KIWIFRUIT COUNTING USING KIWIDETECTOR AND KIWITRACKER

Yi Xia, Minh Nguyen, Wei Qi Yan

Auckland University of Technology, 1010 Auckland, New Zealand

Abstract. Efficient fruit detection and counting are crucial to improve fruit industrial efficiency and assist the farmers to develop reasonable harvesting strategies in advance while significantly reducing human labors and wastage. In this paper, digital video and image datasets of kiwifruits are collected to train a deep learning-based fruit counting model. The model consists of two molecular algorithms: KiwiDetector for kiwifruit detection and KiwiTracker for kiwifruit tracking and counting. The KiwiDetector algorithm is based on the state-of-the-art YOLOv7 algorithm. The KiwiTracker algorithm is from Kalman filtering and global matching for target tracking. The KiwiDetector module obtained mAP 0.937 after model training based on our collected kiwifruit dataset. The KiwiTracker module has an average counting precision 0.802 after model testing based on different videos containing kiwifruits. The methods can assist us in estimating yields efficiently and provide technical references for agricultural automation.

Keywords: Multiple object tracking, kiwifruit detection, kiwifruit counting.

1 Introduction

The field of artificial intelligence and deep learning is advancing rapidly and has significant potentials to provide reliable technological support for smart agriculture [32] [38]. One critical aspect of intelligent agriculture is the collection of crop information, which necessitates precise and resilient yield estimation models [5] [10] [27]. Accurately estimating crop yield can provide kiwifruit farmers with essential data to optimize their harvest schedules, reduce labor costs, and inform decisions related to fruit pricing and orchard revenue forecasting [23].

Accurate and rapid kiwifruit detection and counting are essential for predicting kiwifruit yields [8] [16]. In traditional fruit image detection systems, visual features such as texture, color and shape are usually extracted and recognized [15]. In most of visual recognition tasks, the images used in the experiments are often captured in a strictly limited environment while eliminating the external environmental influence on the image [12]. However, in real-world environments, images are susceptible to light changes, fruit reflections and occlusions, which affect the recognition accuracy of fruit images to varying degrees, so fruit feature extraction algorithms based on artificial features are not robust enough [17].

However, with the rapid development of deep learning, a great deal of fruit detection recognition and localization algorithms have been proposed [6] [18]. These methods have better performance, generalization ability, and adaptability in real scenarios. The

most prominent algorithms are one-stage object detection algorithms such as YOLO family [14] [35] and two-stage object detection algorithms [17] such as R-CNN [11], Fast R-CNN, and Faster R-CNN [7] [13].

Visual object tracking is another important part of achieving kiwifruit counting [6]. MOT (Multiple Object Tracking) has been widely applied to computer vision and received significant attention from both academia and industry [1] [2]. However, MOT task faces challenges such as appearance changes of the same object in different frames, object occlusion, and varying object numbers. Traditional MOT methods rely mainly on hand-crafted features and target tracking algorithms, which suffer from poor accuracy and generalization. In recent years, the rapid development of deep learning has brought new breakthroughs to MOT. Deep learning-based MOT methods can take use of the automatically learned features to improve tracking accuracy and generalization. Therefore, many new deep learning-based MOT methods have been proposed and achieved good results in practical applications [10]. Most of the current mainstream multitarget tracking models are based on deep learning, such as SORT algorithm [3] and a series of improved algorithms [26]. The main contributions of this paper are summarized as follows:

- We propose a deep learning method for kiwifruit detection, a machine learning model for kiwifruit tracking and counting.
- Our proposed models achieve better performance compared to the state-of-the-art object detection and object tracking models.
- We created a dataset of labelled kiwifruit images that could be employed for deep learning model training.

2 Our Methods

To the best of our knowledge, there are currently no labelled open-source image datasets available for training kiwifruit detection. Thus, we downloaded images and videos of real kiwifruits from the internet and split the videos into frames. In total, we collected 1,500 images as the original dataset. We augmented digital images to improve the training speed of the model. Later, we labelled the image dataset by using the software tools Roboflow and Auto-Orient so that we can label the image dataset. In addition, we resized the original images to 640 × 640 pixels in the Roboflow tool to resolve the inability due to the differences in original images.

Image augmentation improves model performance significantly [31]. Deep learning models are sensitive to visual features of the detected objects and may incorrectly detect the same object if it is mirrored or flipped on the given images. Kiwifruits can appear with different orientations and scales within a given image, making it difficult for the model to detect and count them accurately. Thus, in this paper, we augment the dataset using geometric transformations to avoid overfitting and non-convergence. As shown in Fig. 1, image enhancement mainly consists of random horizontal flipping and random vertical flipping. Random horizontal flipping refers to flip the image with the vertical line at its center. Random vertical flipping shows flipping the image centered on the horizontal center line. Through these methods, the original image dataset was augmented to 1,885 images. We split the augmented dataset into a training dataset, a

verification dataset, and a test dataset having 1,516 images, 246 images, and 123 images, respectively.

(a) Original image (b) Mirrored image (c) flipped image

Fig. 1. The examples of original image and augmented images.

2.1 Overview of the Proposed Methodology

In this paper, we propose a kiwifruit counting model combining YOLOv7 [24] and DeepSort algorithm for solving this challenging problem of kiwifruit yield prediction in real orchards. The structure of the model is shown in Figure 2. We train the KiwiDetector module with the preprocessed kiwifruit images and feed the output bounding box and feature map into the KiwiTracker module to track the kiwifruits in a video and output the total number of kiwifruits detected.

2.2 KiwiDetector Module based on YOLOv7 Algorithm

Kiwifruit detection and counting require an accurate detection model [20]. Finding all the features of interest in given images, as well as their classes and positions, is the goal of visual object detection [34]. Because kiwifruit images have various appearances, shapes, and poses, as well as numerous elements such as lighting and occlusion can interfere the imaging, the object detection has historically been one of the most difficult challenges in the field of computer vision [29]. There are two types of deep learning-based object detection algorithms: Two-stage algorithms and one-stage methods. The two-stage algorithms firstly extract the object features to generate a Region Proposal Network and then detect the objects by using convolutional neural network [36]. Typical two-stage object detection algorithms include R-CNN, SPP-Net [9], Fast R-CNN, Faster R-CNN, and R-FCN. One-stage algorithms do not require the generation of a Region Proposal Network (FPN) and extract features directly in the network to predict visual object classification and location, this leads to faster detection speed [22]. OverFeat, YOLO family [21], SSD, and RetinaNet are examples of conventional one-stage target detection algorithms [37]. YOLOv7 is a popular visual object detection model in YOLO family [28].

In this paper, we proffered YOLOv7 as the feature extraction algorithm to improve the performance of the model in videos with smaller kiwifruit objects and complex backgrounds. In this paper, we name this module as KiwiDetector. Figure 2 shows the structure of KiwiDetector module based on YOLOv7 algorithm. The algorithm consists of three main parts: Input, Backbone and Head. The input layer takes use of the

following tricks: Mosaic data augmentation, adaptive anchor frame calculation and adaptive image scaling, while mosaic data augmentation is use of four images, randomly scaled, randomly cropped and arranged to stitch together to solve the problem of small target detection. In the model training, the deep net outputs a predicted frame based on initial anchor frame, which is then compared with the ground truth (GT) to calculate the differences between the two, and then updated in reverse to iterate over the network parameters [19]. In YOLOv3 and YOLOv4 [4], while training different datasets, the calculation of the initial anchor frame is run through another program.

In this paper, YOLOv7 algorithm is improved based on adaptive image scaling. Since different images have distinct aspect ratios, the scaled padding results in different black border sizes at each end, if more padding is conducted, there is information redundancy that affects inference speed, so the original image is adaptively to add the minimum number of black edges to improve the inference speed. The Backbone layer adopts mainly ELAN and MP structures, which allows deeper networks to be trained and converged efficiently by controlling the shortest and longest gradient paths. The MP structure encapsulates both max pooling and convolutions. The backbone layer consists of CBSConv, E-ELAN, and MPConv layers that alternately halve the aspect and multiply the channels to extract features. In contrast to the previous YOLO algorithm, the KiwiDetector module based on YOLOv7 in this paper integrates the Neck layer and the Head layer. The SPPCSPC module consists of an SPP and a CSP module, where the SPP is employed to increase the receptive field and adapt the algorithm to different resolution images, which is offered to obtain receptive fields throughout max pooling. The two parts are finally combined to reduce the module parameters and increase the accuracy.

The workflow of the Head layer is that after output three feature maps as shown in Figure 1, three unprocessed predictions of different sizes are output through three REP and Conv layers, respectively.

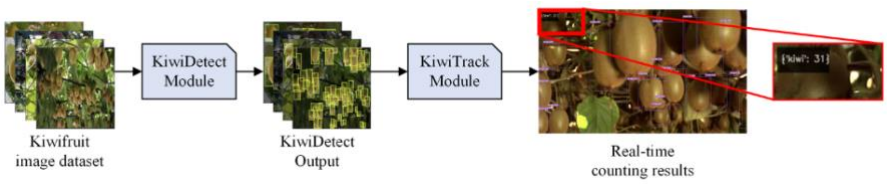


Fig. 2. The architecture of kiwifruit counting model.

2.3 KiwiTracker module based on Multiple Object Tracking

In this paper, a visual object tracking model based on deep learning object detection combined with Kalman filter and Hungarian algorithm is proposed to implement automatic counting of kiwifruits from digital videos [10]. In order to ensure the real-time accuracy of kiwifruit detection and tracking algorithm, the trained KiwiDetector model is put forward for kiwifruit detection with an accurate bounding box and feature map [25]. The Kalman filter algorithm-based predictive tracking of targets. Hungarian algorithm is improved to match targets based on Euclidean distance and Intersection over Union (IoU) to reduce ID duplication and improve kiwifruit counting accuracy.

Figure 4 shows the flowchart of KiwiTracker module in this paper. This algorithm mainly consists of following steps: (1) Detecting the kiwifruits by using the trained KiwiDetector module and get the bounding box and feature map; (2) Predicting the target: The Kalman filter algorithm is proposed to predict the position and motion of the target in the next frame of the video. (3) Matching targets by using the Hungarian algorithm to optimally match targets between the two frames before and after obtained the trajectories of the targets in the video. The target trajectory that fails to match will be temporarily saved and continue to participate in the prediction matching of subsequent frames until the target fails to match for 30 consecutive frames and then is regarded as a fruit and the trajectory is deleted. If the match is successful, the prediction and the counting results are output through the counter. If the current frame is not the last one, the parameters are updated and the object detection step is repeated until the last frame of the current video.

2.4 Experimental evaluation metrics

Evaluation of KiwiDetector. In the kiwifruit detection module proposed in this paper, we take consideration of Precision, Recall and Mean Average Precision (mAP) to evaluate the performance of the proposed models obtained from trained KiwiDetector, where *mAP* is calculated if the *IoU* is set to 0.5 and at different *IoU* thresholds (0.5 to 0.95, in steps of 0.05), respectively. The calculation is shown below, where *TP*, *TN*, *FP*, and *FN* denote the combination of true and predicted categories, and the different predicted results are explained in Table 1, respectively [30]. Higher values of the evaluation metrics indicate better model performance.

$$P = \frac{TP}{TP + FP} \tag{1}$$

$$R = \frac{TP}{TP + FN} \tag{2}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N}\int_{0}^{1} PdR \tag{3}$$

$$\mathrm{ACP} = \frac{\sum_{1}^{\mathrm{n}}\left(1 - \frac{|\mathrm{M} - \mathrm{G}|}{\mathrm{G}}\right)}{\mathrm{n}} \tag{4}$$

Evaluation of KiwiTracker. In this paper, in order to determine the counting accuracy of KiwiTracker, our experimental results were compared to manually counting results to determine the reliability of the KiwiTracker module. The actual number of fruits in the video (ground truth) was firstly obtained by counting each of the 10 videos containing kiwifruits. During the counting process, we firstly recorded the number of kiwifruits in the first frame, and then wrote down the number of new kiwifruits in subsequent frames. The total number of kiwifruits in the video was obtained by firstly recording the number of kiwifruits in the first frame and then the number of new kiwifruits in subsequent frames until the end of this video. The total number of kiwifruits in the video is then obtained, which enables the counting of kiwifruit in the video is possible. Then, the number of kiwifruits in the video is compared with the

result from the human counting result. The accuracy of the kiwifruit counting is verified by comparing the results with the ground truth.

In this paper, we employ the Average Counting Precision (ACP) to assess the counting accuracy of the KiwiTracker algorithm, as illustrated in eq. (4). Specifically, the variables *M* and *G* in eq. (4) represent the number of algorithmic and manual counts, respectively, captured from *n* videos.

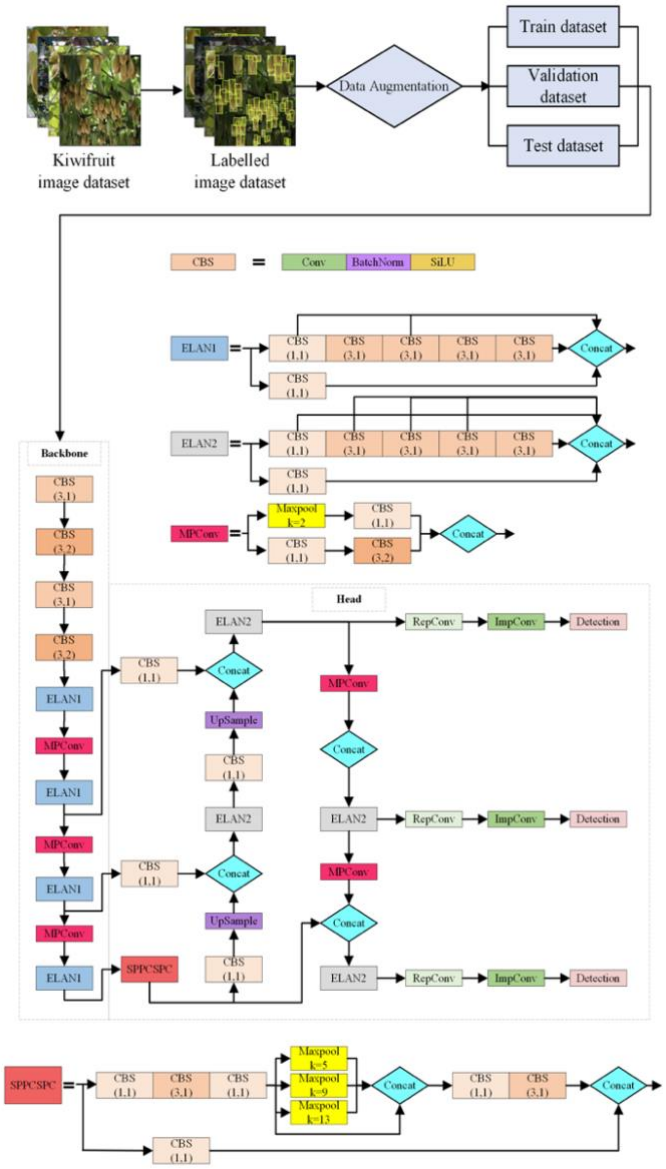


Fig. 3. The structure of dataset and KiwiDetector network.

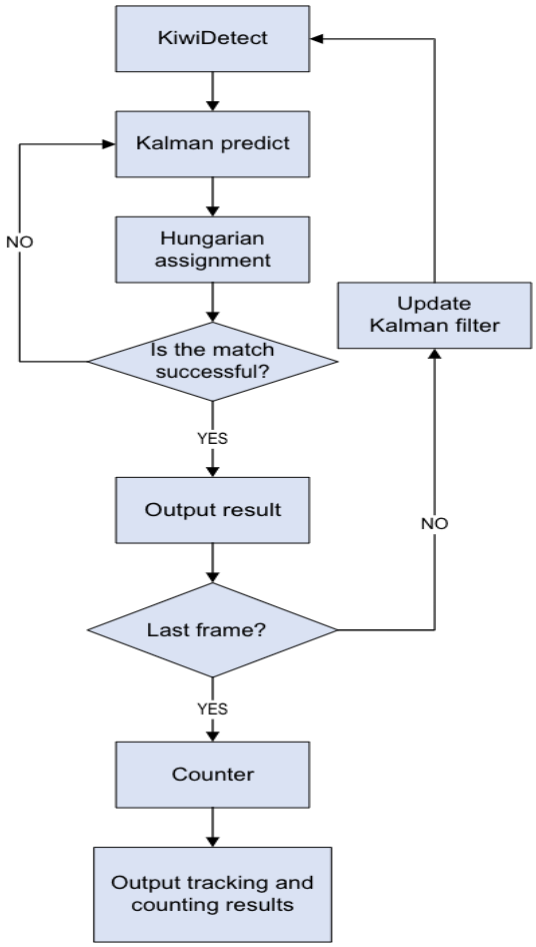


Fig. 4. The flowchart of KiwiTracker for kiwifruits tracking and counting.

3 Results and Discussion

The experiments in this paper were run in a Google Colab notebook with Python version 3.8.16, Pytorch version 1.13.0, and CUDA Version: 11.2. In addition, the GPU for model training was a Colab-supplied Tesla T4 with 16G memory.

3.1 Results of kiwifruit detection

In our experiments with the KiwiDetector module, we have verified the performance of KiwiDetector by comparing three algorithms YOLOv4, YOLOv5, and YOLOv6. The verification results are shown in Table 2. KiwiDetector module shows the best performance in the four evaluation indicators. The results demonstrate that the

KiwiDetector module in this experiment can provide reliable bounding box and feature maps to KiwiTracker.

Figure 5 shows the comparison between the labelled test dataset and the test set predicted by the model. The unmarked kiwifruits in the bottom right corner of Figure 5 (a) are correctly marked by the model in the predicted results. In addition, there are very few missed or incorrect detections in Figure 5 (b). This result demonstrates the ability of the trained KiwiDetector module to detect kiwifruits with a high accuracy.

(a) Examples of the labelled test dataset.

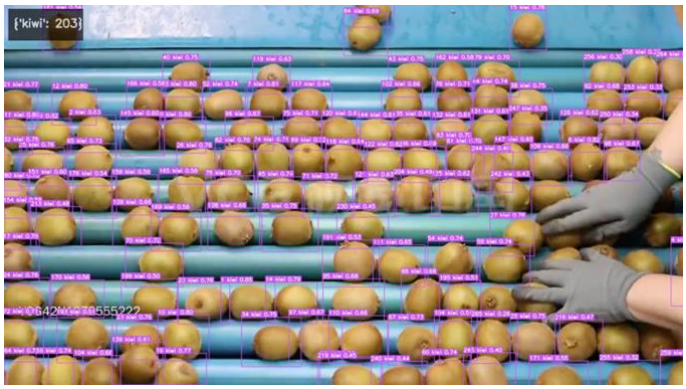
(b) Examples of the prediction test dataset.

Fig. 5. Examples of comparison between predicted results and labelled test dataset.

3.2 Results of kiwifruit counting

Figure 6 shows the performance of KiwiTracker module during counting the static kiwifruits in our collected videos. Figure 6 (a) displays an example of kiwifruit counting in a real orchard video. The top-left corner of the video shows the total number of kiwifruits from the first frame to the current frame, with each labelled kiwifruit being assigned a corresponding ID by the model. Figure 6 (b) shows an example of dynamic kiwifruit counting in a video of kiwifruit sorting conveyor. In this experiment to validate the performance of the KiwiTracker model, we manually counted the number of visible kiwifruits in 10 different videos as the ground truth (GT). Table 3 shows the comparison of our proposed model with Ground Truth in ten videos. We see that our proposed kiwifruit counting model performed well. The main reason for this outcome is that the model counts kiwifruits repeatedly. The final average counting precision of our proposed model is 0.802, and this result can provide reliable technical support for estimating kiwifruit yield.

(a) An example of kiwifruit counting in orchard video.



(b) An example of kiwifruit counting in the conveyor belt.

Fig. 6. An example of kiwifruit detection in the video which shows the total number at the top-left corner of this video frame.

4 Conclusion and Future Work

In this paper, we proposed a deep learning-based model for detecting and displaying kiwifruit quantities. The model has two sub-modules, which are named as KiwiDetector and KiwiTracker. The KiwiDetector makes use of YOLOv7 algorithm based on the iwifruit dataset. After 150 training epochs, KiwiDetector module obtains a mAP@0.5 of 0.937 and a mAP@0.5:0.95 of 0.622. The KiwiDetector module shows the bounding box and feature maps of kiwifruits in each frame were input to KiwiTracker module to obtain the results. The KiwiTracker module predicts kiwifruits based on Kalman filter and measures the similarity of kiwifruit in the predicted results based on Euclidean distance and IoU. Finally, the model rejects kiwifruits that disappear in 30 consecutive frames by using Hungarian algorithm, outputs a kiwifruit with an ID that displays the final outcome after counting. Compared to the results obtained by KiwiTracker with the number of kiwifruits counted manually in the ten videos, the average counting precision of the module was obtained as 0.802. The results of this paper demonstrate the effectiveness of our proposed method.

Table 1. Comparisons of different detection models.

Model	Precision	Recall	mAP@0.5	mAP@0.95
YOLOv4	0.881	0.843	0.904	0.531
YOLOv5	0.902	0.851	0.913	0.585
YOLOv6	0.919	0.876	0.919	0.607
KiwiDetector	**0.933**	**0.889**	**0.937**	**0.622**

Table 2. Counting results of KiwiTracker in 10 videos.

Video ID	Ground Truth	Model Counts	Counting Errors	Counting Precision
1	485	577	+92	0.810
2	1074	1306	+232	0.784
3	712	843	+131	0.816
4	441	521	+80	0.819
5	1379	1691	+312	0.774
6	293	348	+55	0.812
7	491	587	+96	0.804
8	313	369	+56	0.821
9	1098	1349	+251	0.771
10	604	721	+117	0.806
KiwiTracker *ACP* = 0.802				

Despite the promising results of our proposed method, it still has some limitations. Specifically, the performance of our proposed models is suboptimal in videos with

complex backgrounds, particularly in detecting distracting objects like tree trunks, dead foliage, and pedestrians, which can lead to misdetections. To address this issue in future work, we propose augmenting the training dataset with more diverse interference information or categorizing interference information into distinct labels. These steps can help the model to accurately identify kiwifruits by improving its ability to distinguish them from background objects [33].

References

1. An, N., Yan, W.Q.: Anomalies detection and tracking using Siamese neural networks. Master's Thesis, Auckland University of Technology, New Zealand. (2020).
2. An, N., Yan, W.Q.: Multitarget tracking using Siamese neural networks. ACM Transactions on Multimedia Computing, Communications, and Applications. 17, 1–16 (2021).
3. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. 2016 IEEE International Conference on Image Processing (ICIP). (2016).
4. Bochkovskiy, A., Wang, C.-Y., Liao, H.-Y.M.: Yolov4: optimal speed and accuracy of object detection, https://arxiv.org/abs/2004.10934.
5. Fu, Y.H., Yan, W.Q.: Fruit freshness grading using deep learning. Master's Thesis, Auckland University of Technology, New Zealand. (2020).
6. Gao, F., Fang, W., Sun, X., Wu, Z., Zhao, G., Li, G., Li, R., Fu, L., Zhang, Q.: A novel apple fruit detection and counting methodology based on deep learning and trunk tracking in modern Orchard. Computers and Electronics in Agriculture. 197, 107000 (2022).
7. Gao, F., Fu, L., Zhang, X., Majeed, Y., Li, R., Karkee, M., Zhang, Q.: Multi-class fruit-on-plant detection for Apple in SNAP system using faster R-CNN. Computers and Electronics in Agriculture. 176, 105634 (2020).
8. Gu, Q., Yang, J., Kong, L., Yan, W.Q., Klette, R.: Embedded and real-time vehicle detection system for challenging on-road scenes. Optical Engineering. 56, 063102 (2017).
9. Le, H., Nguyen, M., Yan, W.Q., Nguyen, H.: Augmented reality and machine learning incorporation using yolov3 and Arkit. Applied Sciences. 11, 6006 (2021).
10. Liu, W., Li, Y., Tomasetto, F., Yan, W., Tan, Z., Liu, J., Jiang, J.: Non-destructive measurements of Toona sinensis chlorophyll and nitrogen content under drought stress using near infrared spectroscopy. Frontiers in Plant Science. 12, (2022).
11. Liu, Y., Yang, G., Huang, Y., Yin, Y.: Se-mask R-CNN: An improved mask R-CNN for Apple Detection and segmentation. Journal of Intelligent & Fuzzy Systems. 41, 6715–6725 (2021).
12. Liu, X., Yan, W.Q.: Vehicle-related distance estimation using customized Yolov7. Image and Vision Computing. 91–103 (2023).
13. Luo, Z., Yan, W.Q., Nguyen, M.: Kayak and sailboat detection based on the improved Yolo with Transformer. 2022 The 5th International Conference on Control and Computer Vision. (2022).
14. Luo, Z., Yan, W.Q., Nguyen, M.: Sailboat and kayak detection using deep learning methods. Master's Thesis, Auckland University of Technology, New Zealand. (2022).
15. Lv, J., Ni, H., Wang, Q., Yang, B., Xu, L.: A segmentation method of Red Apple Image. Scientia Horticulturae. 256, 108615 (2019).
16. Massah, J., Asefpour Vakilian, K., Shabanian, M., Shariatmadari, S.M.: Design, development, and performance evaluation of a robot for yield estimation of Kiwifruit. Computers and Electronics in Agriculture. 185, 106132 (2021).

17. Pan, C., Liu, J., Yan, W.Q., Cao, F., He, W., Zhou, Y.: Salient object detection based on visual perceptual saturation and two-stream hybrid networks. IEEE Transactions on Image Processing. 30, 4773–4787 (2021).
18. Pan, C., Yan, W.Q.: Object detection based on saturation of Visual perception. Multimedia Tools and Applications. 79, 19925–19944 (2020).
19. Qi, J., Nguyen, M., Yan, W.Q.: Small visual object detection in smart waste classification using Transformers with deep learning. Image and Vision Computing. 301–314 (2023).
20. Rahnemoonfar, M., Sheppard, C.: Deep count: Fruit counting based on deep simulated learning. Sensors. 17, 905 (2017).
21. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2016).
22. Shen, D., Chen, X., Nguyen, M., Yan, W.Q.: Flame detection using Deep Learning. 2018 4th International Conference on Control, Automation and Robotics (ICCAR). (2018).
23. Song, Z., Tomasetto, F., Niu, X., Yan, W.Q., Jiang, J., Li, Y.: Enabling breeding selection for biomass in slash pine using UAV-based imaging. Plant Phenomics. 2022, (2022).
24. 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, https://arxiv.org/abs/2207.02696.
25. Wang, D., He, D.: Apple Detection and instance segmentation in natural environments using an improved mask scoring R-CNN model. Frontiers in Plant Science. 13, (2022).
26. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a Deep Association metric. 2017 IEEE International Conference on Image Processing (ICIP). (2017).
27. Xiao, B., Nguyen, M., Yan, W.Q.: Apple ripeness identification using Deep Learning. Communications in Computer and Information Science. 53–67 (2021).
28. Xia, Y., Nguyen, M., Yan, W.Q.: A real-time kiwifruit detection based on improved Yolov7. Image and Vision Computing. 48–61 (2023).
29. Xin, C., Nguyen, M., Yan, W.Q.: Detection and recognition for multiple flames using deep learning. Master's Thesis, Auckland University of Technology, New Zealand. (2018).
30. Xin, C., Nguyen, M., Yan, W.Q.: Multiple flames recognition using Deep Learning. Handbook of Research on Multimedia Cyber Security. 296–307 (2020).
31. Xing, J.W., Yan, W.Q.: Traffic sign recognition from digital images by using deep learning. Master's Thesis, Auckland University of Technology, New Zealand. (2020).
32. Yan, W.Q.: Computational methods for deep learning: Theoretic, practice and applications. Springer (2021).
33. Yan, W.Q.: Introduction to intelligent surveillance. Springer International Publishing (2019).
34. Zhang, Q., Yan, W.Q.: Currency detection and recognition based on Deep Learning. 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). (2018).
35. Zhang, Y., Shen, K.J., He, Z.F., Pan, Z.S.: Yolo-infrared: Enhancing Yolox for infrared scene. Journal of Physics: Conference Series. 2405, 012015
36. Zhao, K., Yan, W.Q.: Fruit detection from digital images using CenterNet. Communications in Computer and Information Science. 313–326 (2021).
37. Zhao, K., Yan, W.Q.: Fruit detection using CenterNet. Master's Thesis, Auckland University of Technology, New Zealand. (2021).
38. Zhu, Y., Yan, W.Q.: Parasite detection from digital images using Deep Learning. Machine Learning and AI Techniques in Interactive Medical Image Analysis. 124–134 (2022).