruit detection and recognition methods based on DL.

Crops, Description, and Merit	En*	Datasets	Pixels	Sensors	Condition	Improvements	Value (%)
Olive (CNN)/Indian researchers Khosravi, H. et al., (2021) [49] propose a real-time detection method for two olive cultivars in four ripening stages. Adagrad, SGD, SGDM, RMSProp, Adam, and Nadam are evaluated. Nadam shows the best efficiency	√	Train: 14,017; test: 878	256 × 256	Galaxy J6 smartphone camera	Natural lighting	Lighting conditions and fruit image-capturing settings are not considered	Overall accuracy: 91.91; CPU: 12.64 ms; GPU: 4.10 ms
Blueberry (CNN)/Chilean researcher Quiroz, I.A. and Mexican researcher Alférez, G.H. (2020) [50] present a DL solution for image recognition of legacy blueberries in rooting stages	×	Total: 258; train: 168; val: 54; pre: 36	1920 × 1080	Microsoft Lifecam Studio digital camera	Good lighting conditions, not blurred, and without distracting objects in the background	It could use GANs to generate synthetic images that closely resemble real ones, minimizing the need for accessing real data	Accuracy: 86; precision: 86; recall: 88; F1 score: 86
Sour lemon (CNN)/Jahanbakhshi, A. et al., (including Iranian researchers and a researcher based in the UK) (2020) [46] detect apparent defects in sour lemons. Data augmentation and stochastic pooling mechanisms are used to improve detection performance	√	Total: 5456; healthy: 2960; damaged: 2496; train: 70%; val: 30%	16 × 16; 32 × 32; 64 × 64	Camera (Canon, Japan)	A lighting box including two LED lamps	Future work may include accommodating more varied fruit detection conditions	Accuracy: 100
56 diseases infecting 12 plant species (CNN)/Brazilian researcher Barbedo, J.G.A. (2018) [51] studies the effectiveness of DL and TL for plant disease classification	×	Total: 1383; train: 80%; val: 20%	224 × 224 × 3	A variety of digital cameras and mobile devices	Under controlled conditions: 15%; under real conditions: 85%	The number of samples is too small for the CNN to thoroughly capture the characteristics and variations associated with each class	It is a challenge to build fruit databases comprehensive enough for the creation of robust fruit detection models
Strawberry (AlexNet)/Chinese researchers Ni, J. et al., (2021) [52] propose an enhanced AlexNet for strawberry quality evaluation. The size of the convolution kernel is modified. The single convolutional layer is divided into three convolutional layers with different convolution kernels. The BN layer and L2 regularization are used	×	Total: 3006; unripe: 778; medium: 382; fully: 787; bad: 847; malformed: 212; train: 80%; val: 10%; test: 10%	227 × 227	HUAWEI mobile phone	Two different scenes of a field and a laboratory	It is not certain which augmentation method will help improve fruit detection performance	Average accuracy: 90.70; after augmentation: 95.75

Table 3. *Cont.*

Crops, Description, and Merit	En*	Datasets	Pixels	Sensors	Condition	Improvements	Value (%)
Grape bunch (AlexNet)/Italian researchers Marani, R. et al., (2021) [53] investigate the use of DL for grape bunch segmentation in natural fruit images captured using a consumer-grade camera. It is based on the optimal threshold selection of bunch probability maps as an alternative to the conventional minimization of cross-entropy loss for mutually exclusive classes	×	Total: 84; train: 60; val: 24	640 × 480	Intel RealSense R200 RGB-D camera	Fruit images under direct (opposite) sunlight are not considered since they become overexposed, and their colors saturate to white	Depth data could be used to guide the selection of the size N of the moving window for the proposed processing	Mean segmentation accuracy on the bunch class: 80.58; IoU: 45.64
Date fruit (VGGNet)/Saudi Arabian researchers Altaheri, H. et al., (2019) [54] propose an efficient MV framework for date fruit-harvesting robots	×	Total: 8072; 5 date types in different pre-maturity and maturity stages; more than 350 date bunches; belong to 29 date palms	--	RGB video camera	The dataset reflects the challenges, including variations in angles, scales, and illumination conditions	It may lead to confusion in the detection of date fruit maturity, including labeling rules and interference between maturity stages	Type, maturity, and harvesting decision classification accuracies: 99.01, 97.25, 98.59; classification times: 20.6, 20.7, 35.9 ms
Apple (ResNet)/Chinese researchers Wang, D. et al., (2020) [55] develop a remote apple horizontal diameter detection system to achieve automatic measurement of apple growth throughout the entire growth period. The fused convolutional feature network developed can effectively remove complex backgrounds and accurately detect apple edges with near real-time performance	√	Total: 903; train: 743; val: 160; test: 170; 5944 images are eventually obtained through data augmentation; mature red, immature green, semimature	403 × 303	iPhone 7 plus	To prevent distinct edges from forming on the surfaces of apples due to intense natural light, the images are captured on cloudy days or at dusk when the light is not as intense	Future improvements are needed to track the monitored apple in order to achieve the goal of adjusting the camera's shooting angle and selecting seed points automatically	F1 score: 53.1; average run time: 75 ms; mean average absolute error of the apples' horizontal diameters detected: 0.90 mm

Table 3. *Cont.*

Crops, Description, and Merit	En*	Datasets	Pixels	Sensors	Condition	Improvements	Value (%)
Passion fruit (Faster R-CNN)/Chinese researchers Tu, S. et al., (2020) [56] propose a multiple-scale Faster R-CNN approach based on RGB-D images for small passion fruit detection and counting. It detects lower-level features by incorporating feature maps from shallower convolutional feature maps for RoI pooling	√	Total RGB images: 8651; train: 6055; test: 2596; total depth images: 3352; train: 2346; test 1006	1920 × 1080; 512 × 424	Kinect V2	The Kinect V2 sensor is used to avoid strong sunlight and work in shady areas because the ToF technique is unsuitable in strong sunlight conditions	The detection performance of passion fruit in different growth stages could be evaluated and analyzed	Recall: 96.2; precision: 93.1; F1-score: 94.6
Young tomato fruit (Faster R-CNN)/Chinese researchers Wang, P. et al., (2021) [57] propose a method for detecting young tomatoes on near-color backgrounds based on an improved Faster R-CNN with attention mechanisms. Soft non-maximum suppression is used to reduce the missed detection rate of overlapping fruits	×	Total: 2235; train: 80%; val: 10%; test: 10%	3000 × 3000	MI 9 smartphone	Different weather conditions (sunny and cloudy) and different time periods (morning, noon, and evening)	Future work could include accommodating various cultivars of tomatoes and more unstructured environments	mAP: 98.46; average detection time: 84 ms
Lychee (YOLO)/To improve the efficiency of lychee harvesting, Chinese researchers Li, C. et al., (2022) [58] propose a column-comb litchi harvesting method based on K-means 3D clustering partitioning	×	Total: 1049; train: 840; test: 209	1280 × 800; 1280 × 720	Intel RealSense depth camera	Orchard environments (strong light and backlight, sunny and cloudy days, and far and near distances)	Current detection performance are obtained by testing on well-defined fruit images with a limited sample size	Recall: 78.99; precision: 87.43; F1 score: 0.83
Tomato (YOLO)/Chinese researchers Miao, Z. et al. (2022) [59] integrate classic image processing methods with YOLOv5 to increase fruit detection accuracy and robustness	×	Total: 1000; train: 800; val: 200	1920 × 1080; 1280 × 720	Intel RealSense depth camera	Artificial experimental environments	Extended tests and improvements in a real orchard and greenhouse will be the main focus	Average deviation: 2 mm; average operating time: 9 s/cluster

Table 3. *Cont.*

Crops, Description, and Merit	En*	Datasets	Pixels	Sensors	Condition	Improvements	Value (%)
Hass avocado, lemon, apples (SSD)/Vasconez, J.P. et al., (including Chilean researchers and a researcher based in America) (2020) [60] test two of the most common architectures: Faster R-CNN with Inception V2 and SSD with MobileNet. To address the problem of video-based fruit counting, it uses multi-object tracking based on Gaussian estimation	√	Avocado train: 1021; val: 211; test: 211; apple train: 694; val: 191; test: 191; lemon train: 539; val: 202; test: 202	360 × 640	Commercial RGB camera; acquiring at 30 FPS	Hass avocado, lemon, and apple datasets acquired under illumination levels ranging from 1890 to 43,600, 4800 to 52,000, and 3500 to 38,000 lux, respectively	The CNN architectures are highly dependent on the quality of the training set. The results might not be conclusive for other groves with different fruits	SSD with MobileNet, the minimum relative error: 7 (avocados); 13 (apples); 20 (lemons); computing time: 220 ms
Guava (FCN)/Chinese researchers Lin, G. et al., (2019) [61] use a low-cost RGB-D sensor to achieve guava detection and pose estimation. It uses Euclidean clustering to detect all the 3D fruits from the fruit binary maps output by FCN. It also establishes a 3D line segment detection method to reconstruct the branches from the branch binary maps	×	Total: 437; train: 80%; val: 20%	424 × 512	Kinect V2	All kinds of illuminations	Branch is a little difficult to segment	Precision: 98.3; recall: 94.8; 3D pose error: 23.43° ± 14.18°; execution time: 56.5 ms
Lychee clusters (SegNet)/Chinese researchers Li, J. et al., (2020) [62] develop a reliable algorithm based on RGB-D cameras to accurately detect and locate the fruit-bearing branches of multiple lychee clusters. It revises density clustering-based branch extraction and optimal clustering-based parameter analysis	√	Total: 452; train: 80%; val: 20%	1920 × 1080; 512 × 424	Kinect V2	All kinds of illuminations; no artificial shade or lighting interference	Future studies could focus on improving the success rate of picking tasks	Detection accuracy: 83.33; positioning accuracy: 17.29° ± 24.57°; execution time: 464 ms
Apple (SegNet)/Majeed, Y. et al., (including American researchers and researchers based in China) (2020) [63] develop a DL-based semantic segmentation method. Both simple and foreground RGB images are used for training SegNet to segment trunks and branches	√	Total: 509; train: 70%; test: 30%	960 × 540	Kinect V2	Different lighting conditions (sunny, cloudy, and night)	Optimal branches will be selected for training by estimating the essential parameters desired for canopy architecture	Mean accuracy: 89; IoU: 52; boundary-F1-score: 81

Table 3. *Cont.*

Crops, Description, and Merit	En*	Datasets	Pixels	Sensors	Condition	Improvements	Value (%)
Cherry tomato (Mask R-CNN)/Chinese researchers Xu, P. et al., (2022) [64] propose an improved Mask R-CNN for the visual recognition of cherry tomatoes by using depth information and considering the prior adjacent constraint between fruits and stems	√	Total: 3444; train: 80%; val: 20%	640 × 480	Intel RealSense depth camera	Natural conditions	Future work may include reducing the processing time and accommodating more varied conditions	Detection accuracy of fruits: 93.76; accuracy and recall of stems: 89.34 and 94.47; computing time: 40 ms
Strawberry (Mask R-CNN)/Chinese researchers Yu, Y. et al., (2019) [65] perform a visual localization method for strawberry picking points after generating mask images of ripe fruits using Mask R-CNN. ResNet-50 is adopted as the backbone network, combined with the FPN for fruit feature extraction. The RPN is trained end-to-end to create region proposals for each feature map	×	Total: 1900; train: 1520; val: 380; test: 100	640 × 480	Hand-held digital camera	Different periods (morning and afternoon); under varying light intensity (sunny and cloudy conditions); different levels of interference (overlap, occlusion, and oscillation)	Although the average processing frames per second is 8, the speed of the embedded mobile harvesting robot is lower than this result. Therefore, the real-time performance of the model needs to be further improved	Average detection precision: 95.78; recall: 95.41; IoU of instance segmentation: 89.85; average error of picking points: ±1.2 mm

En* represents data enhancement.

2.1. Single-Stage Fruit Detection and Recognition Methods Based on Regression

2.1.1. Fruit Detection and Recognition Methods Based on YOLO

YOLO is one of the most classic and advanced fruit detection algorithms. It can detect and classify target fruits simultaneously in a single image. As shown in Figure 12, YOLO-v1 was the beginning. YOLO-v1 was proposed by American researchers Redmon, J. et al. in 2015 [66]. YOLO-v2 was proposed by American researchers Redmon, J. and Farhadi, A. in 2017 [67]. It included improvements to the structure of YOLO-v1. The K-means clustering algorithm was used to determine the optimal number of anchor boxes and to analyze the relationship between recognition accuracy and speed. Then, they also proposed YOLO-v3 [68], which featured improvements such as the Darknet-53 backbone network and multi-scale prediction. Bochkovskiy, A. et al., (2020) [69] systematically analyzed the processes of data preprocessing and the design of detection and prediction networks. Based on the analysis, they designed an efficient target detector (YOLO-v4) suitable for a single graphics card. YOLO-v5 [70] provided four different sizes of target detectors to meet the needs of different applications. YOLOR [71], YOLOX [72], YOLO-v6 [73], YOLO-v7 [74], and YOLO-v8 [75] also appeared one after another. YOLO-v8 is a SOTA model. It was open-sourced on January 10, 2023. The framework is shown in Figure 13. Specific innovations include a new backbone network, a new anchor-free detection head, and a new loss function that can run on various hardware platforms from CPU to GPU.

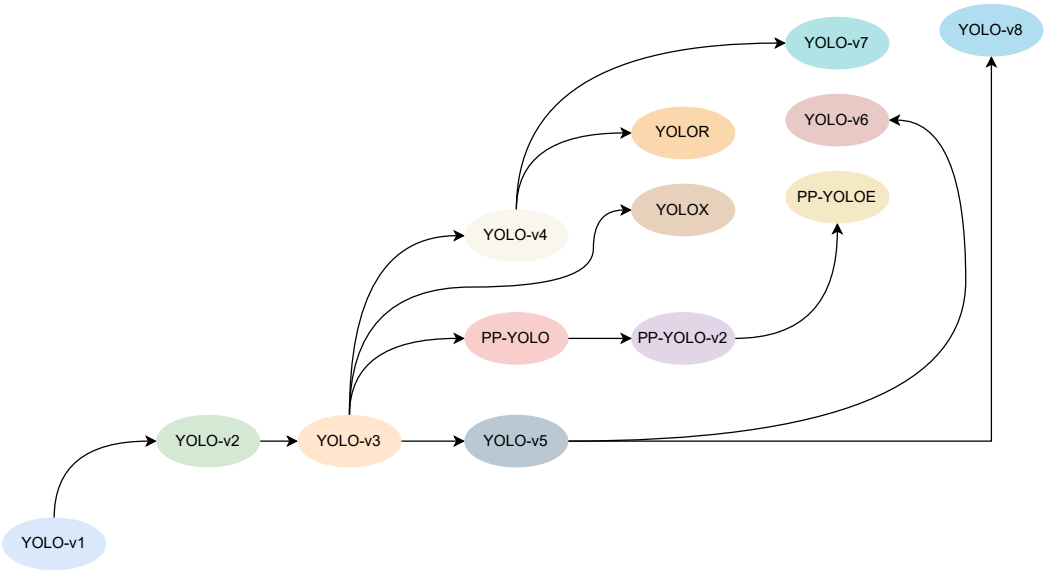


Figure 12. Main research processes of YOLO.

Fruit detection and recognition methods based on YOLO are widely used, by virtue of their advantages. Chinese researchers Xiong, J. et al., (2020) [76] proposed a method based on YOLO-v2 to detect and count mangoes in fruit images taken by an UAV. The processing time is 80ms, and the average detection accuracy is 96.1%. British researchers Birrell, S. et al., (2020) [77] proposed a method based on YOLO-v3 to detect and classify cabbages in four growth stages, achieving a total detection accuracy of 91% and a classification accuracy of 82%. In order to create an even more lightweight fruit detection model, Chinese researchers Li, C. et al., (2022) [58] proposed an improved YOLO-v3-tiny fruit detection model based on K-means 3D clustering partitioning for small and densely packed lychee fruits, and compared it with other fruit detection networks (YOLO-v3-tiny, YOLO-v4, YOLO-v5, and Faster R-CNN). The improved YOLOv3-tiny can recognize lychee fruits more accurately. The check-all rate, check-accuracy rate, and F1 score are 78.99%, 87.43%, and 0.83, respectively. However, fruit detection and recognition methods based on YOLO do not use prior information when predicting fruit positions. This results in a loss of fruit location accuracy. In addition, when YOLO predicts detection results corresponding to each bounding box, it requires that the target fruit's center point must be located inside

the bounding box. This imposes a strong spatial constraint on the prediction process of YOLO and makes fruit detection and recognition methods based on YOLO less effective at detecting small target fruits that appear in groups. In the future, we can input and fuse semantic information (such as fruit scene and context-related information) into fruit detection algorithms to greatly improve fruit detection accuracy. For example, Chinese researchers Miao, Z. et al., (2022) [59] integrated classic image processing methods with YOLO-v5 to increase fruit detection accuracy and robustness. A tomato-harvesting robot can be guided to efficiently harvest truss tomatoes, with an average operating time of 9 s per cluster.

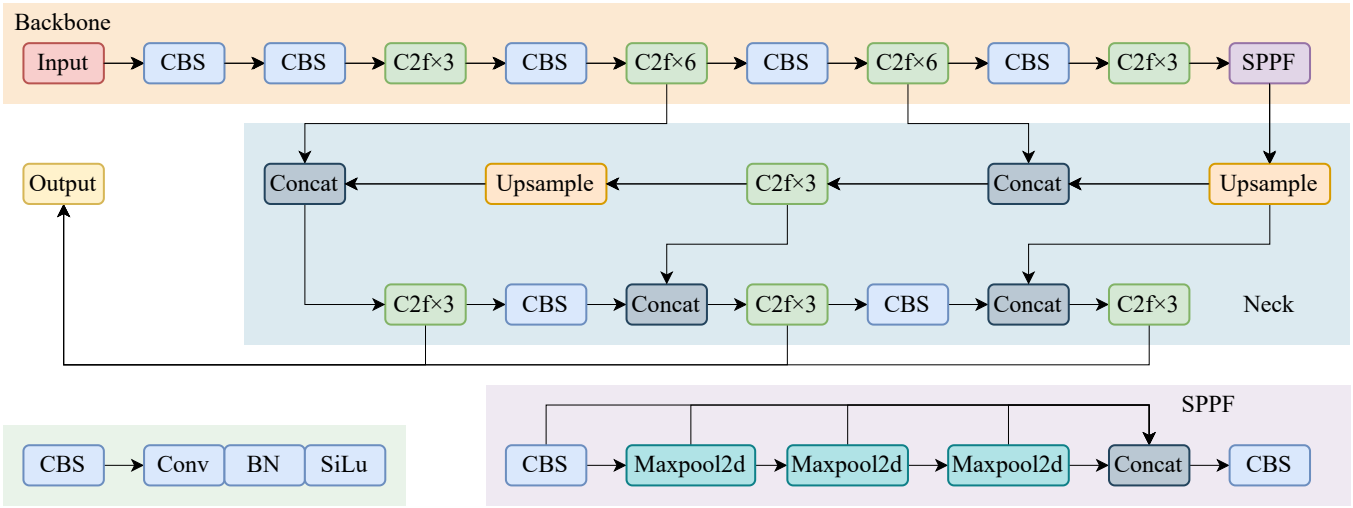


Figure 13. YOLO-v8 framework (image reprinted with permission from ref. [78]. 2023, Lou, H.).

2.1.2. Fruit Detection and Recognition Methods Based on SSD

SSD was proposed by American researchers Liu, W. et al. in 2016 [79]. A typical SSD framework for fruit detection and recognition is shown in Figure 14. It consists of a base network (such as VGG-16) and an additional set of convolutional and pooling layers for fruit feature extraction and detection. It also includes an NMS layer for filtering and selecting the detection results. It borrows the idea of multi-scale fruit detection. Fruit detection tasks are accomplished by generating multiple fruit feature maps of different scales during the fruit detection process. The network model calculates confidence scores for each category in predicted boxes and ground truth boxes, respectively. Then, an NMS operation is performed on the calculated scores of each prediction boxes. Finally, top-ranked prediction boxes are outputted as the final result of fruit detection.

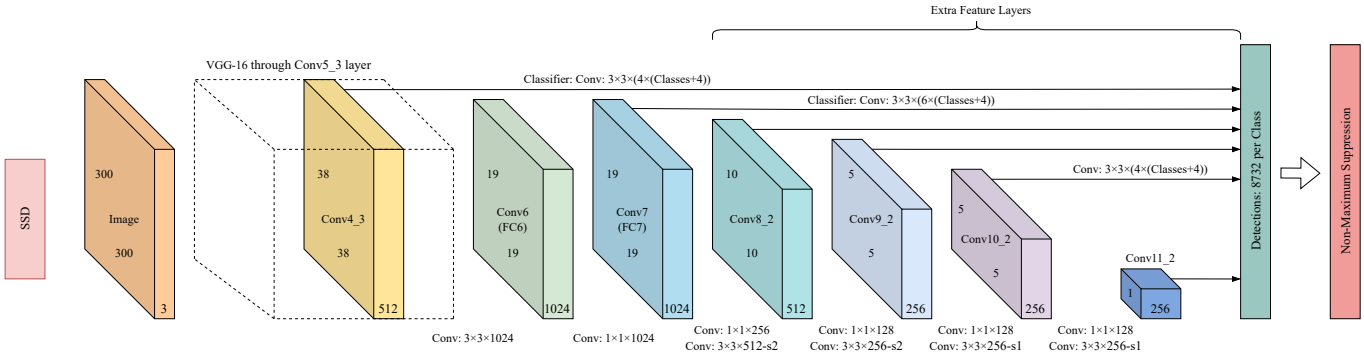


Figure 14. Typical SSD framework for fruit detection and recognition.

Validated on multiple fruit datasets, fruit detection and recognition methods based on SSD have high accuracy and speed. Vasconez, J.P. et al. (including Chilean researchers

and a researcher based in America) (2020) [60] evaluated two of the most widely used architectures (Faster R-CNN with Inception V2 and SSD with MobileNet) for fruit detection. The former achieves 4.55 FPS, whereas the latter achieves a significantly higher performance of approximately 16.67 FPS. However, it is worth noting that fruit detection and recognition methods based on SSD preprocess input fruit images, which may lead to lower fruit detection accuracy for relatively small target fruits when passing through deeper convolutional layers. Chinese researchers Liang, Q. et al., (2018) [80] proposed a real-time detection method for on-tree mangoes based on SSD. New sampling strategies were designed to optimize data augmentation techniques. With optimized data augmentation techniques and default box proposals, SSD outperforms Faster R-CNN in mango detection. Detection results for an almond dataset further confirm the effectiveness of the proposed method. However, it is important to note that the proposed method has deeper layers and a larger number of parameters. This results in slower operation speed and longer computation time.

In general, fruit detection and recognition methods based on SSD also have certain disadvantages. They independently input fruit image features, extracted by different convolutional layers, into corresponding network detection branches. This means that the same fruits in detected images may be identified by bounding boxes of different sizes simultaneously, which can easily lead to the problem of repeated detection. Additionally, each detection branch only operates on target fruits in its respective field, making it difficult to consider the relationship between target fruits of different layers and scales. Therefore, the detection effect of fruit detection and recognition methods based on SSD on small target fruits is not good. Further research could improve SSD in detector frameworks, prediction mechanisms, matching mechanisms, and loss functions.

2.2. Two-Stage Fruit Detection and Recognition Methods Based on Candidate Regions

2.2.1. Fruit Detection and Recognition Methods Based on AlexNet, VGGNet, and ResNet

Typical AlexNet, VGGNet, and ResNet frameworks for fruit detection and recognition are shown in Figure 15. AlexNet was proposed by American researchers Krizhevsky, A. et al. in 2012 [81]. It is the first DL framework that extends CNN to the field of CV. Compared with techniques based on digital image processing and traditional ML, fruit detection and recognition methods based on AlexNet have great advantages in terms of accuracy. Chinese researchers Zhu, L. et al., (2018) [82] proposed a highly effective method for vegetable classification based on AlexNet. The accuracy achieved in the testing set was significantly improved compared to the BP neural network (78%) and SVM classifier method (80.5%), with a remarkable accuracy of 92.1%. Indian researchers Rangarajan, A.K. et al., (2018) [83] demonstrated that the classification accuracy of 13,262 fruit images was 97.49% for AlexNet. Fruit detection and recognition methods based on AlexNet have gained widespread acceptance due to their advantages. By modifying the size of the convolutional kernel and convolutional layer, fruit detection accuracy can be effectively improved. For example, Chinese researchers Ni, J. et al., (2021) [52] improved AlexNet by proposing a new architecture—E-AlexNet. The new architecture enhanced the convolutional layer, reduced kernel size, and used L2 regularization and a BN layer instead of LRN layer. E-AlexNet was compared with the original AlexNet by classifying five strawberry varieties with different qualities. The average recognition accuracy of E-AlexNet was 90.70%, while that of the original AlexNet was 84.50%.

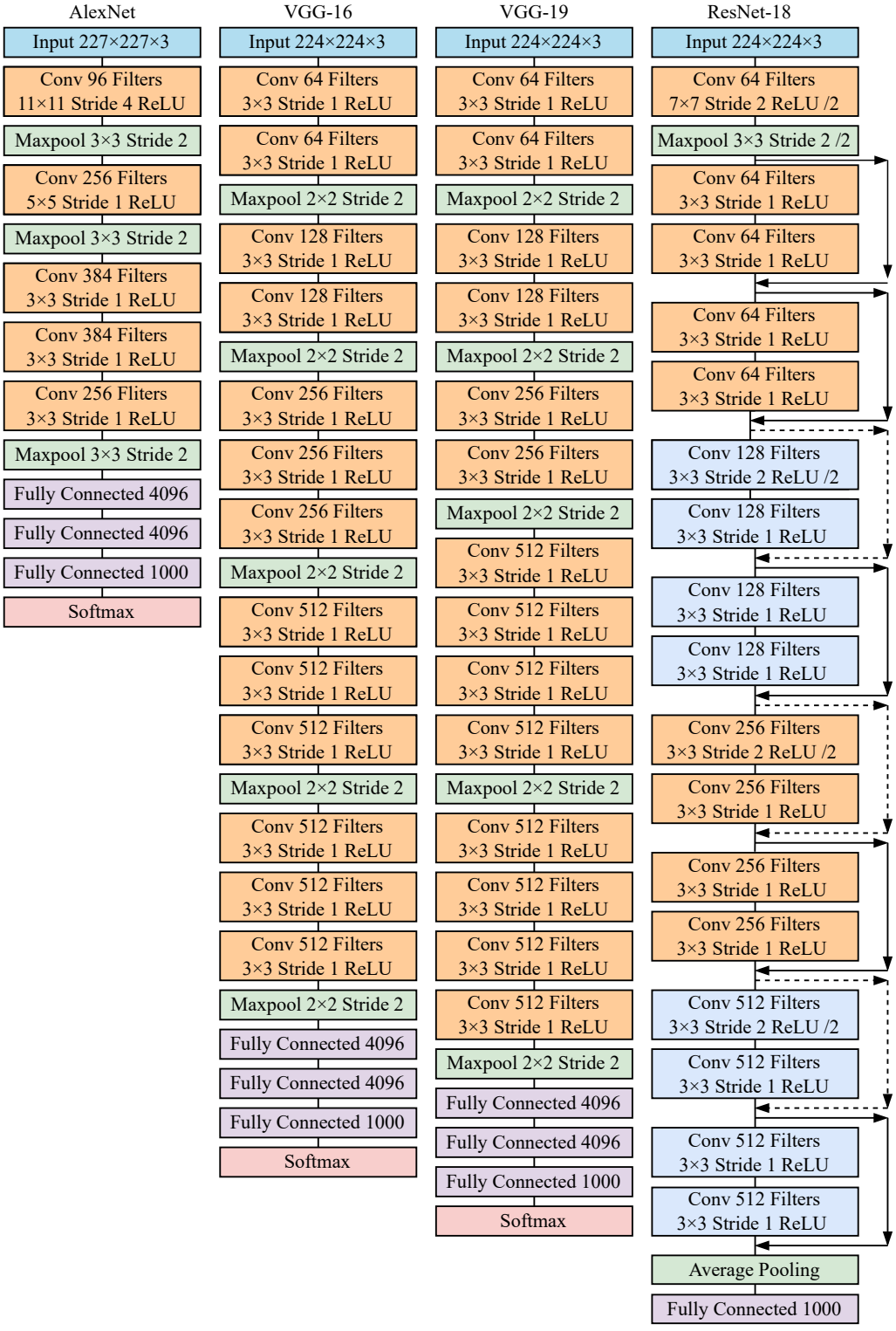


Figure 15. Typical AlexNet, VGGNet, and ResNet frameworks for fruit detection and recognition.

VGGNet was proposed by American researchers Simonyan, K. and Zisserman, A. in 2014 [84]. It has high accuracy in fruit detection and recognition. The biggest improvement of VGGNet is the depth of the network, which has been increased from 8 layers to 16 or 19 layers. Additionally, VGGNet uses a 3×3 convolution kernel to replace the large convolution kernels (11×11, 7×7, 5×5) in AlexNet. In the case of the same receptive field, the accumulation effect of the small convolution kernel is better than that of the large convolution kernel. For example, Indian researchers Mahmood, A. et al., (2022) [85] as-

sessed the effectiveness of two CNN paradigms (AlexNet and VGG-16) in classifying jujube fruits based on their maturity level (unripe, ripe, and overripe). The best accuracy achieved by VGG-16 was 97.65%. Indian researchers Begum, N. and Hazarika, M.K. (2022) [86] used several fruit detection models (VGG-16, VGG-19, Inception V3, ResNet-101, and ResNet-152) to classify three tomato classes (immature, partially mature, and mature). VGG-19 had the best classification accuracy of 97.37% at epoch 50 and batch size 32. Chinese researchers Pérez-Pérez, B.D. et al., (2021) [87] pre-trained seven CNN architectures (AlexNet, VGG-16, VGG-19, ResNet-50, ResNet-101, ResNet-152, and Inception V3) using the ImageNet dataset. VGG-19, with the Adam optimizer, is the one that reported the best accuracy (99.32%).

In order to further improve the accuracy and speed of fruit detection and recognition, Chinese researchers Li, Z. et al., (2020) [88] proposed a fruit recognition and classification method based on VGG-M and VGG-M-BN. On the basis of the original VGG, VGG-M combined the output features of the first two fully connected layers. VGG-M-BN had the BN layer added. The convergence rate of VGG-M-BN is nearly three times faster. The quality of datasets, batch size, and different activation functions also influence fruit recognition and classification accuracy. Firstly, they used VGG-M-BN to train different numbers of vegetable datasets. Recognition accuracy decreases as the quality of datasets decreases. Secondly, by contrasting activation functions, they verified that the rectified linear unit (ReLU) activation function is better than the traditional Sigmoid and Tanh functions in VGG-M-BN. Finally, they verified that the fruit recognition and classification accuracy of VGG-M-BN increases as the batch size increases.

ResNet was proposed by American researchers He, K. et al. in 2015 [89]. It has a high pattern recognition capability. According to the number of backbone layers, ResNet can be further subdivided into ResNet-18, ResNet-50, ResNet-101, and ResNet-152. Fruit detection and recognition methods based on ResNet are widely used, by virtue of their advantages. Helwan, A. et al., (including Lebanese researchers and researchers based in Turkey) (2019) [90] performed automatic segmentation of bananas based on ResNet. Wang, D. et al., (including Chinese researchers and a researcher based in America) (2020) [55] developed a remote apple horizontal diameter detection system based on ResNet to achieve automatic measurement of apples throughout the entire growth period.

Capturing fruit feature information on multiple scales is one way to address the problem that target fruits are overlapped and occluded by branches and leaves. American researchers Rahnemoonfar, M. and Sheppard, C. (2017) [91] optimized the structure of Inception-ResNet. The Improved-Inception-ResNet can count efficiently, even if fruits are under shadow, overlapped, and occluded by leaves. However, although the above fruit detection and recognition methods have high accuracy, they are slow. To address this problem, Australian researchers Kang, H. and Chen, C. (2020) [92] introduced an enhanced deep neural network DaSNet-v2 with ResNet. It has the ability to carry out both detection and instance segmentation of fruits, alongside semantic segmentation of branches. To further improve the speed of fruit detection and meet the real-time requirements of harvesters, Australian researchers Kang, H. and Chen, C. (2019) [93] constructed a multifunctional network for the real-time detection and semantic segmentation of apples and branches. They combined it with the lightweight backbone of ResNet-101 to improve the real-time computational performance of the fruit detection model.

2.2.2. Fruit Detection and Recognition Methods Based on R-CNN, Fast R-CNN, and Faster R-CNN

Typical R-CNN, Fast R-CNN, and Faster R-CNN frameworks for fruit detection and recognition are shown in Figure 16. R-CNN was proposed by American researchers Girshick, R. et al. in 2014 [94]. It is the first algorithm to successfully apply DL to object detection and recognition. Fast R-CNN was proposed by American researcher Girshick, R, one of the creators of R-CNN, in 2015 [95]. It solves some problems of its predecessor, such as slow speed and a large overlap of proposal boxes. One of the key innovations of Fast R-CNN is the "RoI pooling layer", which operates by taking CNN feature maps and

regions of interest as inputs and providing the corresponding features for each region. This allows Fast R-CNN to extract fruit features from all regions of interest in fruit images in a single pass, instead of R-CNN processing each region separately. It significantly improves the speed of fruit detection and recognition. However, Fast R-CNN still requires regions of fruit images to be extracted and provided as inputs to fruit detection models.

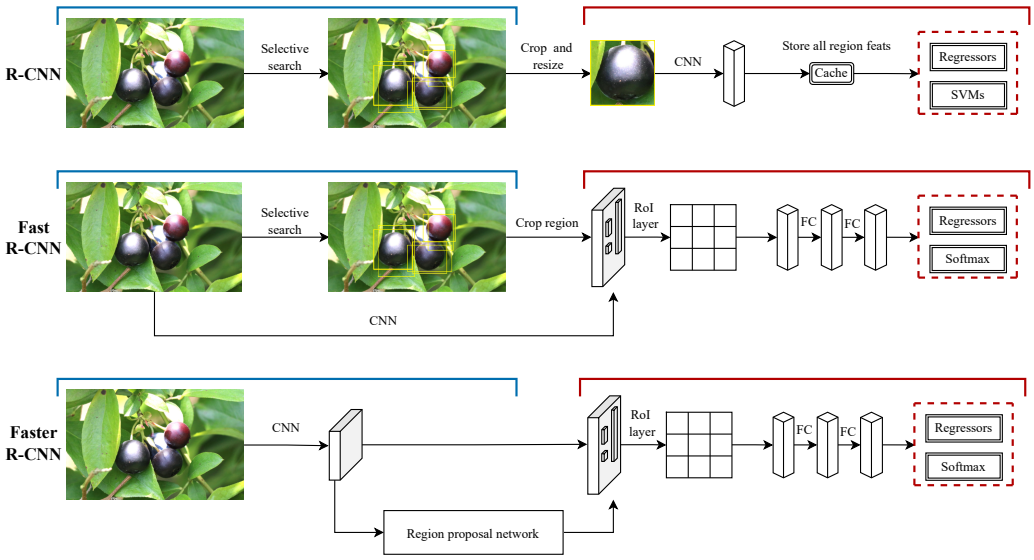


Figure 16. Typical R-CNN, Fast R-CNN, and Faster R-CNN frameworks for fruit detection and recognition.

Faster R-CNN was proposed by American researchers Ren, S. et al. in 2016 [96]. It takes images of fruits as inputs and returns a list of fruit classes, along with their corresponding bounding boxes. Its main innovation is the "RPN". By integrating region detection into the main neural network structure, Faster R-CNN achieves near real-time detection speed with high accuracy and generalization capability.

The fruit detection performance obtained by Faster R-CNN may outperform other networks (YOLOv3, SSD, and ReFCN) [97]. Therefore, fruit detection and recognition methods based on Faster R-CNN are widely used. Chinese researcher Wan, S. and Greek researcher Goudos, S. (2020) [98] proposed a multi-class fruit (apple, mango, and orange) detection method based on Faster R-CNN. The average detection accuracy was 90.72%, and the image processing time was 58ms. Fu, L. et al., (including Chinese researchers and researchers based in America) (2018) [99] proposed a kiwifruit detection method based on Faster R-CNN and evaluated it on kiwifruit images collected in field environments. Zhang, J. et al., (including Chinese researchers and researchers based in America) (2020) [100] used Faster R-CNN to improve a multi-class fruit detection method. They aimed to automatically detect apples, branches, and tree trunks in natural environments and estimate the bobbing locations of collected and captured apples.

Under changing lighting conditions, with low resolution, and with severe occlusion by adjacent fruits and leaves, fruit detection and recognition are very challenging tasks. To solve the problem, Chinese researchers Wang, P. et al., (2021) [57] proposed an improved Faster R-CNN with an attention mechanism based on a near-color background for young tomato detection and recognition. Small target fruit detection and recognition are also very challenging tasks. To solve this problem, in the localization phase, Chinese researchers Cao, C. et al., (2019) [101] proposed an improved loss function based on intersection and ratio for bounding box regression. Additionally, in the recognition phase, the bilinear interpolation method is used to improve the pooling operation of interest regions.

2.2.3. Fruit Detection and Recognition Methods Based on FCN, SegNet, and Mask R-CNN

FCN was proposed by American researchers Long, J. et al. in 2015 [102]. A typical FCN framework for fruit detection and recognition is shown in Figure 17. FCN classifies fruit images at the pixel level and solves the problem of semantic image segmentation. FCN replaces the fully connected layer of the original CNN with the convolutional layer so that the output will be a heatmap instead of a category. Meanwhile, to solve the problem of smaller image size due to convolution and pooling, up-sampling is used to recover image size. Chinese researchers Lin, G. et al., (2019) [61], German researchers Zabawa, L. et al., (2019) [103], and Li, Y. et al., (including Chinese researchers and a researcher based in Germany) (2017) [104] used FCN for the semantic segmentation of guava, grape, and cotton, respectively. Although guava can be segmented easily, the branch is a little difficult to segment. They also compared FCN with SegNet and classification and regression tree classifier (CART). FCN outperforms the other two methods. However, FCN makes some false predictions due to the effects of overlaps and changing lighting conditions. American researchers Chen, S.W. et al., (2017) [105] proposed a method based on FCN for accurate fruit counting in complex natural environments. The method works well even under highly shaded conditions. Furthermore, American researchers Liu, X. et al., (2018) [106] combined deep convolutional segmentation to accurately count sequential images of visible fruits.

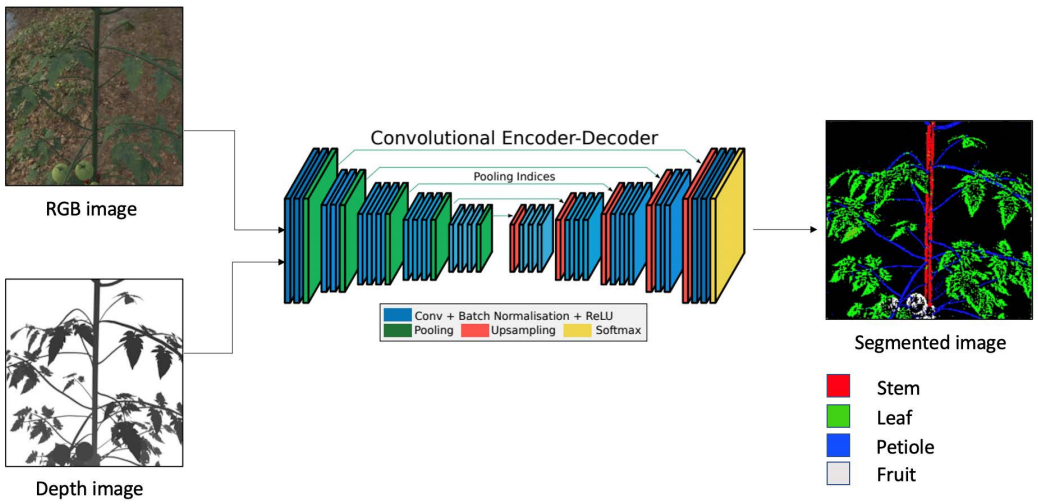


Figure 17. Typical FCN framework for fruit detection and recognition (Source: https://github.com/Alpharouk (accessed on 5 January 2023).

In general, fruit detection and recognition methods based on FCN can accept fruit image inputs of arbitrary size, and the recognition efficiency is higher. They avoid the problem of repeated storage and computational convolution caused by the use of pixel blocks. They reduce the computational effort of the whole fruit detection operation. However, the recognition accuracy is not high because they are insensitive to the details in fruit images, and the classification does not consider inter-pixel relationships.

SegNet was proposed by British researchers Badrinarayanan, V. et al. in 2017 [107]. A typical SegNet framework for fruit detection and recognition is shown in Figure 18. It follows the segmentation idea of FCN and is a symmetric network model with a supervised coding and decoding structure. SegNet can handle fruit image inputs of arbitrary sizes. The coding part reduces the size of input fruit images and the number of parameters stage by stage through maximum pooling, and records the pooling index positions in the fruit images. In order to ensure consistency in resolution between input and output fruit images, decoding processes recover fruit image information through up-sampling. Finally, it outputs semantic segmentation results through the SoftMax classifier. The major difference between SegNet and FCN is the method used for up-sampling low-resolution feature maps to high-resolution feature maps.

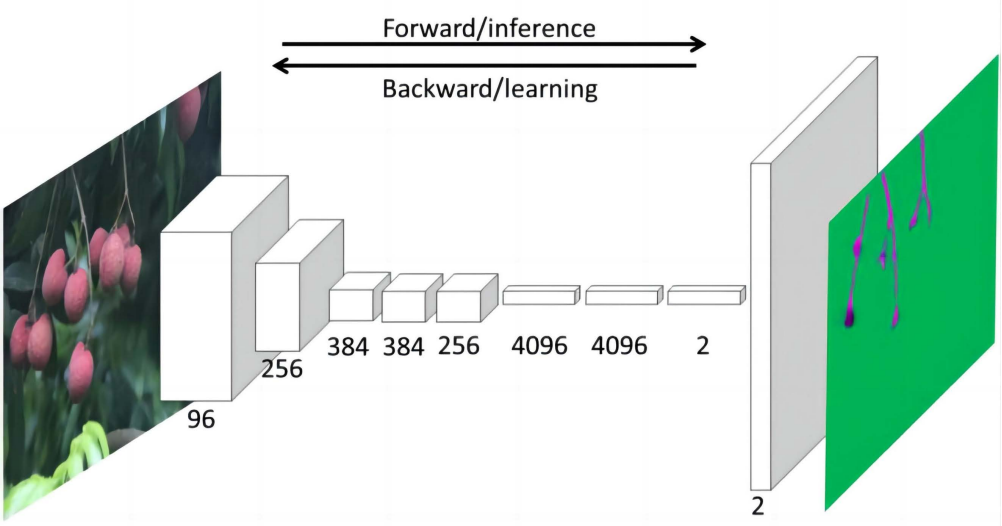


Figure 18. Typical SegNet framework for fruit detection and recognition (image reprinted with permission from ref. [108]. 2020, Peng, H.).

Harvesting robots usually operate in complex natural environments, and the random growth of trunks and branches poses a challenge for fruit detection and recognition. Majeed, Y. et al., (including American researchers and a researcher based in China) (2018) [109] developed a trunk and branch segmentation method using a Kinect V2 sensor. Harvesting robots need to optimize the position of the end effector based on the position and angle between fruits and robot components before approaching, grasping, and cutting target fruits. For this purpose, Dutch researchers Barth, R. et al., (2019) [110] proposed inferring the position of fruits and stems through sparse semantic segmentation in the image plane. In addition, to improve the efficiency of fruit detection and enhance real-time performance, Australian researchers Kang, H. and Chen, C. (2019) [93] used a semantic segmentation network to detect and segment apples and branches in an orchard in real-time. Meanwhile, in order to enable harvesting robots to simultaneously recognize and locate multiple target fruit clusters, Chinese researchers Li, J. et al., (2020) [62] proposed a semantic segmentation method to segment fruit RGB images into three categories: background, fruit, and branch. The method achieved accurate and automatic detection of fruits and branches of multiple lychee clusters in complex natural environments and guided robots to complete continuous harvesting tasks.

Mask R-CNN was proposed by American researchers He, K. et al. in 2017 [111]. A typical Mask R-CNN framework for fruit detection and recognition is shown in Figure 19. It consists of three parts. Firstly, the backbone network extracts fruit feature maps from input fruit images. Secondly, the fruit feature maps outputted by the backbone network are sent to the RPN to generate proposals. Finally, the proposals outputted by the RPN are mapped, and the corresponding target fruit features are extracted from the shared feature maps. These features are outputted to the FC and FCN for fruit classification and instance segmentation, respectively. The process generates classification confidence, bounding boxes, and mask images.

Mask R-CNN combines semantic segmentation with object detection by outputting mask images. This improves the localization accuracy of small target fruits, as well as the prediction accuracy of mask images. Fruit detection and recognition methods based on Mask R-CNN have better robustness and generality for fruit detection and recognition, especially in situations of clustered fruit growth. Chinese researchers Yu, Y. et al., (2019) [65] and Jia, W. et al., (2020) [112] used a Mask R-CNN instance segmentation network model to recognize overlapping strawberries and apples, respectively. They can determine not only categories but also individuals. Since some ripe green tomatoes are similar in color to branches and leaves, shaded by branches and leaves, or overlapped by other tomatoes, accurate detection and localization of these tomatoes is difficult. Chinese researchers

Zu, L. et al., (2021) [113] proposed using Mask R-CNN for the detection and segmentation of ripe green tomatoes. The research results showed the effectiveness of the method. The best model performance was achieved when the IoU was 0.5, and the F1-score of both the testing set bounding box and the masked region reached 92%. Chinese researchers Xu, P. et al., (2022) [64] proposed an improved Mask R-CNN network model for the recognition of cherry tomatoes, considering the prior neighborhood constraint between fruits and stalks.

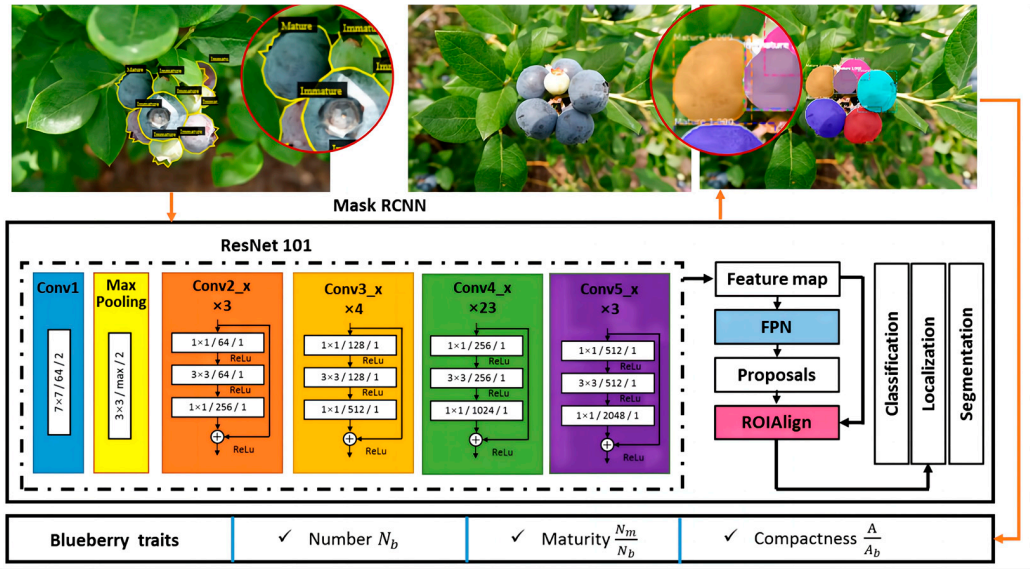


Figure 19. Typical Mask R-CNN framework for fruit detection and recognition (image reprinted with permission from ref. [114]. 2020, Ni X.).

In the future, improvements in fruit detection and recognition methods based on Mask R-CNN should focus on integrating more convolutions to improve performance, and reducing the computational complexity of multi-head attention in the transformer. In addition, multimodal fruit detection methods could be adopted to design Mask R-CNN fruit detection models based on vision, LiDAR, millimeter-wave radar, and other multisensor fusion technologies.

3. Discussion

Currently, there are many factors leading to low accuracy, slow speed, and poor robustness of fruit detection and recognition. They can be summarized in the following aspects: scarcity of high-quality fruit datasets, detection of small target fruits, fruit detection in occluded and dense scenarios, detection of multi-scale and multi-species fruits, and lightweight fruit detection models.

(1) Scarcity of high-quality fruit datasets. Fruit datasets, as signal sources to guide fruit detection algorithms based on DL for information understanding [41], largely determine the final performance of trained fruit detection models. Fruit detection and recognition methods based on DL have two requirements for datasets. One is the sufficiency of data, and the other is the richness of data categories. Fruit datasets are mainly collected in real field environments and through internet channels. A comparison of the advantages and shortcomings of the two collection methods is shown in Table 4. In order to objectively compare the performance of fruit detection and recognition methods, as shown in Table 5, international communities provide some public benchmark datasets. Different fruit datasets have significant differences in the number, quality, and category of images. Researchers can choose compatible fruit datasets for experiments according to their needs. The Fruits-360 dataset is the most commonly used public benchmark dataset. The total number of categories in this dataset is as high as 131, and the total amount of images is considerable.

However, it suffers from the problems of single-image backgrounds, insufficient data diversity, and category imbalance.

Table 4. Comparison of fruit image collection methods.

Types	Methods	Advantages	Shortcomings
Real fruit detection environment	Hand-held camera	High image quality of fruits; close to real scenes	The process of shooting is time-consuming and laborious; fruit image quality is unstable; fruit image quantization and contrast are difficult
	UGV		
	UAV		
Internet channel	--	No need for a camera; easy and fast collection	There are situations such as blurred images and incorrect labels; data cleaning and inspection are required

Table 5. Some frequently used fruit image databases.

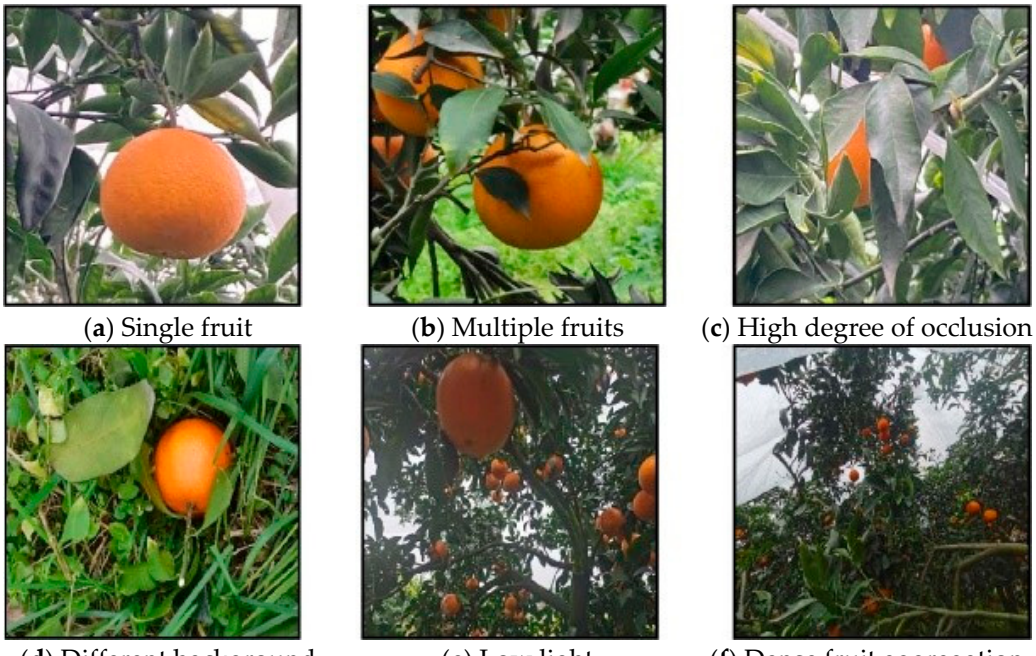
Datasets	Samples			Species	Web-Link	Year
	Total	Training Set	Testing Set			
Fruit images of MS COCO	-	-	-	-	https://cocodataset.org/#download (accessed on 6 March 2023)	2017
Fruit images of ImageNet	-	-	-	-	https://image-net.org/challenges/LSVRC/index.php (accessed on 6 March 2023)	2012
Fruits-360	90,380	67,692	22,688	131 (100 × 100 pixels)	www.kaggle.com/datasets/moltean/fruits (accessed on 16 February 2023)	2020
Fruit-A	22,495	16,854	5641	33 (100 × 100 pixels)	www.kaggle.com/datasets/sshikamaru/fruit-recognition (accessed on 16 February 2023)	2022
Fruit-B	21,000	15,000	vail: 3000 text: 3000	15 (224 × 224 pixels)	www.kaggle.com/datasets/misrakahmed/vegetable-image-dataset (accessed on 16 February 2023)	2021
Fruit quality classification	19,526	-	-	18 (256 × 256/ 192 pixels)	www.kaggle.com/datasets/ryandpark/fruit-quality-classification (accessed on 16 February 2023)	2022
Fresh and rotten fruits	13,599	10,901	2698	6	www.kaggle.com/datasets/sriramr/fruits-fresh-and-rotten-for-classification (accessed on 16 February 2023)	2019

When public benchmark fruit detection datasets cannot meet practical needs, some scholars have created individual fruit datasets to train a fruit detection model for fruit detection and recognition in specific environments. In particular, most of the existing public benchmark fruit detection datasets, such as fruit images of MS COCO and ImageNet, are collected through internet channels. Many of these images differ greatly from actual fruit recognition and harvesting situations. They consist of data from simple scenes, mainly for large and medium-sized fruits. Additionally, datasets for small target fruit detection in complex scenes are especially scarce. International communities might consider continually providing and updating quality public benchmark fruit detection datasets, for example, establishing a unified standard fruit data-sharing platform. The public can upload their fruit images to the platform, and the platform organizes personnel to identify and annotate them.

Due to the scarcity of high-quality fruit datasets, there are potential directions for development in the future: (1) Fruit detection and recognition methods based on small-sample learning may be a key breakthrough. For certain fruit categories for which it is difficult to obtain a large number of samples, this method allows a small number of fruit samples to be selected as representative of new fruit categories. Then, the inherent internal connection between the base fruit class and the new fruit class is used to realize effective knowledge transfer. (2) Fruit detection and recognition methods based on unsupervised learning/semi-supervised learning may be another key breakthrough. Current methods

are mainly based on supervised learning, in which performance relies on a large amount of labeled fruit data. In unsupervised/semi-supervised learning, the model is pre-trained on the data with no or little labeled information.

(2) Detection of small target fruits. Fruits always grow in complex environments with many uncertainties (as shown in Figure 20). We usually define a small target fruit as being smaller than 32 × 32 pixels relative to the absolute size of the image it is in. The difficulties of small target fruit detection are as follows: (1) Limited features that can be extracted. Small target fruits have a small area share and low resolution in images, and they contain limited features themselves. (2) Convolution operations can cause loss of small target features. Fruit detection and recognition methods based on DL extract information of interest about fruits by performing convolution operations on fruit images containing a large amount of redundant information [50]. Fruit feature maps keep shrinking as the number of convolutions increases. If the down-sampling rate is too high, a lot of detailed information for small target fruit detection will be lost. (3) Requirements for the positioning accuracy of the small target fruit bounding boxes are higher. Compared with large target fruits, small target fruits are more sensitive to the offset of prediction boxes and less tolerant of errors. (4) The scale of anchor boxes has not been designed properly. When the scale of anchor boxes is too large, the area of small target fruits is reduced. Therefore, even if small target fruits are within anchor frames, the IoU may not reach the threshold value, resulting in missed detection. In addition, when the receptive field is too large, the fruit detection results are easily disturbed by a large number of other features. When the preset scale of anchor boxes is too close, the spatial difference after down-sampling cannot be guaranteed, resulting in small target fruits being ignored. (5) Sample imbalance. The IoU-based positive and negative samples are considered negative if the IoU is smaller than the threshold. This may lead to small target fruits being ignored in the process of model learning due to the small number of positive samples. Small target fruits usually grow in clusters, which may further cause occlusion and dense detection problems. When small target fruits appear together with other scaled fruits, this gives rise to multi-scale detection problems.



(**a**) Single fruit (**b**) Multiple fruits (**c**) High degree of occlusion

(**d**) Different background (**e**) Low light (**f**) Dense fruit aggregation

Figure 20. Examples of fruit images acquired under different conditions (photos reprinted with permission from ref. [48]. 2022, Chen J.).

Current solutions for small target fruit detection mainly include: (1) Increasing the number of small target fruit samples through data preprocessing and enhancement, such

as in the research presented in [46,49,55,56,60,62–64]; (2) Generating higher-quality small target fruit candidate regions by improving the RPN, such as in the research presented in [56,57,65]; (3) Ensuring the sensory field and small target fruit matching by optimizing anchor boxes, such as in the research presented in [58,61,62]. Combining traditional methods for small target fruit image detection may be a trend for future development. Some small target fruit images contain little information, and they lack the necessary semantic information. Fruit detection and recognition methods based on DL have limited feature extraction ability for small target fruits at the pixel level. Therefore, traditional feature extractors can be introduced to make them more capable of representing features of fruit images. In addition, depth features extracted using CNN can also be combined with traditional methods, such as saliency detection and superpixel segmentation, to obtain a more effective fruit feature representation.

(3) Fruit detection in occluded and dense scenarios. Fruit growth environments are usually complex. There are cases of inter-fruit occlusion and occlusion by shadows or other distractions, such as branches and leaves. The difficulty of detecting fruits in occluded and dense scenarios lies in improving the recall rate of the occluded target fruits [29]. In overlapping cases, the main reasons for missing the detection of obscured target fruits are: (1) Fruits are incomplete, and extractable fruit features are sharply reduced. (2) Overlapping target fruits usually have highly similar features, and it is difficult for fruit detection models to determine whether they belong to different individuals. (3) The NMS post-processing method directly discards objects with lower scores in overlapping regions. For fruit detection in occluded and dense scenarios, the main method of improvement is to enhance fruit feature extraction.

Commonly used methods to enhance the feature extraction capability of fruits are: (1) increasing the width or depth of networks, such as in the research presented in [52,53,57,59]. However, this method will increase the computational load of models. This requires us to strike a balance between performance improvement and computational cost increase. (2) Adding attention mechanisms, such as in the research presented in [55–57]. The introduction of attention mechanisms can help fruit detection models fully consider the connection between each position of target fruits, effectively enhancing the ability of fruit detection models to learn fruit features. Current scholars divide them into the channel attention mechanism and spatial attention mechanism, according to the way the attention acts on feature maps. In fruit detection models, common implementations of attention mechanisms include squeeze-and-excitation networks (SENet) and the convolutional block attention module (CBAM). However, adding an attention mechanism will make fruit detection models more complex and increase convergence time. At the same time, adding an attention mechanism requires careful consideration of whether the design principle of attention, as well as the position and method of action, are suitable for current tasks. Otherwise, it may have a negative impact on fruit detection models. How to reasonably design and implement attention mechanisms, and efficiently use a wide range of environmental features, are important research directions for the future.

(4) Detection of multi-scale and multi-species fruit. Most current fruit detection models are solutions for specific crops. When the detected fruits appear on multiple scales or in multiple types, it is difficult to guarantee the model's generalization ability. For the multi-scale fruit detection problem, the multi-scale training method may be a key breakthrough. It can enable fruit detection models to process fruit information at different scales and improve their ability to capture cross-scale fruit information. Overall, the use of multi-scale fruit prediction networks can make full use of receptive fields, which can effectively alleviate the lack of scale invariance in convolutional neural networks. However, this also increases the number of calculations, resulting in higher demand for hardware facilities. For multi-category fruit detection problems, a common solution is to use transfer learning technology to fine-tune existing models. However, this may result in a loss of detection accuracy for the original fruit categories. Adding a large amount of new category data to the original dataset and retraining a new model can ensure the detection effectiveness of

the original fruit categories. However, every time a new category appears, it needs to be trained from scratch. This not only consumes time and resources, but also cannot satisfy complex, dynamic environments such as farmland orchards. For this problem, we believe that introducing the idea of incremental learning can improve the generalization ability and adaptive learning ability of fruit detection models.

(5) Lightweight fruit detection models. With the continuous development of the field of fruit detection and recognition, researchers are also committed to improving the accuracy of fruit detection models, and the fruit detection models are gradually becoming more complex. For example, some researchers add a super-resolution module to the localization part of fruit detection networks. This may increase the computational load, which, in turn, makes the fruit detection model more dependent on high-performance computing resources. How to further optimize network structures, reduce the number of model parameters, decrease computational complexity, improve running speed, and deploy them on mobile devices are currently hot research topics. Model pruning, quantization, knowledge distillation, and matrix decomposition are effective ways to achieve lightweight and high efficiency. For example, the lightweight MobileNet [11] or ResNet-101 [23] are used to replace the original backbone feature extraction network for fruit feature extraction. At the same time, optimizing operators within frameworks and using AI chips in hardware can greatly accelerate the running speed and parallelism of fruit detection models.

4. Conclusions

Fruit detection and recognition methods based on DL are the mainstream methods for accurate, fast, and robust fruit detection and recognition. These methods are also an important development trend. They are relatively less affected by environments. Our work focuses on providing an overview and review of DL applied to fruit image recognition, mainly in the areas of detection and classification. In order to further define the study areas of this paper, we identify fruit detection and classification tasks as the determination of the class based on their specific types. In general, current fruit detection and recognition methods based on DL can be divided into the following areas: methods based on YOLO, SSD, AlexNet, VGGNet, ResNet, Faster R-CNN, FCN, SegNet, and Mask R-CNN. These methods can also be classified into two categories: single-stage fruit detection and recognition methods (YOLO, SSD) based on regression, and two-stage fruit detection and recognition methods (AlexNet, VGGNet, ResNet, Faster R-CNN, FCN, SegNet, and Mask R-CNN) based on candidate regions.

Most of the current research work is based on two-stage fruit detection and recognition methods. Improvement and application research based on Faster R-CNN (21%) is currently a hotspot. The recognition accuracy of fruit detection and recognition methods based on Faster R-CNN is high, but the recognition speed is limited by complex anchor frame mechanisms. When there are mobile deployment and high recognition speed requirements, fruit detection and recognition methods based on YOLO (17%) are used most frequently. Their recognition speed is fast, but the recognition effect on small target fruits is not very good. In addition, ResNet (11%) is the most popular backbone network, followed by AlexNet (7%). Most of the research focuses on apples (32.14%), followed by tomatoes (8.93%), and citrus (7.14%). These three kinds of fruits are in high demand and yield globally. There are some reasons that make them ideal candidates for automatic harvesting. Firstly, they hang from plants individually, making them easily detectable based on their distinctive features. Secondly, they have no extreme variations in size or weight. Lastly, they are relatively hard and not easily damaged in mechanical operations. However, in terms of fruit dimensions and peduncle length, different cultivars may exhibit different characteristics that can affect fruit detection and recognition performance. This poses challenges for adapting fruit detection and recognition methods for different cultivars. Future work could aim to identify cultivars that are more suitable for automatic harvesting.

The scarcity of high-quality fruit datasets, detection of small target fruits, fruit detection in occluded and dense scenarios, detection of multiple scales and multiple species

of fruits, and lightweight fruit detection models are the current challenges of fruit detection and recognition based on DL for automatic harvesting. The quality and scale of fruit datasets, appropriate improvement strategies, and underlying model architectures all have a significant impact on the detection and recognition performance. For example, fruit data preprocessing can standardize data by cleaning and adjusting them. Fruit data augmentation can effectively expand data and increase data diversity, thereby reducing the dependence on specific factors and improving model robustness. Fruit feature fusion is conducive to alleviating the problem of fruit feature disappearance and improving the detection effect of small target fruits and multi-scale fruits. Building a multi-task learning model, the original fruit detection framework is beneficial for obtaining more fruit information by combining other learning tasks. Moreover, establishing a parameter-sharing mechanism through multi-task learning can significantly improve the performance of fruit detection and recognition. Two-stage fruit detection and recognition methods pursue faster speeds and lighter weights while ensuring fruit detection accuracy. Single-stage fruit detection and recognition methods improve fruit detection accuracy while maintaining the advantages of detection speed and model size. Achieving higher fruit detection performance and a balance between fruit detection precision and speed are current development trends.

Future research should prioritize addressing these current challenges and improving the accuracy, speed, robustness, and generalization of fruit vision detection systems, while reducing the overall complexity and cost. This paper hopes to provide a reference for follow-up research in the field of fruit detection and recognition based on DL for automatic harvesting.

Author Contributions: Conceptualization, F.X.; methodology, F.X.; analysis, F.X. and Y.X.; investigation, F.X., Y.X., H.W. and R.Z.; resources, F.X. and H.W.; data curation, F.X.; writing—original draft preparation, F.X.; writing—review and editing, F.X. and H.W.; visualization, F.X.; supervision, H.W.; project administration, F.X. and H.W.; funding acquisition, H.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of Heilongjiang Province of China (LH2020C047) and China Postdoctoral Science Foundation (2019T120248).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Brown, J.; Sukkarieh, S. Design and Evaluation of a Modular Robotic Plum Harvesting System Utilizing Soft Components. *J. Field Robot.* **2021**, *38*, 289–306. [CrossRef]
2. Yan, B.; Fan, P.; Lei, X.; Liu, Z.; Yang, F. A Real-Time Apple Targets Detection Method for Picking Robot Based on Improved YOLOv5. *Remote Sens.* **2021**, *13*, 1619. [CrossRef]
3. He, L.; Fu, H.; Karkee, M.; Zhang, Q. Effect of Fruit Location on Apple Detachment with Mechanical Shaking. *Biosyst. Eng.* **2017**, *157*, 63–71. [CrossRef]
4. Ji, W.; Zhao, D.; Cheng, F.; Xu, B.; Zhang, Y.; Wang, J. Automatic Recognition Vision System Guided for Apple Harvesting Robot. *Comput. Electr. Eng.* **2012**, *38*, 1186–1195. [CrossRef]
5. Zhao, D.; Lv, J.; Ji, W.; Zhang, Y.; Chen, Y. Design and Control of an Apple Harvesting Robot. *Biosyst. Eng.* **2011**, *110*, 112–122. [CrossRef]
6. Arad, B.; Balendonck, J.; Barth, R.; Ben-Shahar, O.; Edan, Y.; Hellström, T.; Hemming, J.; Kurtser, P.; Ringdahl, O.; Tielen, T.; et al. Development of a Sweet Pepper Harvesting Robot. *J. Field Robot.* **2020**, *37*, 1027–1039. [CrossRef]
7. Lehnert, C.; English, A.; McCool, C.; Tow, A.W.; Perez, T. Autonomous Sweet Pepper Harvesting for Protected Cropping Systems. *IEEE Robot. Autom. Lett.* **2017**, *2*, 872–879. [CrossRef]
8. Bac, C.W.; Hemming, J.; Van Henten, E.J. Stem Localization of Sweet-Pepper Plants Using the Support Wire as a Visual Cue. *Comput. Electron. Agric.* **2014**, *105*, 111–120. [CrossRef]
9. Xiong, Y.; Ge, Y.; Grimstad, L.; From, P.J. An Autonomous Strawberry-Harvesting Robot: Design, Development, Integration, and Field Evaluation. *J. Field Robot.* **2020**, *37*, 202–224. [CrossRef]
10. Xiong, Y.; Peng, C.; Grimstad, L.; From, P.J.; Isler, V. Development and Field Evaluation of a Strawberry Harvesting Robot with a Cable-Driven Gripper. *Comput. Electron. Agric.* **2019**, *157*, 392–402. [CrossRef]

11. Hayashi, S.; Shigematsu, K.; Yamamoto, S.; Kobayashi, K.; Kohno, Y.; Kamata, J.; Kurita, M. Evaluation of a Strawber-ry-Harvesting Robot in a Field Test. *Biosyst. Eng.* **2010**, *105*, 160–171. [CrossRef]
12. Xiong, J.; He, Z.; Lin, R.; Liu, Z.; Bu, R.; Yang, Z.; Peng, H.; Zou, X. Visual Positioning Technology of Picking Robots for Dynamic Litchi Clusters with Disturbance. *Comput. Electron. Agric.* **2018**, *151*, 226–237. [CrossRef]
13. Feng, Q.; Zou, W.; Fan, P.; Zhang, C.; Wang, X. Design and Test of Robotic Harvesting System for Cherry Tomato. *Int. J. Agric. Biol. Eng.* **2018**, *11*, 96–100. [CrossRef]
14. Kondo, N.; Yata, K.; Iida, M.; Shiigi, T.; Monta, M.; Kurita, M.; Omori, H. Development of an End-Effector for a Tomato Cluster Harvesting Robot. *Eng. Agric. Environ. Food* **2010**, *3*, 20–24. [CrossRef]
15. Williams, H.A.M.; Jones, M.H.; Nejati, M.; Seabright, M.J.; Bell, J.; Penhall, N.D.; Barnett, J.J.; Duke, M.D.; Scarfe, A.J.; Ahn, H.S.; et al. Robotic Kiwifruit Harvesting Using Machine Vision, Convolutional Neural Networks, and Robotic Arms. *Biosyst. Eng.* **2019**, *181*, 140–156. [CrossRef]
16. Xiao, F.; Wang, H.; Li, Y.; Cao, Y.; Lv, X.; Xu, G. Object Detection and Recognition Techniques Based on Digital Image Processing and Traditional Machine Learning for Fruit and Vegetable Harvesting Robots: An Overview and Review. *Agronomy* **2023**, *13*, 639. [CrossRef]
17. Fu, L.; Gao, F.; Wu, J.; Li, R.; Karkee, M.; Zhang, Q. Application of Consumer RGB-D Cameras for Fruit Detection and Localization in Field: A Critical Review. *Comput. Electron. Agric.* **2020**, *177*, 105687. [CrossRef]
18. Okamoto, H.; Lee, W.S. Green Citrus Detection Using Hyperspectral Imaging. *Comput. Electron. Agric.* **2009**, *66*, 201–208. [CrossRef]
19. Wachs, J.P.; Stern, H.I.; Burks, T.; Alchanatis, V. Low and High-Level Visual Feature-Based Apple Detection from Multi-Modal Images. *Precis. Agric.* **2010**, *11*, 717–735. [CrossRef]
20. Rehman, T.U.; Mahmud, M.S.; Chang, Y.K.; Jin, J.; Shin, J. Current and Future Applications of Statistical Machine Learning Algorithms for Agricultural Machine Vision Systems. *Comput. Electron. Agric.* **2019**, *156*, 585–605. [CrossRef]
21. Patrício, D.I.; Rieder, R. Computer Vision and Artificial Intelligence in Precision Agriculture for Grain Crops: A Systematic Review. *Comput. Electron. Agric.* **2018**, *153*, 69–81. [CrossRef]
22. Yandun Narvaez, F.; Reina, G.; Torres-Torriti, M.; Kantor, G.; Cheein, F.A. A Survey of Ranging and Imaging Techniques for Precision Agriculture Phenotyping. *IEEE/ASME Trans. Mechatron.* **2017**, *22*, 2428–2439. [CrossRef]
23. Jha, K.; Doshi, A.; Patel, P.; Shah, M. A Comprehensive Review on Automation in Agriculture Using Artificial Intelligence. *Artif. Intell. Agric.* **2019**, *2*, 1–12. [CrossRef]
24. Wolfert, S.; Ge, L.; Verdouw, C.; Bogaardt, M.J. Big Data in Smart Farming—A Review. *Agric. Syst.* **2017**, *153*, 69–80. [CrossRef]
25. Saleem, M.H.; Potgieter, J.; Arif, K.M. Plant Disease Detection and Classification by Deep Learning. *Plants* **2019**, *8*, 468. [CrossRef]
26. Wang, D.; Vinson, R.; Holmes, M.; Seibel, G.; Bechar, A.; Nof, S.; Tao, Y. Early Detection of Tomato Spotted Wilt Virus by Hyperspectral Imaging and Outlier Removal Auxiliary Classifier Generative Adversarial Nets (OR-AC-GAN). *Sci. Rep.* **2019**, *9*, 4377. [CrossRef]
27. Wang, A.; Zhang, W.; Wei, X. A Review on Weed Detection Using Ground-Based Machine Vision and Image Processing Techniques. *Comput. Electron. Agric.* **2019**, *158*, 226–240. [CrossRef]
28. Lv, J.; Xu, H.; Xu, L.; Zou, L.; Rong, H.; Yang, B.; Niu, L.; Ma, Z. Recognition of Fruits and Vegetables with Similar-Color Background in Natural Environment: A Survey. *J. Field Robot.* **2022**, *39*, 888–904. [CrossRef]
29. Li, Y.; Feng, Q.; Li, T.; Xie, F.; Liu, C.; Xiong, Z. Advance of Target Visual Information Acquisition Technology for Fresh Fruit Robotic Harvesting: A Review. *Agronomy* **2022**, *12*, 1336. [CrossRef]
30. Aslam, F.; Khan, Z.; Tahir, A.; Parveen, K.; Albasheer, F.O.; Ul Abrar, S.; Khan, D.M. A Survey of Deep Learning Methods for Fruit and Vegetable Detection and Yield Estimation. In *Big Data Analytics and Computational Intelligence for Cybersecurity*, 2nd ed.; Ouaissa, M., Boulouard, Z., Ouaissa, M., Khan, I.U., Kaosar, M., Eds.; Springer: Cham, Switzerland, 2022; Volume 111, pp. 299–323. [CrossRef]
31. Li, Z.; Yuan, X.; Wang, C. A Review on Structural Development and Recognition-Localization Methods for End-Effector of Fruit-Vegetable Picking Robots. *Int. J. Adv. Robot. Syst.* **2022**, *19*, 17298806221104906. [CrossRef]
32. Darwin, B.; Dharmaraj, P.; Prince, S.; Popescu, D.E.; Hemanth, D.J. Recognition of Bloom/Yield in Crop Images Using Deep Learning Models for Smart Agriculture: A Review. *Agronomy* **2021**, *11*, 646. [CrossRef]
33. Maheswari, P.; Raja, P.; Apolo-Apolo, O.E.; Pérez-Ruiz, M. Intelligent Fruit Yield Estimation for Orchards Using Deep Learning Based Semantic Segmentation Techniques-A Review. *Front. Plant Sci.* **2021**, *12*, 684328. [CrossRef] [PubMed]
34. Bhargava, A.; Bansal, A. Fruits and Vegetables Quality Evaluation Using Computer Vision: A Review. *J. King Saud Univ. Comput. Inf. Sci.* **2021**, *33*, 243–257. [CrossRef]
35. Saleem, M.H.; Potgieter, J.; Arif, K.M. Automation in Agriculture by Machine and Deep Learning Techniques: A Review of Recent Developments. *Precis. Agric.* **2021**, *22*, 2053–2091. [CrossRef]
36. Tang, Y.; Chen, M.; Wang, C.; Luo, L.; Li, J.; Lian, G.; Zou, X. Recognition and Localization Methods for Vision-Based Fruit Picking Robots: A Review. *Front. Plant Sci.* **2020**, *11*, 510. [CrossRef]
37. Jia, W.; Zhang, Y.; Lian, J.; Zheng, Y.; Zhao, D.; Li, C. Apple Harvesting Robot under Information Technology: A Review. *Int. J. Adv. Robot. Syst.* **2020**, *17*, 1729881420925310. [CrossRef]
38. Tripathi, M.K.; Maktedar, D.D. A Role of Computer Vision in Fruits and Vegetables among Various Horticulture Products of Agriculture Fields: A Survey. *Inf. Process. Agric.* **2020**, *7*, 183–203. [CrossRef]

39. Naranjo-Torres, J.; Mora, M.; Hernández-García, R.; Barrientos, R.J.; Fredes, C.; Valenzuela, A. A Review of Convolutional Neural Network Applied to Fruit Image Processing. *Appl. Sci.* **2020**, *10*, 3443. [CrossRef]
40. Koirala, A.; Walsh, K.B.; Wang, Z.; McCarthy, C. Deep Learning-Method Overview and Review of Use for Fruit Detection and Yield Estimation. *Comput. Electron. Agric.* **2019**, *162*, 219–234. [CrossRef]
41. Zhu, N.; Liu, X.; Liu, Z.; Hu, K.; Wang, Y.; Tan, J.; Huang, M.; Zhu, Q.; Ji, X.; Jiang, Y.; et al. Deep Learning for Smart Agriculture: Concepts, Tools, Applications, and Opportunities. *Int. J. Agric. Biol. Eng.* **2018**, *11*, 32–44. [CrossRef]
42. Martín-Martín, A.; Orduna-Malea, E.; Thelwall, M.; Delgado López-Cózar, E. Google Scholar, Web of Science, and Scopus: A Systematic Comparison of Citations in 252 Subject Categories. *J. Informetr.* **2018**, *12*, 1160–1177. [CrossRef]
43. Hinton, G.E.; Salakhutdinov, R.R. Reducing the Dimensionality of Data with Neural Networks. *Science* **2006**, *313*, 504–507. [CrossRef] [PubMed]
44. LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Comput.* **1989**, *1*, 541–551. [CrossRef]
45. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-Based Learning Applied to Document Recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
46. Jahanbakhshi, A.; Momeny, M.; Mahmoudi, M.; Zhang, Y.D. Classification of Sour Lemons Based on Apparent Defects Using Stochastic Pooling Mechanism in Deep Convolutional Neural Networks. *Sci. Hortic.* **2020**, *263*, 109133. [CrossRef]
47. Sakib, S.; Ashrafi, Z.; Sidique, A.B. Implementation of Fruits Recognition Classifier Using Convolutional Neural Network Algorithm for Observation of Accuracies for Various Hidden Layers. *arXiv* **2019**, arXiv:1904.00783. [CrossRef]
48. Chen, J.; Liu, H.; Zhang, Y.; Zhang, D.; Ouyang, H.; Chen, X. A Multiscale Lightweight and Efficient Model Based on YOLOv7: Applied to Citrus Orchard. *Plants* **2022**, *11*, 3260. [CrossRef]
49. Khosravi, H.; Saedi, S.I.; Rezaei, M. Real-Time Recognition of on-Branch Olive Ripening Stages by a Deep Convolutional Neural Network. *Sci. Hortic.* **2021**, *287*, 110252. [CrossRef]
50. Quiroz, I.A.; Alférez, G.H. Image Recognition of Legacy Blueberries in a Chilean Smart Farm through Deep Learning. *Comput. Electron. Agric.* **2020**, *168*, 105044. [CrossRef]
51. Barbedo, J.G.A. Impact of Dataset Size and Variety on the Effectiveness of Deep Learning and Transfer Learning for Plant Disease Classification. *Comput. Electron. Agric.* **2018**, *153*, 46–53. [CrossRef]
52. Ni, J.; Gao, J.; Li, J.; Yang, H.; Hao, Z.; Han, Z. E-AlexNet: Quality Evaluation of Strawberry Based on Machine Learning. *Food Meas.* **2021**, *15*, 4530–4541. [CrossRef]
53. Marani, R.; Milella, A.; Petitti, A.; Reina, G. Deep Neural Networks for Grape Bunch Segmentation in Natural Images from a Consumer-Grade Camera. *Precis. Agric.* **2021**, *22*, 387–413. [CrossRef]
54. Altaheri, H.; Alsulaiman, M.; Muhammad, G. Date Fruit Classification for Robotic Harvesting in a Natural Environment Using Deep Learning. *IEEE Access* **2019**, *7*, 117115–117133. [CrossRef]
55. Wang, D.; Li, C.; Song, H.; Xiong, H.; Liu, C.; He, D. Deep Learning Approach for Apple Edge Detection to Remotely Monitor Apple Growth in Orchards. *IEEE Access* **2020**, *8*, 26911–26925. [CrossRef]
56. Tu, S.; Pang, J.; Liu, H.; Zhuang, N.; Chen, Y.; Zheng, C.; Wan, H.; Xue, Y. Passion Fruit Detection and Counting Based on Multiple Scale Faster R-CNN Using RGB-D Images. *Precis. Agric.* **2020**, *21*, 1072–1091. [CrossRef]
57. Wang, P.; Niu, T.; He, D. Tomato Young Fruits Detection Method under Near Color Background Based on Improved Faster R-CNN with Attention Mechanism. *Agriculture* **2021**, *11*, 1059. [CrossRef]
58. Li, C.; Lin, J.; Li, B.; Zhang, S.; Li, J. Partition Harvesting of a Column-Comb Litchi Harvester Based on 3D Clustering. *Comput. Electron. Agric.* **2022**, *197*, 106975. [CrossRef]
59. Miao, Z.; Yu, X.; Li, N.; Zhang, Z.; He, C.; Li, Z.; Deng, C.; Sun, T. Efficient Tomato Harvesting Robot Based on Image Processing and Deep Learning. *Precis. Agric.* **2022**, *24*, 254–287. [CrossRef]
60. Vasconez, J.P.; Delpiano, J.; Vougioukas, S.; Auat Cheein, F. Comparison of Convolutional Neural Networks in Fruit Detection and Counting: A Comprehensive Evaluation. *Comput. Electron. Agric.* **2020**, *173*, 105348. [CrossRef]
61. Lin, G.; Tang, Y.; Zou, X.; Xiong, J.; Li, J. Guava Detection and Pose Estimation Using a Low-Cost RGB-D Sensor in the Field. *Sensors* **2019**, *19*, 428. [CrossRef]
62. Li, J.; Tang, Y.; Zou, X.; Lin, G.; Wang, H. Detection of Fruit-Bearing Branches and Localization of Litchi Clusters for Vision-Based Harvesting Robots. *IEEE Access* **2020**, *8*, 117746–117758. [CrossRef]
63. Majeed, Y.; Zhang, J.; Zhang, X.; Fu, L.; Karkee, M.; Zhang, Q.; Whiting, M.D. Deep Learning Based Segmentation for Automated Training of Apple Trees on Trellis Wires. *Comput. Electron. Agric.* **2020**, *170*, 105277. [CrossRef]
64. Xu, P.; Fang, N.; Liu, N.; Lin, F.; Yang, S.; Ning, J. Visual Recognition of Cherry Tomatoes in Plant Factory Based on Improved Deep Instance Segmentation. *Comput. Electron. Agric.* **2022**, *197*, 106991. [CrossRef]
65. Yu, Y.; Zhang, K.; Yang, L.; Zhang, D. Fruit Detection for Strawberry Harvesting Robot in Non-Structural Environment Based on Mask-RCNN. *Comput. Electron. Agric.* **2019**, *163*, 104846. [CrossRef]
66. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016. [CrossRef]
67. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017. [CrossRef]
68. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767. [CrossRef]

69. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934. [CrossRef]
70. YOLOv5. Available online: https://github.com/ultralytics/yolov5 (accessed on 7 February 2023).
71. Wang, C.Y.; Yeh, I.H.; Liao, H.Y.M. You Only Learn One Representation: Unified Network for Multiple Tasks. *arXiv* **2021**, arXiv:2105.04206. [CrossRef]
72. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430. [CrossRef]
73. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976. [CrossRef]
74. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. *arXiv* **2022**, arXiv:2207.02696. [CrossRef]
75. YOLOv8. Available online: https://github.com/ultralytics/ultralytics (accessed on 7 February 2023).
76. Xiong, J.; Liu, Z.; Chen, S.; Liu, B.; Zheng, Z.; Zhong, Z.; Yang, Z.; Peng, H. Visual Detection of Green Mangoes by an Unmanned Aerial Vehicle in Orchards Based on a Deep Learning Method. *Biosyst. Eng.* **2020**, *194*, 261–272. [CrossRef]
77. Birrell, S.; Hughes, J.; Cai, J.Y.; Iida, F. A Field-Tested Robotic Harvesting System for Iceberg Lettuce. *J. Field Robot.* **2020**, *37*, 225–245. [CrossRef] [PubMed]
78. Lou, H.; Duan, X.; Guo, J.; Liu, H.; Gu, J.; Bi, L.; Chen, H. DC-YOLOv8: Small-Size Object Detection Algorithm Based on Camera Sensor. *Electronics* **2023**, *12*, 2323. [CrossRef]
79. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016. [CrossRef]
80. Liang, Q.; Zhu, W.; Long, J.; Wang, Y.; Sun, W.; Wu, W. A Real-Time Detection Framework for On-Tree Mango Based on SSD Network. In Proceedings of the International Conference on Intelligent Robotics and Applications, Newcastle, NSW, Australia, 9–11 August 2018. [CrossRef]
81. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
82. Zhu, L.; Li, Z.; Li, C.; Wu, J.; Yue, J. High Performance Vegetable Classification from Images Based on AlexNet Deep Learning Model. *Int. J. Agric. Biol. Eng.* **2018**, *11*, 217–223. [CrossRef]
83. Rangarajan, A.K.; Purushothaman, R.; Ramesh, A. Tomato Crop Disease Classification Using Pre-Trained Deep Learning Algorithm. *Procedia Comput. Sci.* **2018**, *133*, 1040–1047. [CrossRef]
84. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556. [CrossRef]
85. Mahmood, A.; Singh, S.K.; Tiwari, A.K. Pre-Trained Deep Learning-Based Classification of Jujube Fruits According to Their Maturity Level. *Neural Comput. Appl.* **2022**, *34*, 13925–13935. [CrossRef]
86. Begum, N.; Hazarika, M.K. Maturity Detection of Tomatoes Using Transfer Learning. *Meas. Food* **2022**, *7*, 100038. [CrossRef]
87. Pérez-Pérez, B.D.; García Vázquez, J.P.; Salomón-Torres, R. Evaluation of Convolutional Neural Networks' Hyperparameters with Transfer Learning to Determine Sorting of Ripe Medjool Dates. *Agriculture* **2021**, *11*, 115. [CrossRef]
88. Li, Z.; Li, F.; Zhu, L.; Yue, J. Vegetable Recognition and Classification Based on Improved VGG Deep Learning Network Model. *Int. J. Comput. Intell. Syst.* **2020**, *13*, 559–564. [CrossRef]
89. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016. [CrossRef]
90. Helwan, A.; Sallam Ma'aitah, M.K.; Abiyev, R.H.; Uzelaltinbulat, S.; Sonyel, B. Deep Learning Based on Residual Networks for Automatic Sorting of Bananas. *J. Food Qual.* **2021**, *2021*, 5516368. [CrossRef]
91. Rahnemoonfar, M.; Sheppard, C. Deep Count: Fruit Counting Based on Deep Simulated Learning. *Sensors* **2017**, *17*, 905. [CrossRef]
92. Kang, H.; Chen, C. Fruit Detection, Segmentation and 3D Visualisation of Environments in Apple Orchards. *Comput. Electron. Agric.* **2020**, *171*, 105302. [CrossRef]
93. Kang, H.; Chen, C. Fruit Detection and Segmentation for Apple Harvesting Using Visual Sensor in Orchards. *Sensors* **2019**, *19*, 4599. [CrossRef] [PubMed]
94. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 24–27 June 2014. [CrossRef]
95. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015. [CrossRef]
96. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef]
97. Parvathi, S.; Tamil Selvi, S. Detection of Maturity Stages of Coconuts in Complex Background Using Faster R-CNN Model. *Biosyst. Eng.* **2021**, *202*, 119–132. [CrossRef]
98. Wan, S.; Goudos, S. Faster R-CNN for Multi-Class Fruit Detection Using a Robotic Vision System. *Comput. Netw.* **2020**, *168*, 107036. [CrossRef]
99. Fu, L.; Feng, Y.; Majeed, Y.; Zhang, X.; Zhang, J.; Karkee, M.; Zhang, Q. Kiwifruit Detection in Field Images Using Faster R-CNN with ZFNet. *IFAC Pap.* **2018**, *51*, 45–50. [CrossRef]

100. Zhang, J.; Karkee, M.; Zhang, Q.; Zhang, X.; Yaqoob, M.; Fu, L.; Wang, S. Multi-Class Object Detection Using Faster R-CNN and Estimation of Shaking Locations for Automated Shake-and-Catch Apple Harvesting. *Comput. Electron. Agric.* **2020**, *173*, 105384. [CrossRef]
101. Cao, C.; Wang, B.; Zhang, W.; Zeng, X.; Yan, X.; Feng, Z.; Liu, Y.; Wu, Z. An Improved Faster R-CNN for Small Object Detection. *IEEE Access* **2019**, *7*, 106838–106846. [CrossRef]
102. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015. [CrossRef]
103. Zabawa, L.; Kicherer, A.; Klingbeil, L.; Milioto, A.; Topfer, R.; Kuhlmann, H.; Roscher, R. Detection of Single Grapevine Berries in Images Using Fully Convolutional Neural Networks. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA, 16–17 June 2019. [CrossRef]
104. Li, Y.; Cao, Z.; Xiao, Y.; Cremers, A.B. DeepCotton: In-Field Cotton Segmentation Using Deep Fully Convolutional Network. *J. Electron. Imaging* **2017**, *26*, 16. [CrossRef]
105. Chen, S.W.; Shivakumar, S.S.; Dcunha, S.; Das, J.; Okon, E.; Qu, C.; Taylor, C.J.; Kumar, V. Counting Apples and Oranges with Deep Learning: A Data-Driven Approach. *IEEE Robot. Autom. Lett.* **2017**, *2*, 781–788. [CrossRef]
106. Liu, X.; Chen, S.W.; Aditya, S.; Sivakumar, N.; Dcunha, S.; Qu, C.; Taylor, C.J.; Das, J.; Kumar, V. Robust Fruit Counting Combining Deep Learning, Tracking, and Structure from Motion. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018. [CrossRef]
107. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef]
108. Peng, H.; Xue, C.; Shao, Y.; Chen, K.; Xiong, J.; Xie, Z.; Zhang, L. Semantic Segmentation of Litchi Branches Using DeepLabV3+ Model. *IEEE Access* **2020**, *8*, 164546–164555. [CrossRef]
109. Majeed, Y.; Zhang, J.; Zhang, X.; Fu, L.; Karkee, M.; Zhang, Q.; Whiting, M.D. Apple Tree Trunk and Branch Segmentation for Automatic Trellis Training Using Convolutional Neural Network Based Semantic Segmentation. *IFAC Pap.* **2018**, *51*, 75–80. [CrossRef]
110. Barth, R.; Hemming, J.; Van Henten, E.J. Angle Estimation between Plant Parts for Grasp Optimisation in Harvest Robots. *Biosyst. Eng.* **2019**, *183*, 26–46. [CrossRef]
111. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017. [CrossRef]
112. Jia, W.; Tian, Y.; Luo, R.; Zhang, Z.; Lian, J.; Zheng, Y. Detection and Segmentation of Overlapped Fruits Based on Optimized Mask R-CNN Application in Apple Harvesting Robot. *Comput. Electron. Agric.* **2020**, *172*, 105380. [CrossRef]
113. Zu, L.; Zhao, Y.; Liu, J.; Su, F.; Zhang, Y.; Liu, P. Detection and Segmentation of Mature Green Tomatoes Based on Mask R-CNN with Automatic Image Acquisition Approach. *Sensors* **2021**, *21*, 7842. [CrossRef]
114. Ni, X.; Li, C.; Jiang, H.; Takeda, F. Deep Learning Image Segmentation and Extraction of Blueberry Fruit Traits Associated with Harvestability and Yield. *Hortic. Res.* **2020**, *7*, 110. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.