Computers and Electronics in Agriculture 171 (2020) 105302

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Fruit detection, segmentation and 3D visualisation of environments in apple orchards

Hanwen Kang, Chao Chen*

Department of Mechanical and Aerospace Engineering, Monash University, Melbourne, Australia

ARTICLE INFO

Keywords:
Fruit detection
Fruit segmentation
Branch segmentation
Deep learning
Robotic harvesting

ABSTRACT

Development of an accurate and reliable fruit detection system is a challenging task. There are many complex conditions in orchard environments, such as changing illumination, appearance variation, and occlusion. Robotic vision is required to understand the working environments from the sensory data and guide the robotic arm to detach the fruits. In our previous work, a deep neural network DaSNet-v1 was developed to perform detection and segmentation on fruits and branches in orchard environments. However, semantic segmentation returns the mask for each class instead of each object. Segmentation on each fruit is important as it can provide abundant information of each object, especially for those overlapped fruits. This work presents an improved deep neural network DaSNet-v2, which can perform detection and instance segmentation on fruits, and semantic segmentation on branches. DaSNet-v2 is tested and validated by experimental results obtained from field-testing in an apple orchard. From the experiment results, DaSNet-v2 with resnet-101 achieves 0.868, 0.88 and 0.873 on recall and precision of detection, and accuracy of instance segmentation on fruits, and 0.794 on the accuracy of branches segmentation, respectively. DaSNet-v2 with light-weight backbone resnet-18 achieves 0.85, 0.87 and 0.866 on recall and precision of detection, and accuracy of instance segmentation on fruits, and 0.775 on the accuracy of branches segmentation, respectively. The average running time and weight size of light-weight DaSNet-v2 are 55 ms and 8.1 M, respectively. Experimental results show DaSNet-v2 can robustly and efficiently perform the vision sensing for robotic harvesting in apple orchards.

1. Introduction

Nowadays, with the increasing cost and difficulty in availability of the labour resource (ABARES, 2018), the agricultural industry requires transformation from the labour-intensive industry to the technology-intensive industry. Robotic technology has shown a promising prospect in terms of improving the efficiency and yield of agriculture production. Different from the harvesting equipments which are designed to perform autonomous harvesting of commercial crops such as wheat and soybean in the structured working environments, to design a robotic system for automatic harvesting of fruits in orchard environments is much more challenging (Vasconez et al., 2019). Among the challenging issues in developing a fruit harvesting robot, the vision system is a crucial issue since it senses the working environment and guides the robotic arm to detach the fruits. Due to the complex conditions in real working environments, issues such as densely arranged branches and fruits in orchards should be taken into account when designing a fruit harvesting robots. In other words, fruit harvesting robots are required to understand the working environment to increase the rate of success during the harvesting (Zhao et al., 2016). Meanwhile, other environmental factors, such as various illumination conditions, changing object appearances, and occlusion or overlap of objects, can also critically affect the performance of the robotic vision system. In the previous work (Kang and Chen, 2019), a multi-task deep neural network DaSNet-v1 was developed, which can perform detection and semantic segmentation on fruits and branches in orchard environments. However, semantic segmentation can only segment the images into different classes while lacking capability of segmenting each object within the class (which also known as instance segmentation). Instance segmentation of each fruit is important as it can provide geometric property (shape and size) of each fruit, and such information can be used to compute the poses (RGB-D camera applied) of the objects. Therefore, further development of the techniques to obtain instance segmentation of each object is demanded.

In this work, an improved multi-task deep neural network DaSNet-v2 is developed to perform multi-task vision sensing for robotic harvesting in apple orchards. Firstly, DaSNet-v2 combines multi-task into the network architecture, which can perform detection and instance segmentation on fruits, and semantic segmentation on branches. Secondly, the network architecture of DaSNet-v2 is optimised compared to the DaSNet-v1 to

* Corresponding author.
E-mail addresses: hanwen.kang@monash.edu (H. Kang), chao.chen@monash.edu (C. Chen).

https://doi.org/10.1016/j.compag.2020.105302
Received 21 November 2019; Received in revised form 19 February 2020; Accepted 20 February 2020
Available online 17 March 2020
0168-1699/

obtain better performance on detection and segmentation. Additional, a light-weight designed DaSNet-v2 (light-weight backbone applied) is trained and validated in this work to ensure the computational availability of the model on the embedded computing devices. DaSNet-v2 is tested and validated by experimental results obtained from field-test in an apple orchard. 3D visualisation of experimental results by means of the DaSNet-v2 is also illustrated in this work.

The rest of the paper is organised as follows. Section 2 reviews the related works. Sections 3 and 4 introduce the methodology and experiment of the work, respectively. In Section 5, the conclusions and future work are presented.

2. Literature review

Vision sensing in fruit orchards has been extensively studied. Currently, there are two classes of approaches: traditional machine-learning based algorithms and deep-learning based algorithms. Traditional machine-learning based algorithms apply feature descriptors to extract object features from the sensory data and machine-learning based classifier to perform classification, detection, or segmentation (Kapach et al., 2012). There are numbers of work which have applied traditional machine-learning based algorithms on vision sensing in agricultural applications (Vibhute and Bodhe, 2012; Zhao et al., 2016). Nguyen et al. (2016) applied colour features and geometric features to encode the appearance of the red apples. Then a clustering algorithm based on Euclidean distance in feature space is used to segment and detect the fruits from the input images. The similar processing techniques of performing segmentation and detection in vision sensing in orchard environment are also presented in several works (Zhou et al., 2012; McCool et al., 2016; Lin et al., 2019a; Liu et al., 2018). Recently, Wang and Lihong (2018) applied multiple image features and Latent Dirichlet Allocation (LDA) model to perform unsupervised instance/semantic segmentation of the plants and fruits in the greenhouse environments.

The development of deep-learning algorithms is more recent. Compared to the traditional machine-learning based algorithms, deep-learning based algorithms have demonstrated higher accuracy on detection, and segmentation (Han et al., 2018). Deep-learning based algorithms can be classified into two classes: two-stage detector and one-stage detector (Lin et al., 2017). The representative work of the two-stage detector is the Region Convolution Neural Network (RCNN), which includes fast/faster-RCNN (Girshick, 2015; Ren et al., 2015) and mask-RCNN (He et al., 2017). Faster-RCNN applies Region Proposal Network (RPN) and Region of Interest (RoI) pooling to combine the RoI searching and classification into a single network architecture, which increases the computational efficiency of the model. Mask-RCNN further combines instance segmentation into the detection network, which allows the network to segment the corresponding area for each object within the images. On the other hand, the representative work of the one-stage detector is You Only Look Once (YOLO) (Redmon and Farhadi, 2018) and Single Shot Detection (SSD) (Liu et al., 2016). One-stage detector predicts the object on each grid of feature maps, and it achieves similar performance with improved computational efficiency compared to the RCNN. Recently, Single Pixel Reconstruction Network (SPRNet) (Yao et al., 2019) improves the one-stage detector by introducing the instance segmentation into the network architecture, which allows one-stage detector to perform multi-task vision sensing similar to the mask-RCNN.

Recently, deep-learning based algorithms are being studied and applied in many agricultural applications (Kamilaris and Prenafeta-Boldú, 2018). Sa et al. (2016),Bargoti and Underwood (2017) applied faster-RCNN in multiple-classes fruit detection, and accurate detection performance was reported from both of work. Liu et al. (2019) applied faster-RCNN on detection of kiwifruit by using RGB and NIR images, an average-precision of 0.904 was reported from their work. Yu et al. (2019) applied mask-RCNN in the application of strawberry harvesting in a non-structured environment. Tian et al. (2019) applied YOLO-v3 in the monitoring of apple growth during different stages, an F_1 score of 0.817 was achieved in their work. Kang and Chen (2019) combined the semantic segmentation and detection into a one-stage detector, to perform the fruit detection and branch segmentation in the apple orchard for robotic harvesting. Other deep-learning based algorithms such as Fully Convolution Network (FCN) (Long et al., 2015) are also being studied and applied in performing vision sensing in the agriculture applications (Lin et al., 2019b; Xu et al., 1873).

3. Methodologies and materials

3.1. Network architecture

DaSNet-v2 follows the network architecture design of the one-stage detection network (such as YOLO), as shown in Fig. 1. It applies a 5-

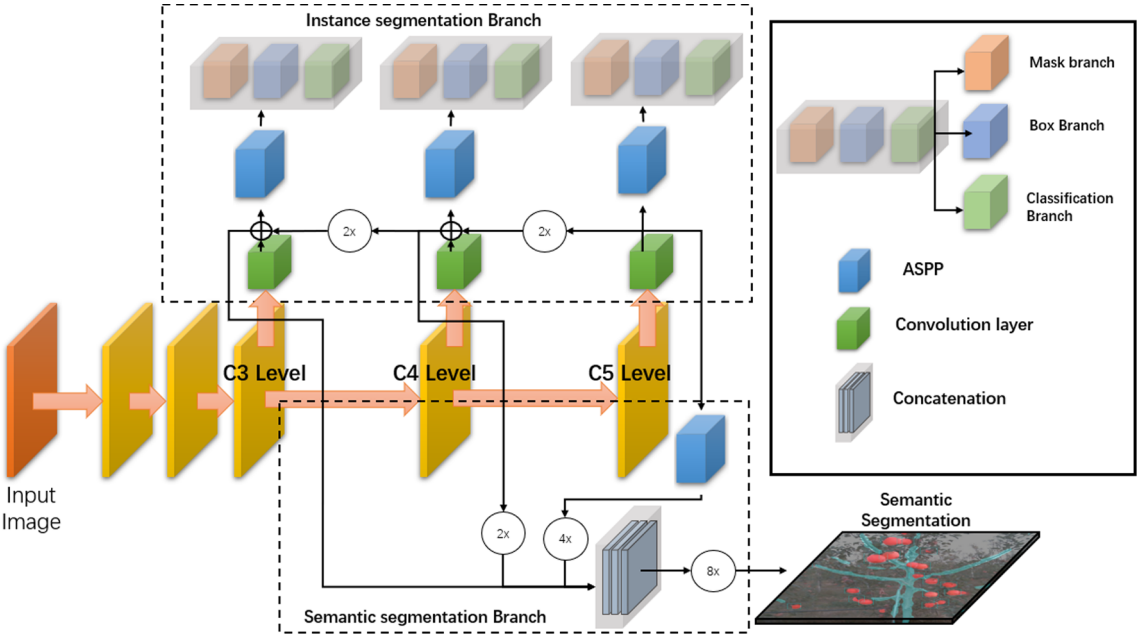

Fig. 1. DaSNet-v2 includes an instance segmentation branch and a semantic segmentation branch for detection and segmentation on apples and branches.

levels network for images classification as the backbone, which generates feature maps of 1/8, 1/16, and 1/32 size of the input image from the C3, C4 and C5 levels, respectively. A 3-level Feature Pyramid Network (FPN) is applied in the DaSNet-v2 to receive and fuse the feature maps from the C3, C4 and C5 level of the backbone to generate the detection and segmentation of fruits and branches. Feature maps from different levels of FPN are in different resolutions and contain the information or features of the objects in different scales. Therefore, the feature maps from different levels of FPN are used to detect the objects in different scales. For example, feature maps in lower-level of the FPN (C3 level) are used to detect the objects in small-scale while feature maps in higher-level (C5 level) are used to detect the objects in large-scale. Meanwhile, feature maps from the higher-level of the network contain more semantic information of objects, which can improve the accuracy of object classification in the detection. Therefore, the feature maps from the C5 level and C4 level of are two times upsampled and be added to the feature maps of the C4 level and C3 level by the FPN, respectively.

On each level of FPN, an instance segmentation branch which predicts the bounding boxes and masks for objects is applied. Besides, a semantic segmentation branch which is used to segment branches from images is grafted on the FPN. Semantic segmentation branch receives the feature maps of the C3, C4, and C5 levels of the FPN (as shown in Fig. 1). By combining the outputs of the instance segmentation branch and the semantic segmentation branch, the processing results of input images are generated.

3.1.1. Instance segmentation branch

Instance segmentation branch is applied to predict, classify and segment the objects from output feature maps of each level of FPN. The instance segmentation branch of DaSNet-v2 follows the design developed in SPRNet. The instance segmentation branch includes the boxes branch, classes branch, and masks branch to predict bounding boxes, classes, and masks of objects, respectively. SPRNet applies a shared decoder to reconstruct the instance masks for objects from individual positive pixels within the feature maps. To encode the multi-scale rich features of an object into a single pixel within the feature maps, the Atrous Spatial Pyramid Pooling (ASPP) is applied before the mask branch in the SPRNet. Our implemented instance segmentation branch in DaSNet-v2 (as shown in Fig. 2) is different from the SPRNet. DaSNet-v2 applies ASPP before the instance segmentation branch (including boxes branch, classes branch, and masks branch) as our implementation suggest that such setup can improve the localisation accuracy of the

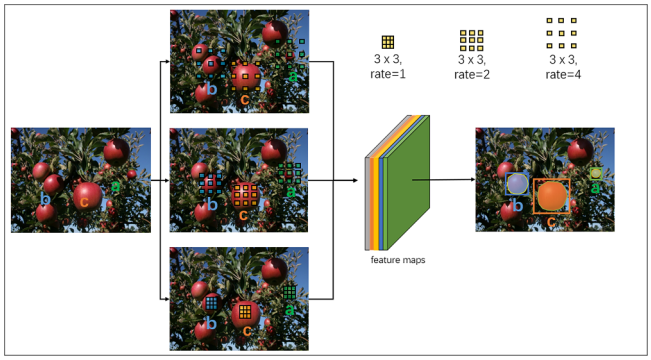

Fig. 3. ASPP applies atrous convolution kernels with given dilate rates to encode extract features of objects at different scales.

bounding boxes. The mask branch of DaSNet-v2 is simplified compared to the SPRNet in terms of improving computational efficiency.

The applied ASPP (as shown in Fig. 3) uses three dilation convolution kernels (3×3) with dilation rate 1, 2 and 4 and a 1×1 kernel to encode the multi-scale features into a single pixel (The implementation suggests that ASPP with large dilation rate may introduce redundant information which can lead to low recall on detection of overlapped objects). The reconstructed masks from the mask branch will be rescaled to the size of the predicted object box. Each level of FPN in the DaSNet-v1 has two preset anchor boxes (3-levels in total). The experimental results in Table 2 show that such setup can efficiently cover the changing shape of bounding boxes in apple detection.

3.1.2. Semantic segmentation branch

Instance segmentation branch can detect and segment the fruits from the input images to stand the location, size and shape of the fruits in working space. However, such information is limited to guide a robot to perform successful harvesting in the orchards setting. There are many obstacles which are presented in the working space of orchards, such as densely arranged branches. To provide more information for robots to understand the current working space, a semantic segmentation branch is applied to perform the branches segmentation from the input images.

The DaSNet-v2 applies 'Encoder-Decoder with atrous convolution' which were developed in Deeplab-v3+ (Chen et al., 2018) to perform the branches segmentation. The semantic segmentation branch of the DaSNet-v2 receives the feature maps of the C3, C4, and C5 levels of the

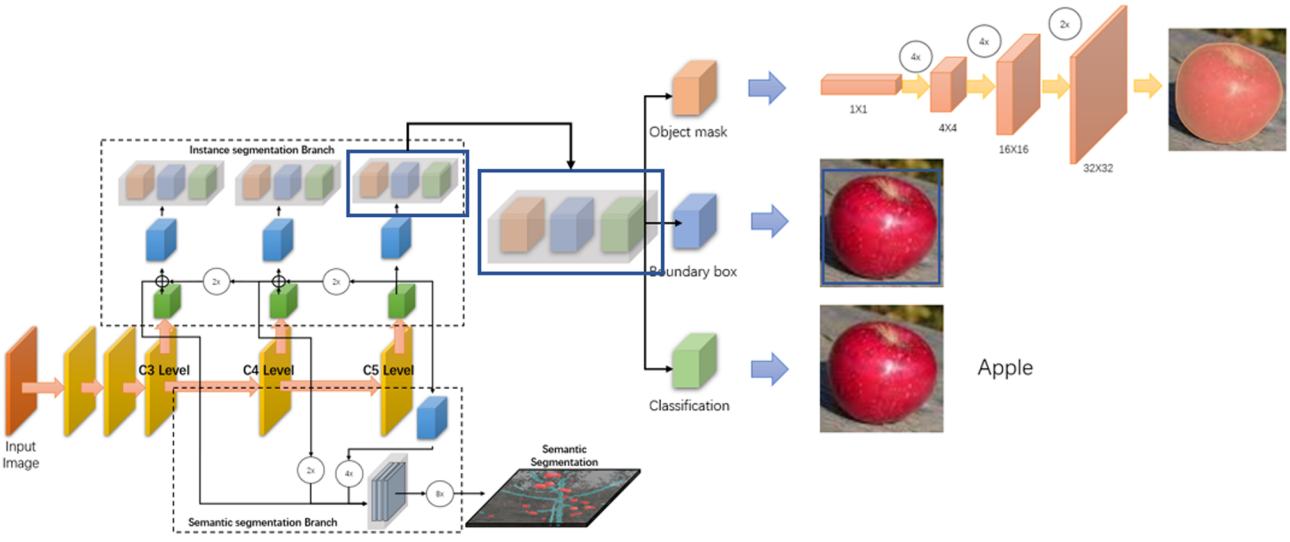

Fig. 2. Network architecture design of the instance segmentation branch applied in the DaSNet-v2, which includes a box branch, a classification branch, and a mask branch.

FPN. The feature maps of C5 level are processed by the ASPP to extract features in different scales. To introduce detail features of objects, the feature maps of C3 and C4 level are concatenated with the processed feature maps from the C5 level. The output tensor of the semantic segmentation branch is 8 times upsampled to match the size of the input images. Different from the DaSNet-v1, the semantic segmentation branch of the DaSNet-v2 only perform segmentation on branches (including branches and trunks), since segmentation of fruits has been included in the instance segmentation branch. .

3.1.3. Compared to the DaSNet-v1

Compared to the DaSNet-v1, the performance of DaSNet-v2 is improved in the following points. Firstly, DaSNet-v2 improves the network model by introducing the instance segmentation into the detection branch. This improvement allows the vision system to provide geometric information (such as shape) of each object. Secondly, DaSNet-v2 optimises the architecture design of the FPN and semantic segmentation branch, compared to the DaSNet-V1. On the one hand, DaSNet-v2 adopts a simplified FPN design, which improves training efficiency and performance of the model. On the other hand, DaSNet-v2 adopts the 'Encoder-Decoder with atrous convolution' from the Deeplab-v3+ to improve the accuracy of branches segmentation.

3.2. Visualisation of working space

In the fruit orchards which is not optimised for robotic operation, the branch and fruits are presented randomly, which can heavily affect the performance of harvesting robots. Densely arranged branches can obstruct the path of robotic arms or even damage the robotic arm (Megalingam et al., 2017). Besides, densely arranged fruits and different types of the stem-branch joint of fruits may also affect the success rate of fruit harvesting (Lin et al., 2019b). To provide a more intuitive understanding of the working environments and guide the manipulator and gripper, 3D modelling and visualisation of the working space in orchards are important (Comba et al., 2018).

DaSNet-v2 can detect and segment the fruit and branches in the orchard environments. For the fruit class, different colours are assigned to the detected fruits to stand their shape and corresponding area. For the branches, a unified colour is assigned to the segmented mask. Other elements within the working space such as ground, fence and leaves are presented in black pixel. Leaf segmentation is not included in the task of DaSNet-v2 since our previous experiments suggest that leaves only block the sight of vision system without obstructing path for picking during the harvesting. PPTK tool-kit (Heremaps, 2018) is used to visualise the 3D point clouds of the working space, an example of 3D visualisation of an orchard environment is shown in Fig. 4.

3.3. Implementation details

3.3.1. Data augmentation

Data augmentation plays an important role in the training of the deep-learning model. To avoid network over-fitting to the training data, extensive image augmentations are introduced. The applied data augmentation includes random crop, random scale (from 50% to 150%), random flip (horizontal only), and random rotation (± 10°). Further, the randomly adjust of brightness (0.5–1.5) and saturation (0.5–1.5) of images in HSV colour space are also applied in the augmentation.

3.3.2. Training method

Focal Loss (FL) (Lin et al., 2017) is used in the training to balance the uneven distribution of the foreground class objects (obj) and background class objects (noobj). The focal loss can be expressed as:

$$FL(p) = \sum^{obj} -\alpha(1-p)^{\gamma}log(p) + \sum^{noobj} -\beta(p)^{\gamma}log(1-p) \quad (1)$$

p is the confidence score of the object. α, β, and γ are the inner parameters to adjust the profile of the loss function. We set α, β, and γ as 8, 0.5, and 2 in the training, respectively. Other training tasks including regression of bounding boxes and classification follow the design of the YOLO-V3 (Redmon and Farhadi, 2018). Cross-entropy loss is used in the training of the instance segmentation and semantic segmentation tasks. Adam-optimizer is used in the training of the network model. The learning rate, decay rate, and batch size used in the training are 0.01, 0.9 and 32, respectively. The backbone weights of the network are fixed in the first 100 epochs of the training. Then the overall network is trained for another 50 epochs.

3.3.3. Other details

The programming of DaSNet-v2 was performed by using TensorFlow-slim image classification model library (Silberman and Guadarrama, 2016) in Ubuntu 16.04. 3D visualisation of the point cloud is achieved by using PPTK tool-kit. The DaSNet-v2 is trained on the GTX-1080Ti (Nvidia, United States) and be tested on Jetson-TX2 (Nvidia, United States) and GTX-1080Ti. Intel RealSense D-435 RGB-D camera (Intel, United States) is used to perform vision sensing in the field-test. It is controlled by using the realsense-ros SDK (Intel-Corp, 2018) in ROS-kinetic on Ubuntu 16.04.

To ensure the computation availability of the DaSNet-v2 model on the Jetson-TX2, a light-weight modified resnet-18 (Kang and Chen, 2019) (as shown in Fig. 5) was used as the backbone in the model of DaSNet-v2. Resnet-50 and resnet-101 (He et al., 2016) were also used as the backbone in the model of DaSNet-v2. The implemented code and ImageNet pre-trained weights of the resnet-50 and resnet-101 were from the Github publicly code library (Taylor et al., 2018), the resnet-18 was pre-trained on Cifar (Krizhevsky et al., 2009).

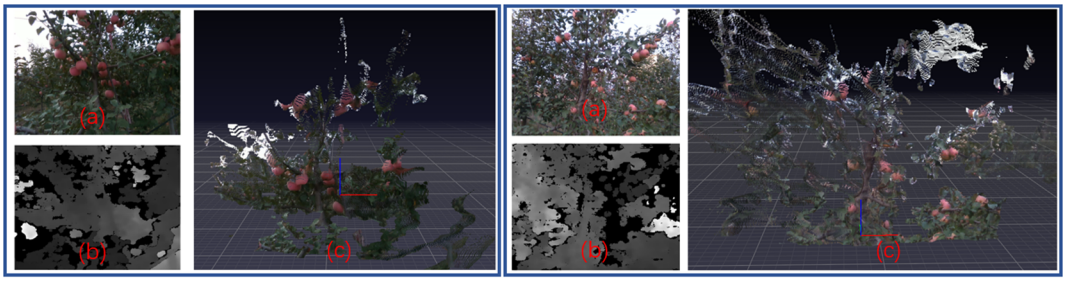

Fig. 4. (a), (b), and (c) of figure (i) and (ii) are the RGB images, depth images, and point clouds in 3D space, respectively.

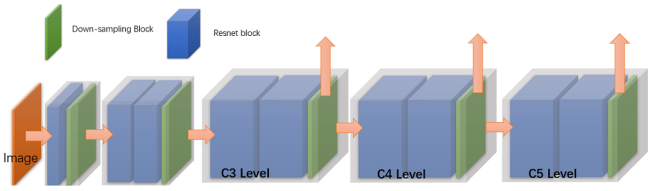

Fig. 5. Architecture of the resnet-18, it applies bottleneck designed resnet block to reduce the weight size and improve the computational efficiency.

Table 1
Numbers of image data from different dataset in training, validation, and test set.

	Device	Type	Number	Training	Validation	Test
A	C-615	RGB	427	227	50	150
B	D-435	RGB	382	192	50	140
C	D-435	RGB-D	468	148	50	270
Total	–	–	1277	567	150	560

4. Experiment and discussion

4.1. Data collection

Both RGB-D images and RGB images were collected from the apple orchard located at Qingdao, China. The collection time of the image data was from 10:00 am to 21:00 pm through the day by using the Intel RealSense D-435 RGB-D camera and Logitech webcam-C615 (Logitech, Switzerland). The images were collected at the distance of 0.5–1.5 m from the camera to apple trees, which is the distance from the camera to trees during the robotic harvesting. There were about 400 RGB-D images and more than 800 RGB images which were collected during the field-test in the apple orchard (as shown in Table 1). Image data A, B and C were collected by using handheld webcam C-615, depth-camera D-435 on the robotic arm (see Fig. 6), and handheld depth-camera D-435, respectively. We used 148 RGB-D images (only RGB information was used in training) and 419 RGB images as the training set and applied another 150 images as the validation set. The rest of the images were used to evaluate the performance of the trained model.

Table 2
Comparison of performance on detection and instance segmentation among different networks models on GTX-1080Ti, image size (640 × 480).

Model	F_1	Recall	Precision	IoU_{box}	IoU_{mask}	Time
Faster-RCNN	0.852	0.836	0.872	0.858	–	127 ms
YOLO-v3	0.86	0.852	0.87	0.851	–	43 ms
DaSNet-v1	0.863	0.857	0.875	0.856	–	54 ms
Mask-RCNN	0.868	0.86	0.882	0.863	0.878	158 ms
DaSNet-v2	0.873	0.868	0.88	0.861	0.873	70 ms

Table 3
Comparison of performance on detection by DaSNet-v2 with different backbones on GTX-1080Ti, image size (640 × 480).

Backbone	F_1	Recall	Precision	IoU_{box}	IoU_{mask}	Time
Resnet-18	0.857	0.85	0.87	0.858	0.866	54 ms
Resnet-50	0.868	0.861	0.876	0.859	0.872	64 ms
Resnet-101	0.873	0.868	0.88	0.861	0.873	70 ms

4.2. Evaluation method

The performance evaluation includes three tasks, which are the accuracy evaluation on fruit detection and segmentation, and branch segmentation. To evaluate accuracy of fruit detection, Intersection of Union (IoU) and F_1 score are used as performance metric in this work. IoU computes a ratio of the intersection and the union of two sets (Garcia-Garcia et al., 2017). In the case of detection and segmentation, the sets can be bounding boxes or masks of prediction and ground-truth. IoU measures the localisation accuracy of bounding box or accuracy of segmentation. When assessing the performance of detection, we compute the IoU by using bounding boxes (denoted in IoU_{box}) between prediction and ground truth. The predicted objects with IoU_{box} and confidence score higher than 0.5 will be treated as true-positive. F_1 score measures the detection performance by using the *Recall* and *Precision*. *Recall* measures the fraction of true-positive objects that are successfully detected, while *Precision* measures the fraction of true-positive objects in the predictions. The expression of the *Precision*, *Recall* and F_1 score are listed as follow:

$$Precision = \frac{TruePositive(TP)}{TruePositive(TP) + FalsePositive(FP)} \quad (2)$$

Fig. 6. (a) Setting of the apple orchard, (b) apple harvesting robot and mobile platform, (c) setting of depth camera.

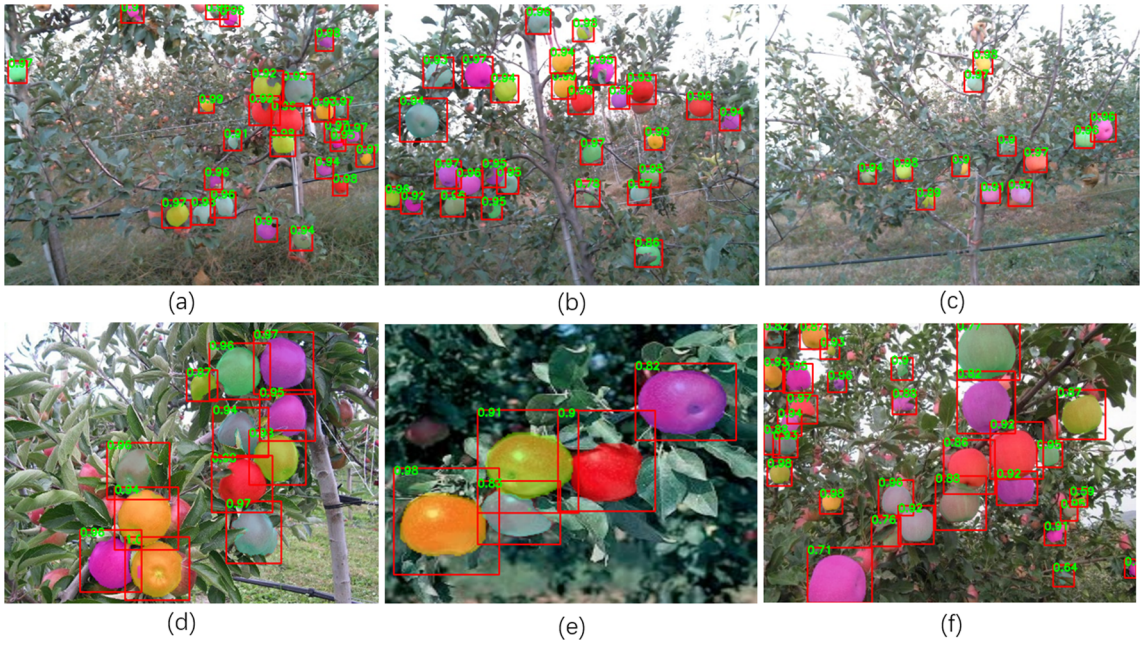

Fig. 7. Instance segmentation of fruits by using DaSNet-v2 with resnet-101. Each fruit is drawn in a distinguished colour, green numbers are the confidence values of detected objects within the boxes. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

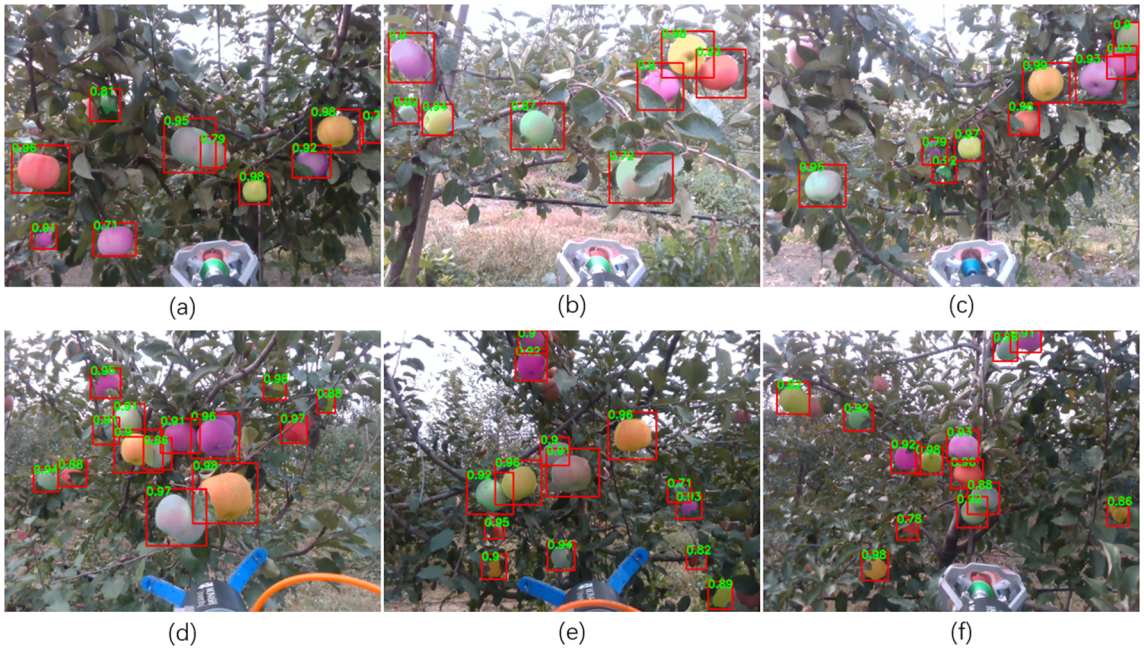

Fig. 8. Instance segmentation of fruit by using DaSNet-v2 with resnet-101, the images are collected by depth-camera under the view of robotic arm during harvesting. Each fruit is drawn in a distinguished colour.

Table 4
Comparison of performance on semantic segmentation of branches among different models, image size (640 × 480).

Model (Backbone)	IoU_{branch}	Time
FCN-8s (Resnet-101)	0.757	52 ms
DaSNet-v1 (Resnet-101)	0.772	54 ms
DaSNet-v2 (Resnet-101)	0.794	70 ms

Table 5
Comparison of performance of DaSNet-v2 on semantic segmentation with different backbones, image size (640 × 480).

Backbone	IoU_{branch}	Time
Resnet-18	0.775	54 ms
Resnet-50	0.788	64 ms
Resnet-101	0.794	70 ms

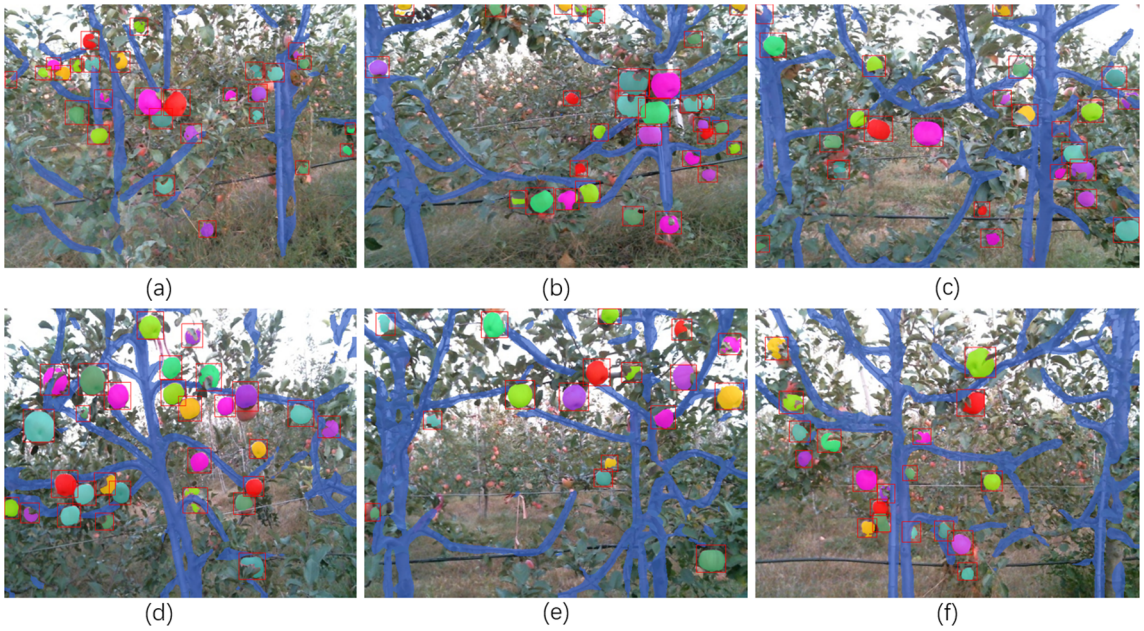

Fig. 9. Detection and segmentation of fruits and branches by using the DaSNet-v2 in the orchard. Fruits are drawn in distinguished colours, branches are drawn in the colour of blue. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

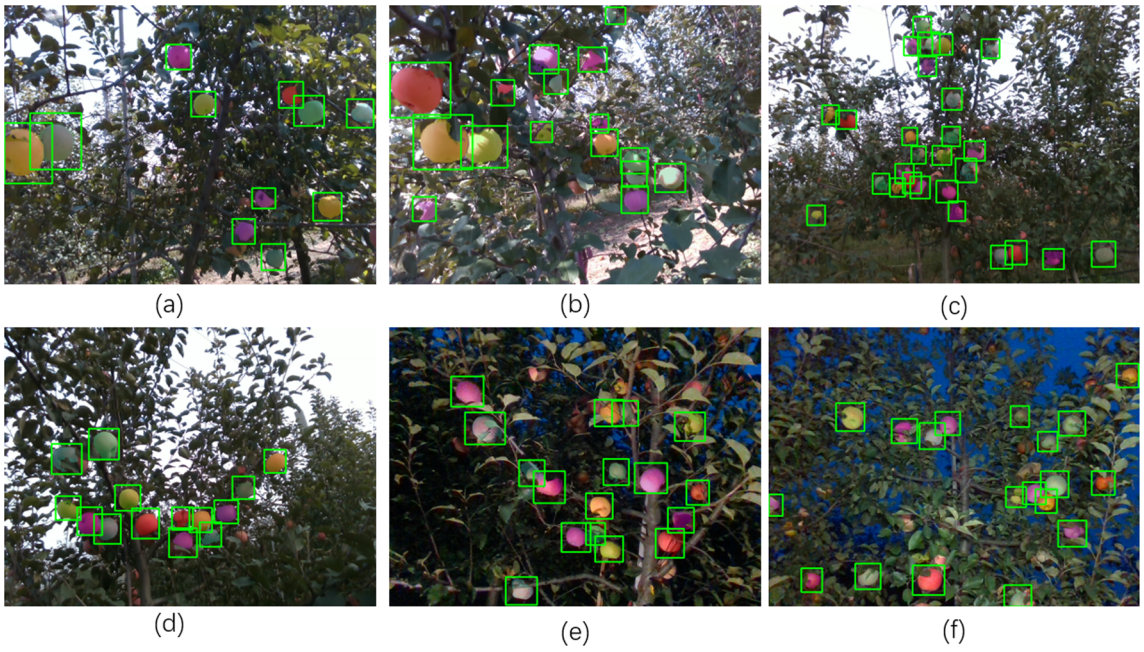

Fig. 10. Detection and segmentation of fruits by using the DaSNet-v2 in different times. (a) and (b) are images taken in 11:00 am-13:00 pm, (c) and (d) are images taken between 4:00 pm to 6:00 pm, (e) and (f) are images taken between 7:00 pm to 9:00 pm under artificial lighting.

$$Recall = \frac{TP}{TP + FalseNegative\,(FN)} \tag{3}$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4}$$

In the performance evaluation, the F_1 score is calculated by averaging the F_1 score of each image within the test set. The IoU is also used to evaluate the accuracy of instance segmentation and semantic segmentation of network models on fruits and branches, which are denoted as IoU_{mask} and IoU_{branch}, respectively.

4.3. comparison to state of the art

4.3.1. Evaluation of detection and instance segmentation

A series of experiments were conducted to compare the detection performance among the DaSNet-v2 and DaSNet-v1, YOLO-v3, faster-RCNN and the mask-RCNN. YOLO-v3 is the representative work of the one-stage detector, which applies darknet-53 as the backbone and a 3-level FPN in the model. The implemented code of YOLO-v3 is from Github publicly code library (Kapica, 2019). Faster-RCNN and mask-RCNN are the representative works of the two-stage detector. The implemented code of faster-RCNN (Jia et al., 2014) use VGG-19 (Simonyan and Zisserman, 2014) as the backbone, while FPN is not applied in the model. The implemented code of mask-RCNN (Abdulla,

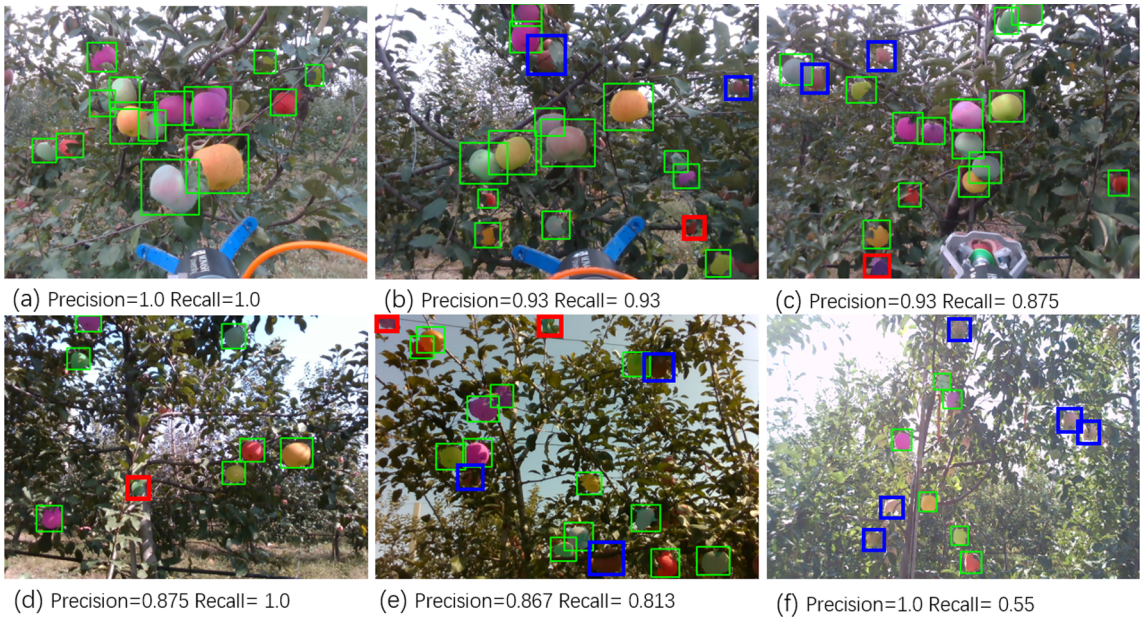

Fig. 11. Success and Failure detection by using DaSNet-v2 in the orchard. Green, red, and blue boxes represent the True-Positive (TP), False-Positive (FP), and False-Negative (FN) of detection, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 6
Comparison of computational efficiency of network models on Jetson-TX2, image size (640 × 480).

Model	Weight Size	Time
YOLO-v3 (darknet-53)	246 M	235 ms
DaSNet-v1 (resnet-101)	192 M	306 ms
DaSNet-v2 (resnet-18)	8.1 M	342 ms
DaSNet-v2 (resnet-101)	187 M	437 ms
Faster-RCNN (VGG-19)	533 M	1.1 s
Mask-RCNN (resnet-101)	244 M	1.3 s

2017) (FPN applied), DaSNet-v1, and DaSNet-v2 applies resnet-101 as backbone. In the experiment, all the network models were trained and tested on our collected training set and test set. We set 0.5 as the threshold for confidence and IoU_{box} in all network models. The experimental results among different network models are shown in Table 2. The comparison results among DaSNet-v2 with different backbones are shown in Table 3.

As shown in Table 2, DaSNet-v2 and mask-RCNN outperform other network models in terms of the fruit detection. The F_1 score of DaSNet-v2 and mask-RCNN are 0.873 and 0.868, respectively. The implemented code of faster-RCNN does not adopt FPN in the network model. Experimental results show that it has lower recall on small-scale

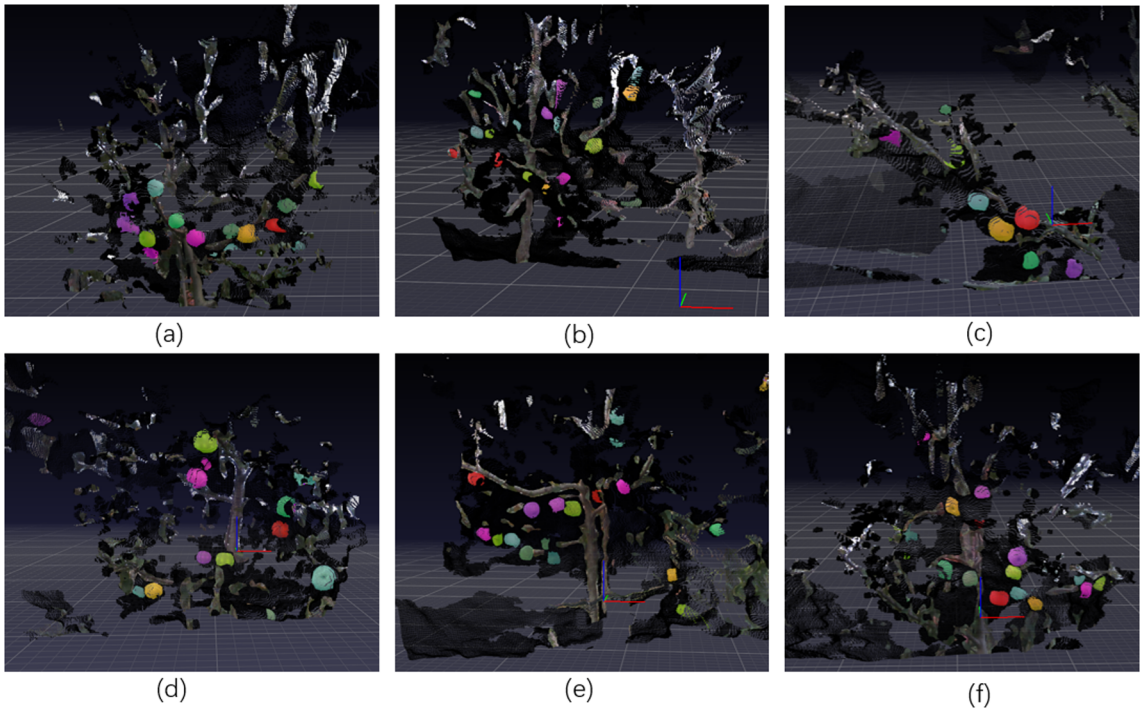

Fig. 12. 3D visualisation of the processed orchard by using PPTK. The fruits are drawn in distinguished colours, branches are drawn in original colour.

objects compared to the other network models. Therefore, a lower score on recall and F_1 score are reported on fruit detection by using faster-RCNN, which are 0.836 and 0852, respectively. As shown in (a-c) of Fig. 7, many fruits in images are presented in small-scale, especially in the images which are collected from orchard environments. Compared to the DaSNet-v1, DaSNet-v2 optimises the network architecture and training procedures. Therefore, a higher score on both recall and precision are obtained by DaSNet-v2, which are 0.868 and 0.88, respectively. Compared to the faster-RCNN, mask-RCNN with FPN design achieves a higher score on both recall and precision of detection, which are 0.86 and 0.882, respectively. In terms of the instance segmentation, mask-RCNN and DaSNet-v2 achieve similar score on the accuracy of instance segmentation, which are 0.878 and 0.873, respectively. Figs. 7 and 8 show the examples of detection and instance segmentation of apples by using DaSNet-v2.

From the experimental results shown in Table 2, one-stage detectors have better computational efficiency compared to two-stage detectors. The average computational time of faster-RCNN and mask-RCNN are 127 ms and 158 ms, respectively. While the average computational time of YOLO-v3, DaSNet-v1, and DaSNet-v2 are 43 ms, 54 ms, and 70 ms, respectively. DaSNet-v2 achieves similar performance on fruit detection compared to the mask-RCNN with better computational efficiency. Table 3 shows the performance of DaSNet-v2 with different backbones. To apply the DaSNet-v2 in the embedded computational device such as Jetson-TX2, a light-weight backbone Resnet-18 is adopted in the DaSNet-v2. Experimental results show that DaSNet-v2 with Resnet-18 can achieve similar performance on recall and precision of detection compared to the YOLO-v3. The recall and precision of DaSNet-v2 with Resnet-18 are 0.85 and 0.87, respectively. The weight size and average computational time of DaSNet-v2 with Resnet-18 are 8.1 MB and 54 ms (as shown in Table 6), respectively.

4.3.2. Evaluation of semantic segmentation

This experiment compares the performance of semantic segmentation between the DaSNet-v2, DaSNet-v1 and the FCN-8s. The implemented code of FCN-8s with resnet-101 is from Github publicly code library (Pakhomov et al., 2017). The experimental results are shown in Table 4.

From the experimental results shown in Table 4, the accuracy of semantic segmentation on branches achieved by DaSNet-v2 is improved compared to the DaSNet-v1 and FCN-8s. The IoU value on branches segmentation achieved by FCN-8s, DaSNet-v1 and DaSNet-v2 are 0.757, 0.772 and 0.794, respectively. Compared to the DaSNet-v1, DaSNet-v2 only applies ASPP on the feature maps from C5 level, as experimental results suggest ASPP on lower-level (such as C3 and C4) will introduce noise and lead to under-fitting of the model. Compared to the FCN-8s, DaSNet-v2 achieves 3.7% higher value on IoU_{branch}. Table 5 compares the accuracy on branches segmentation by DaSNet-v2 with different backbones. Experimental results show that backbones with better performance can improve the accuracy of branches segmentation. The IoU_{branch} achieved by DaSNet-v2 with Resnet-18, Resnet-50, and Resnet-101 are 0.775, 0.788 and 0.794, respectively.

The average computational time of DaSNet-v2 is increased compared to the DaSNet-v1, which is due to the increasing computational consumption on instance segmentation branch. Although DaSNet-v2 has shown an improved ability on branches segmentation, to classify tree under various conditions accurately is still a challenging task. The segmentation of branches and fruits by using DaSNet-v2 are shown in Fig. 9.

4.4. Visual sensing in orchards

There are various factors which are presented in orchards environments, such as illumination variation, overlapped fruits or branches, and appearance variation. These factors can heavily affect the accuracy of detection and segmentation. The DaSNet-v2 was tested in the apple orchard in different setup (including operation time and mode), the results which are processed by DaSNet-v2 are visualised in Figs. 8 and 10. Several examples of success or fail detections by DaSNet-v2 in orchard environments are shown in Fig. 11. The detection errors include two types: false-positive and false-negative, which are linked to the precision and recall of detection, respectively. From experimental results shown in Fig. 11, false-positive in detection mainly caused by false detection on leaves or branches. The reasons that lead to false-negative in detection are varied. Environment factors such as strong sunlight reflection, shadow, and appearances variation of fruits in colours, shape, occlusion, or view-angle can lead to the false-negative in detection. These factors can cause inaccurate in the detection, while the experimental results in Table 2 shows that DaSNet-v2 achieves high recall and precision on detection of apples in orchard environments.

The developed fruit harvesting robot applies Jetson-TX2 as computation centre to process vision sensing and robot control. The comparison of weight size and average computation time of different network models on Jetson-TX2 are shown in Table 6. It can be seen that the one-stage detectors,such as YOLO-v3 and DaSNet-v2, have better computational efficiency compared to the two-stage detector.

4.5. 3D visualisation of orchards

The collected RGB-D images from the orchard are processed by using the DaSNet-v2 and visualised by using the PPTK, which are shown in Fig. 12. As shown in figures, 3D point clouds with semantic information added can clearly describe the working environments of the harvesting robot in orchard environments. These information can be used to construct the 3D map of working spaces (Lang et al., 2013) and compute the pose of each fruit (Wong et al., 2017), which can increase the success rate of robotic harvesting (Bac et al., 2014). Such works will be included in the future works of development of intelligent robotic system for fruit harvesting.

5. Conclusion and future works

In this study, a multi-function deep neural network DaSNet-v2 was proposed and validated. DaSNet-v2 combines an instance segmentation branch and a semantic segmentation branch into the architecture of the one-stage detection network, which allows DaSNet-v2 to perform detection and segmentation on each fruit, and semantic segmentation on branches. Besides, DaSNet-v2 adopts FPN and ASPP to improve the performance on detection and segmentation of fruits and branches. To improve the computational efficiency of network model running on embedded computational devices, DaSNet-v2 with a light-weight backbone resnet-18 was trained and validated in this work. In the experiments, DaSNet-v2 was tested and validated by experimental results obtained from field tests in an apple orchard. DaSNet-v2 with resnet-101 achieved 0.868 and 0.88 on recall and precision of detection, 0.873 on the accuracy of fruits segmentation, and 0.794 on the accuracy of branches segmentation, respectively. DaSNet-v2 with resnet-18 achieved 0.85 and 0.87 on recall and precision of detection, 0.866 on the accuracy of fruits segmentation, and 0.757 on the accuracy of branches segmentation, respectively. The weight size and average computational time of DaSNet-v2 with resnet-18 to process an image (640 × 480) on GTX-1080Ti are 8.1 M and 54 ms, respectively. From the experiment results, DaSNet-v2 demonstrated a robust and efficient performance on vision sensing in orchards. Future work will focus on developing the orchard reconstruction algorithm based on the DaSNet-v2, corresponding control strategy for guiding the automatic robotic fruit harvesting will also be included.

CRediT authorship contribution statement

Hanwen Kang: Conceptualization, Methodology, Software, Data curation, Visualization, Validation, Writing - original draft. **Chao Chen:**

Conceptualization, Writing - review & editing, Validation, Supervision, Project administration.

Declaration of Competing Interest

The authors declare no conflict of interest.

Acknowledgement

This work is supported by ARC ITRH IH150100006 and THOR TECH PTY Ltd. We acknowledge Zijue Chen and Hongyu Zhou for their assistance in the data collection. And we also acknowledge Zhuo Chen for her assistance in preparation of this work.

References

ABARES, 2018. Australian vegetable growing farms: an economic survey, 2016-17 and 2017-18. Australian Bureau of Agricultural and Resource Economics (ABARE): Canberra.

Abdulla Waleed, 2017. Mask r-cnn for object detection and instance segmentation on keras and tensorflow, 2017. https://github.com/matterport/Mask_RCNN. [Online; accessed Sep-2019].

Bac, C. Wouter, van Henten, Eldert J., Hemming, Jochen, Edan, Yael, 2014. Harvesting robots for high-value crops: State-of-the-art review and challenges ahead. J. Field Robot. 31 (6), 888–911.

Bargoti, Suchet, Underwood, James, 2017. Deep fruit detection in orchards. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 3626–3633.

Chen Liang-Chieh, Zhu Yukun, Papandreou George, Schroff Florian, Adam Hartwig, 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 801–818.

Comba, Lorenzo, Biglia, Alessandro, Aimonino, Davide Ricauda, Gay, Paolo, 2018. Unsupervised detection of vineyards by 3d point-cloud uav photogrammetry for precision agriculture. Comput. Electron. Agric. 155, 84–95.

Garcia-Garcia Alberto, Orts-Escolano Sergio, Oprea Sergiu, Villena-Martinez Victor, Garcia-Rodriguez Jose, 2017. A review on deep learning techniques applied to semantic segmentation. arXiv preprint arXiv:1704.06857.

Girshick Ross, 2015. Fast r-cnn. In: Proceedings of the IEEE international Conference on Computer Vision, pp. 1440–1448.

Han, Junwei, Zhang, Dingwen, Cheng, Gong, Liu, Nian, Dong, Xu., 2018. Advanced deep-learning techniques for salient and category-specific object detection: a survey. IEEE Signal Process. Mag. 35 (1), 84–100.

He Kaiming, Zhang Xiangyu, Ren Shaoqing, Sun Jian, 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.

He Kaiming, Gkioxari Georgia, Dollár Piotr, Girshick Ross, 2017. Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2961–2969.

Heremaps, 2018. heremaps/pptk, URL https://github.com/heremaps/pptk. [Online; accessed July-2019].

Intel-Corp, 2018. Intel realsense sdk 2.0, https://github.com/IntelRealSense/realsense-ros.

Jia Yangqing, Shelhamer Evan, Donahue Jeff, Karayev Sergey, Long Jonathan, Girshick Ross, Guadarrama Sergio, Darrell Trevor, 2014. Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093.

Kamilaris, Andreas, Prenafeta-Boldú, Francesc X., 2018. Deep learning in agriculture: A survey. Comput. Electron. Agric. 147, 70–90.

Kang, Hanwen, Chen, Chao, 2019. Fruit detection and segmentation for apple harvesting using visual sensor in orchards. Sensors 19 (20), 4599.

Kapach, Keren, Barnea, Ehud, Mairon, Rotem, Edan, Yael, Ben-Shahar, Oh.ad., 2012. Computer vision for fruit harvesting robots–state of the art and challenges ahead. Int. J. Comput. Vision Robot. 3 (1/2), 4–34.

Kapica Pawel, 2019. tensorflow-yolov3, https://github.com/mystic123/tensorflow-yolo-v3. [Online; accessed july-2019].

Krizhevsky Alex, Hinton Geoffrey, et al., 2009. Learning multiple layers of features from tiny images.

Lang, Dagmar, Friedmann, Susanne, Paulus, Dietrich, 2013. Semantic 3d octree maps based on conditional random fields. MVA 13, 185–188.

Lin, Guichao, Tang, Yunchao, Zou, Xiangjun, Xiong, Juntao, Fang, Yamei, 2019a. Color-, depth-, and shape-based 3d fruit detection. Precision Agric. 1–17.

Lin, Guichao, Tang, Yunchao, Zou, Xiangjun, Xiong, Juntao, Li, Jinhui, 2019b. Guava detection and pose estimation using a low-cost rgb-d sensor in the field. Sensors 19 (2), 428.

Lin, Tsung-Yi, Goyal, Priya, Girshick, Ross, He, Kaiming, Dollár, Piotr, 2017. Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2980–2988.

Liu, Wei, Anguelov, Dragomir, Erhan, Dumitru, Szegedy, Christian, Reed, Scott, Cheng-Yang, Fu., Berg, Alexander C, 2016. Ssd: Single shot multibox detector. In: European Conference on Computer Vision. Springer, pp. 21–37.

Liu, Xiaoyang, Jia, Weikuan, Ruan, Chengzhi, Zhao, Dean, Yuwan, Gu., Chen, Wei, 2018. The recognition of apple fruits in plastic bags based on block classification. Precis. Agric. 19 (4), 735–749.

Liu Zhihao, Wu Jingzhu, Fu Longsheng, Majeed Yaqoob, Feng Yali, Li Rui, Cui Yongjie, 2019. Improved kiwifruit detection using pre-trained vgg16 with rgb and nir information fusion. IEEE Access.

Long, Jonathan, Shelhamer, Evan, Darrell, Trevor, 2015. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440.

McCool Christopher, Sa Inkyu, Dayoub Feras, Lehnert Christopher, Perez Tristan, Upcroft Ben, 2016. Visual detection of occluded crop: For automated harvesting. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 2506–2512.

Megalingam Rajesh Kannan, Vivek Gedela Vamsy, Bandyopadhyay Shiva, Rahi Mohammed Juned, 2017. Robotic arm design, development and control for agriculture applications. In: 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS). IEEE, pp. 1–7.

Nguyen, Tien Thanh, Vandevoorde, Koenraad, Wouters, Niels, Kayacan, Erdal, De Baerdemaeker, Josse G., Saeys, Wouter, 2016. Detection of red and bicoloured apples on tree with an rgb-d camera. Biosyst. Eng. 146, 33–44.

Pakhomov Daniil, Premachandran Vittal, Allan Max, Azizian Mahdi, Navab Nassir, 2017. Deep residual learning for instrument segmentation in robotic surgery. arXiv preprint arXiv:1703.08580.

Redmon Joseph, Farhadi Ali, 2018. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

Ren Shaoqing, He Kaiming, Girshick Ross, Sun Jian, 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99.

Sa, Inkyu, Ge, Zongyuan, Dayoub, Feras, Upcroft, Ben, Perez, Tristan, McCool, Chris, 2016. Deepfruits: A fruit detection system using deep neural networks. Sensors 16 (8), 1222.

Silberman N., Guadarrama, S., 2016. Tensorflow-slim image classification model library, URL https://github.com/tensorflow/models/tree/master/research/slim.

Simonyan Karen, Zisserman Andrew, 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Taylor Robie, Hongkun Yu, Wu Neal, 2018. tensorflow-resnet. URL https://github.com/tensorflow/models/tree/master/research/resnet. [Online; accessed Nov-2018].

Tian, Yunong, Yang, Guodong, Wang, Zhe, Wang, Hao, Li, En, Liang, Zize, 2019. Apple detection during different growth stages in orchards using the improved yolo-v3 model. Comput. Electron. Agric. 157, 417–426.

Vasconez, Juan P., Kantor, George A., Auat Cheein, Fernando A., 2019. Human–robot interaction in agriculture: A survey and current challenges. Biosyst. Eng. 179, 35–48.

Vibhute, Anup, Bodhe, S.K., 2012. Applications of image processing in agriculture: a survey. Int. J. Comput. Appl. 52 (2).

Wang, Yi, Lihong, Xu., 2018. Unsupervised segmentation of greenhouse plant images based on modified latent dirichlet allocation. PeerJ 6, e5036.

Wong, Jay M., Kee, Vincent, Le, Tiffany, Wagner, Syler, Mariottini, Gian-Luca, Schneider, Abraham, Hamilton, Lei, Chipalkatty, Rahul, Hebert, Mitchell, Johnson, David M.S., et al., 2017. Segicp: Integrated deep semantic segmentation and pose estimation. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 5784–5789.

Xu, Hui, Chen, Guodong, Wang, Zhenhua, Sun, Lining, Fan, Su., 1873. Rgb-d-based pose estimation of workpieces with semantic segmentation and point cloud registration. Sensors 19 (8), 2019.

Yao Jinghan, Yu Zhou, Yu Jun, Tao Dacheng, 2019. Single pixel reconstruction for one-stage instance segmentation. arXiv preprint arXiv:1904.07426.

Yu, Yang, Zhang, Kailiang, Yang, Li, Zhang, Dongxing, 2019. Fruit detection for strawberry harvesting robot in non-structural environment based on mask-rcnn. Comput. Electron. Agric. 163, 104846.

Zhao, Yuanshen, Gong, Liang, Huang, Yixiang, Liu, Chengliang, 2016. A review of key techniques of vision-based control for harvesting robot. Comput. Electron. Agric. 127, 311–323.

Zhou, Rong, Damerow, Lutz, Sun, Yurui, Blanke, Michael M, 2012. Using colour features of cv.'gala'apple fruits in an orchard in image processing to predict yield. Precision Agric. 13 (5), 568–580.