Precision Agriculture (2020) 21:387–402
https://doi.org/10.1007/s11119-019-09673-7

A deep-level region-based visual representation architecture for detecting strawberry flowers in an outdoor field

P. Lin[1,2] · W. S. Lee[2] · Y. M. Chen[1,2] · N. Peres[3] · C. Fraisse[2]

Published online: 7 June 2019

Abstract

An accurate and robust strawberry flower representation and detection scheme is a key step to enable the reliable forecasting of fruit yield for use in precision agricultural applications. A state-of-the-art deep-level object detection framework which processes images through several layers using a region-based convolutional neural network (R-CNN) was developed to visually represent the instances of strawberry flowers in outdoor fields and improve the detection accuracy. A modified version of the visual geometry group 19 (VGG19) architecture, which had 47 layers, was used to represent the multiple scales of strawberry flower image features. The networks were trained entirely on 400 strawberry flower images and tested on another 100 images. Different region-based object detection methods, including the R-CNN, Fast R-CNN and Faster R-CNN, were used to represent the strawberry flower instances. The Faster R-CNN model achieved a better performance than the R-CNN and Fast R-CNN in detecting the instances and had a lower execution time. The detection accuracy of the Faster R-CNN model was 86.1%, which was higher than those of the R-CNN and Fast R-CNN models (63.4% and 76.7%, respectively). The experimental results showed the effectiveness of the deep-level Faster R-CNN framework for representing the strawberry flower instances under various camera view-points, different distances to flowers, overlaps, complex background illumination, blur, etc. The system developed for automatic and accurate strawberry flower detection provides an important and significant solution that enables subsequent applications to estimate the strawberry yield in outdoor fields.

Keywords Strawberry flower · Object detection · Yield estimation · Precision agriculture

✉ Y. M. Chen
billrange@126.com

[1] College of Electrical Engineering, Yancheng Institute of Technology, No. 1 Middle Road Hope Avenue, Yancheng 224051, Jiangsu Province, People's Republic of China

[2] Department of Agricultural and Biological Engineering, University of Florida, Gainesville, FL 32611, USA

[3] Gulf Coast Research and Education Center, University of Florida, Wimauma, FL 33598, USA

Introduction

In recent years, object detection has become the most popular problem in the field of computer vision. It is an image segmentation method based on the geometric and statistical features of a target (Arel et al. 2010; Girshick et al. 2016; Lin et al. 2012). It combines the segmentation and recognition of a target, and accuracy and real-time processing are important performance measures of the whole system. A number of object detection techniques have been used in the fields of information and industry (Arel et al. 2010; Cireşan et al. 2013), i.e., facial recognition (Wen et al. 2016; Chaudhry and Chandra 2017), unmanned driving (Dairi et al. 2018) and other fields (Zou et al. 2012). In recent years, object detection technology has been extended to agricultural applications. A grape detection system based on a radial symmetry transform was developed for estimating the number of grapes (Nuske et al. 2011). A computer vision-based system based on colour and distinctive specular reflection patterns was explored for automated, rapid and accurate yield estimation in apple orchards (Wang et al. 2013). However, the system was operated at night-time with controlled artificial lighting to reduce the variance of natural illumination. A multispectral system was exploited to segment sweet peppers that used artificial lighting and a series of features, such as original multispectral data, normalized difference index, and the entropy-based texture features. However, it was not accurate enough to establish a reliable obstacle map (Bac et al. 2013). Images of outdoor orchards pose new challenges for fruit detection. As the images are taken in outdoor scenes, there are various kinds of interference factors, such as object orientation, occlusion, light intensity, illumination conditions, fruit distance, fruit clustering, and camera view. These factors are due to the appearance changes of fruits in the field, including the shape, colour, texture, size and reflectance properties, which lead to a distortion of the target and pose new challenges for object detection (Nuske et al. 2014; Yamamoto et al. 2014). A computer vision system was developed to estimate the amount of fallen citrus fruit in varying illumination conditions and determine the decay stage of the fallen fruit in an orchard; the authors noted that the varying outdoor illumination presented a large challenge (Choi et al. 2016). Although many methods have been put forward to solve the problems of agricultural object detection in recent years, the establishment of an accurate and reliable detection system is still a challenging task.

Convolutional neural networks are a category of neural networks that have been shown to be highly effective in areas such as image recognition and classification. The framework of the deep-level region-based convolutional neural network that processes images through several layers was developed to improve the accuracy of object detection models (Underwood et al. 2016; Bargoti and Underwood 2017). The deep-level region convolution network architecture with 47 layers was used to detect strawberry flower images in this study. The convolutional neural networks automatically capture the feature representations from training images in the different scale spaces for object detection, thus avoiding the need to hand-engineer the features by capturing the data distribution discriminately. The detection framework mainly consists of two steps. The first step of the framework employs a region proposal method which generates a set of candidate boxes that pre-locate the possible location of the target in the image, such as the selective search (SS) method (Uijlings et al. 2013) and the edgebox method (Zitnick and Dollár 2014). After extracting the features from the regions of interest (ROIs) of these proposals, the features are then input into a deep neural network for further classification. Although it has a high recall performance, the framework has a large computational load, which makes it unsuitable for a real time robotic application. The region proposal networks (RPNs) (Ren et al. 2015; He et al. 2015) address the problem by combining the object proposal network with a deep convolutional network for classification, which enables the system to predict the scope of the objects and classify them at the same time. Moreover, the parameters of the two networks are shared, resulting in a much faster performance and making it suitable for practical applications.

In this study, a modified version of an object detection framework based on a deep-level region-based convolutional neural network (R-CNN) using the 47-layer VGG19 architecture was presented to represent the features of strawberry flower images at multiple scales. The original VGG-19 model was trained on a subset of the ImageNet database, which can be used to classify images into 1000 object categories (Sendik and Cohen-Or 2017). Different region-based object detection methods, including the R-CNN, Fast R-CNN and Faster R-CNN, were introduced to represent and detect the instances of strawberry flowers. There were various blurred, scale-variant, intra-class variant and inter-class similar objects in the experimental image dataset. The photographs of the flowers were acquired in natural outdoor environments with rich and complex backgrounds. Although the background is usually a distraction for the detection model, sometimes it can supply useful information, so the background content was also considered as feature information for the detection target.

The ultimate goal of this research was to build a deep-level artificial convolutional neural network architecture with a region proposal network and a classification mechanism to accurately detect the regions of strawberry flowers in a field. The deep-level region-based artificial intelligent strawberry flower visual representation and recognition system can offer much value to farmers for the prediction of strawberry yield. Yield prediction is very important for strawberry production in the U.S., since all strawberries are hand-harvested, and labour shortages will soon become a major concern. Over a growing season, strawberry yields vary significantly from time to time depending on temperature and other factors. Therefore, an accurate yield prediction is important to efficiently manage the harvesting labour. An accurate strawberry flower detection system is a critical component that enables yield estimation and mapping by detecting accurate locations for the individual strawberry plants in a field. Precise localization of the strawberry flowers is also an essential part of an automated robotic harvesting system, which can help to reduce one of the most labour-intensive tasks in crop production (Kapach et al. 2012). At present, yield estimation based on manual counting is a very time-consuming and expensive process, and it is unrealistic for large fields. Automatic yield estimation based on robotic technology could be a viable solution.

Materials and methods

Image collection

Bare root strawberry plants (variety: Sensation) were transplanted in an experimental research farm at the University of Florida, Institute of Food and Agricultural Sciences (IFAS) in Wimauma, Florida, USA. The strawberry flower images were acquired with four high-resolution imagers (HDR-AS300 Action Cam, Sony) mounted on an imaging cart, which was custom-built in the Precision Agriculture Laboratory at the University of Florida. Inside the cart, two LED lights (Anywhere Light 20 LED, LightIt, Oregon City, OR) were used to maintain constant illumination. The imaging cart was pulled by a tractor, which traversed different rows of the fields to collect the strawberry flower image data. Figure 1 illustrates some representative instances from a training dataset of distinct

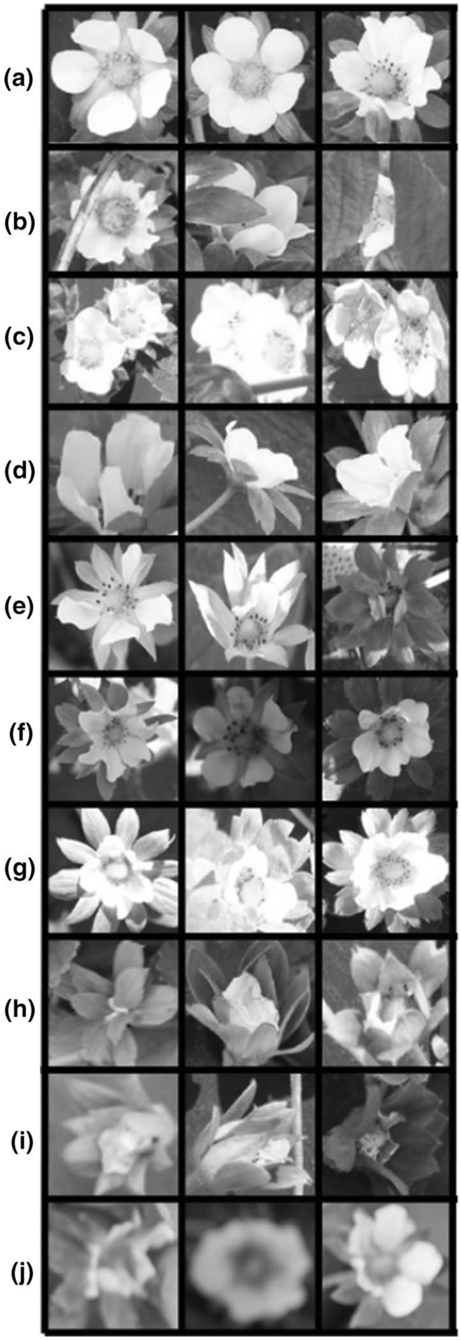

Fig. 1 Illustration of the representative instances in the training dataset of distinct morphological images of strawberry flowers. The rows demonstrate **a** top-view intact, **b** sheltered, **c** overlapped, **d** non-carpel, **e** imperfect, **f** shadow, **g** blazed, **h** white bud-shaped, **i** pink bud-shaped and **j** blurred images of the strawberry flowers

morphological images, including the top-view intact, sheltered, overlapped, non-carpel, imperfect, shadow, blazed, white bud-shaped, pink bud-shaped and blurred images of strawberry flowers. The pictures of the flowers were taken in a natural outdoor environment with a complex and rich background. Although the background is usually a distraction to the detection model, it can sometimes provide useful information, so the background content was also considered as feature information for the detection targets. The networks were trained entirely on 400 strawberry flower images and tested on another 100 images.

Image labelling

To identify the objects in an image using a deep R-CNN, the location and class of the objects need to be determined first. Generally, these instances are basically hand-annotated. Rectangular and circular bounding boxes are often used to label the objects. Circular annotations are needed initially and then converted to rectangular bounding boxes that enclose a target of equal width and height; this is more suitable for round objects, such as citrus fruits and apples. The R-CNN operates on rectangular bounding box prediction, which outputs the corresponding coordinates of the bounding boxes for the object; therefore, the ground truth annotations for the strawberry flowers were collected using rectangular bounding boxes. The graphical image annotation tool, named labelImg and provided by the Computer Science and Artificial Intelligence Laboratory at MIT, was used to label the object bounding boxes in the images. The manually labelled flower objects in the bounding boxes were mainly composed of yellow pollen, white petals and green sepals.

Hardware

The hardware for the R-CNNs used on the strawberry flower training image dataset consisted of an Alienware 17 R3 laptop (DELL, USA) with an NVIDIA GeForce GTX 980 M integrated RAMDAC, 8 GB graphics card and Intel Core(TM) i7-6820HK CPU. The algorithms were programmed in MATLAB R2017a (The Math Works, Natick, USA) under the Windows 10 (Microsoft, USA) operating system. The Caffe model (Chan et al. 2015), which was originally developed by the Berkeley Vision and Learning Center, was used as the deep learning framework. In this experiment, an NVIDIA GTX 980 graphics card with 4 GB memory and 1024 kernels was used for validation.

Region-based CNNs for object detection

A new region-based CNN (R-CNN) framework was introduced for object detection. First, approximately 2000 bottom-up region proposals were generated according to a selective search of the input images. Then, the features of each proposal were extracted by a large convolutional neural network. Finally, the feature vectors were then sent to the linear support vector machines (SVMs) to classify each region and then to a regressor to adjust the detection position. However, each object proposal needs a forward pass through the convolution net, which leads to a heavy computational load. To solve this problem, a spatial pyramid pooling (SPP) network (He et al. 2014) was used. The spatial pyramid build pyramid in image space and quantize feature space. The SPP network enables the input of images with varying sizes or scales during training and generates a full-image representation that may increase the scale invariance and reduce the risk of over-fitting. The proposed Fast R-CNN ran through the CNN exactly once for the input of the image to relieve some of the computational load and speed up the R-CNN (Girshick 2015). Then, a fixed-length feature vector was extracted for each object proposal from the feature map. Each feature vector was sent to the fully connected layers, which output the bounding-box for each object. The process of Fast R-CNN was 213 times faster than that of the R-CNN. However, both the R-CNN and the Fast R-CNN relied on general input object proposals, which usually come from a handmade model, such as a selective search, MultiBox (Szegedy et al. 2014; Erhan et al. 2014) or an EdgeBox model (Dollár and Zitnick 2015). The calculations for generating the proposal regions accounted for most of the computational time of the whole process. Although some of the deeply trained models appeared to generate proposal regions, e.g., the DeepBox model (Kuo et al. 2015), the processing time was still not negligible. Although the running time of the detection networks was reduced by the improved network, the computation time to generate the region proposals was still the bottleneck. Therefore, Ren et al. (Ren et al. 2015) proposed a modified network, called the Faster R-CNN. In this work, a regional proposal network (RPN) was introduced to share the full-image convolutional features with the detection network, which made generating the region proposals very quickly. The RPN and Fast R-CNN were trained to share the convolutional features and optimize the convolution characteristics. The Faster R-CNN can be considered as a system that is composed of regional proposal networks and fast regions with convolutional neural networks (Fast R-CNNs). The RPN substituted the selective search algorithm of the Fast R-CNN. The key to a Faster R-CNN is to share the same convolutional layers of the RPN and Fast R-CNN detector with its own fully connected layers. Then, an entire image was passed through the CNN only once to generate and refine the object proposals. More importantly, because of the shared convolutional layers, a very deep network (Simonyan and Zisserman 2014) could be used to generate high-quality object proposals.

The anti-oscillatory stochastic gradient descent method

The gradient descent algorithm was used to optimize the network parameters in order to minimize the back-propagation error for the training dataset. The gradient descent algorithm updated the parameter vector to minimize the loss function by taking small steps in the direction of the negative gradient of the loss function:

$$\chi_{i+1} = \chi_i - \lambda \nabla \psi(\chi_i) \tag{1}$$

where λ is the learning rate, χ is the parameter vector, $\psi(\chi)$ is the loss function and i denotes the iteration number. The standard gradient descent algorithm sometimes oscillates along the steepest decreasing route to search for the optimum solution. To reduce the oscillation, a momentum coefficient was included in the above gradient descent function:

$$\chi_{i+1} = \chi_i - \lambda \nabla \psi(\chi_i) + \tau(\chi_i - \chi_{i-1}) \tag{2}$$

where $\tau \in [0,1]$ is the momentum coefficient. The normal gradient descent algorithm estimates the gradient of the loss function, $\psi(\chi)$, using the entire dataset simultaneously. The anti-oscillatory stochastic gradient descent algorithm estimates the gradient of the loss function, $\psi(\chi)$, and renews the parameters using a stochastic subset of the dataset (Girshick et al. 2016). In this paper, the number of stochastic subsets used to train the CNN model was set to 10.

Evaluation criteria

The output of the detected images were the bounding boxes of the strawberry flower objects in the images. The correctness of a detected strawberry flower object was evaluated by the intersection-over-union (IoU) overlap with the corresponding ground truth bounding box (Girshick et al. 2016). The IoU overlap was defined as follows:

$$IoU_{overlap} = \frac{Area(GroundTruth \cap Detected)}{Area(GroundTruth \cup \mathrm{Detected})} \tag{3}$$

A detected strawberry flower object was accepted as a true positive (*TP*) if its IoU overlap with the ground truth bounding box was greater than a certain threshold. If a detected strawberry flower object did not match the ground truth bounding box, it was considered a false positive (*FP*). A false negative (*FN*) was determined if the detected strawberry flower object had no matches with the ground truth bounding box. The effectiveness of the R-CNN was evaluated by the precision and recall scores, which were defined as follows:

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

Results and discussion

Network architecture

The R-CNN-based strawberry flower detection models were trained based on the complex deep-level architecture of a VGG19 convolutional neural network that was pre-trained on more than one million images (Liu et al. 2016; Sendik and Cohen-Or 2017). As a result, the model learned rich feature representations for a wide range of images. The network had 47 layers. There were 19 layers with learnable weights: 16 convolutional layers and 3 fully connected layers. To have a pre-trained network perform the new task of detecting the strawberry flower instances, transfer learning (Bengio 2012; Shin et al. 2016; Lin et al. 2018) was used to quickly transfer the learned features to the detection application using a smaller number of training images. Fine-tuning a network with transfer learning is usually much faster and easier than training a network with randomly initialized weights from scratch. One image per batch was randomly sampled for training. The anti-oscillatory stochastic gradient descent method was utilized to solve 50,000 iterations with a basic learning rate of 0.001.

Faster R-CNN

The Faster R-CNN object detection system (Ren et al. 2017) consisted of two modules: first, a regional proposal network (RPN) was employed to detect the region of interest (RoI) in the image, and second, a classification module was used to classify the individual regions and regress a bounding box around the object. During the training process, the inputs to the network were the RGB (red green blue) images of arbitrary sizes and the annotated bounding boxes around each flower. According to the selection of the CNN network, the image data were propagated through a number of convolutional layers. In this study, the deep VGG19 network was used, which contained 16 convolutional layers and 3 fully connected layers. The output of the convolution layers was a high-dimensional feature map because there were 16 strides in the pooling layers. A stride is the step size for moving along the images vertically and horizontally. The local regions in the feature map were propagated forward to two fully connected layers: one was the box-classification layer, which classified the object into the correct category, and the other was the box-regression layer, which refined the location of the bounding box. The individual proposals were propagated through the subsequent fully connected layers and finally through two layers with a finer region classification output and an associated object bounding box. The anti-oscillatory stochastic gradient descent method was used for end-to-end training, which allowed the convolutional layers to be shared between the RPN and the R-CNN components.

During the testing process, the network returned 400 bounding boxes for each image with the class probabilities. The probability threshold method was used with a non-maximum suppression operation to detect the strawberry flower objects (Rothe et al. 2014). Figure 2 shows the intermediate outputs of the Faster R-CNN network for the detection of strawberry flowers. First, an RGB image was propagated through a set of convolutional layers. Each ROI box was propagated through the fully connected layers, finally returned the class probability and regressed a finer bounding box for each object. As shown in the last image of Fig. 2, the ground truth from the input image was used in the RPN and the R-CNN layers during training. During testing, a class-specific detection threshold was applied to the output, which was followed by non-maximum suppression to remove the overlapping results.

Determination of the momentum parameter

Figure 3 shows four curves of the estimated loss function, $\psi(\chi)$, during the iterative optimization process to train the Faster R-CNN object detector to extract regional proposals from the training images using momentum factors of τ = 0.3, 0.5, 0.7 and 0.9. The momentum coefficient was the contribution of the previous gradient change. The contribution of the gradient changes from the previous iteration to the current iteration greatly affected the convergence of the loss function. As the momentum coefficient values increased from τ = 0.3 to 0.9, the convergence performance gradually improved. Although the convergence speed of the curve with τ = 0.3 was better than that with τ = 0.5 and 0.7 at the beginning stage, the convergence performance of the curve with τ= 0.3 was obviously shocked

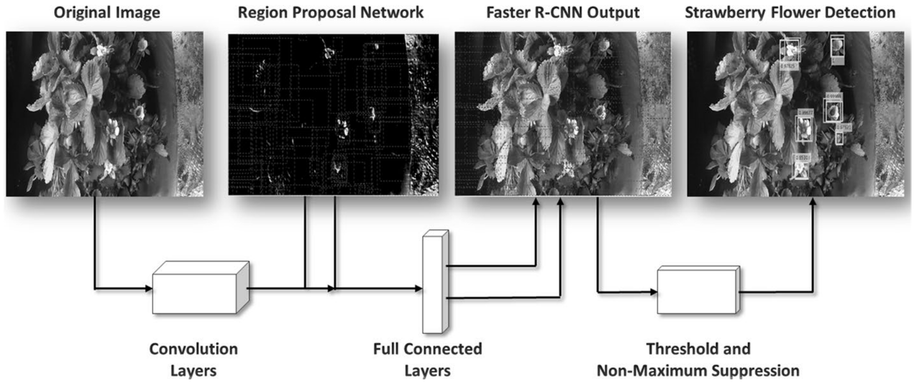

Fig. 2 Schematic diagram of the deep-level Faster R-CNN network architecture for the visual representation of the detection of strawberry flowers

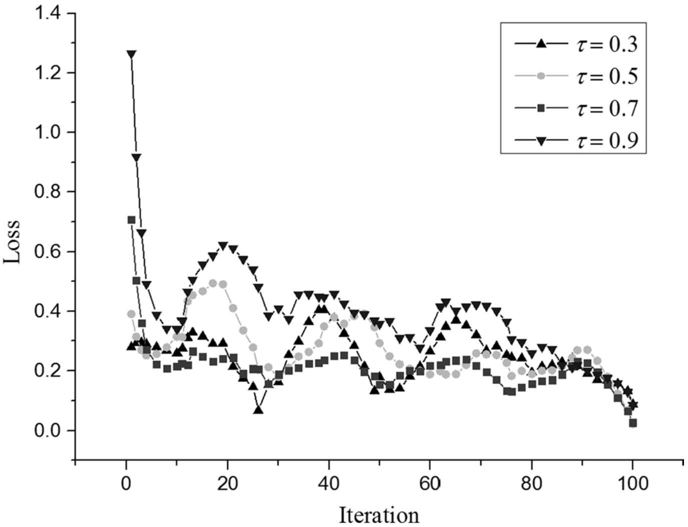

Fig. 3 Four curves of the estimated loss function, $\psi(\chi)$, for the iterative optimization process to train the Faster R-CNN model to extract region proposals from the training images with momentum coefficient values of $\tau = 0.3$, 0.5, 0.7 and 0.9

severely in the iterations from 20 to 80. The oscillation degrees of the curves with momentum coefficient values of τ= 0.3 and 0.5 were greater than that with a coefficient value of τ= 0.7 after 10 iterations. When the number of iterations was greater than 90, the convergence of each curve tended to be consistent for all momentum factor values. A continuous increase in the momentum coefficient would improve the convergence performance; however, it would also increase the training time, so it was not recommended. As the momentum coefficient values increased from $\tau = 0.7$ to 0.9, the convergence performance became worse. This indicated that the momentum coefficient was able to suppress the oscillation when the algorithm searched for the optimum solution along the convex route. Therefore, the momentum parameter value of $\tau = 0.7$ was chosen for the anti-oscillatory stochastic gradient descent function for training the Faster R-CNN model.

RPN performance

High-quality region proposals had an important contribution to improving the object detection performance. In this paper, the influence of the number of regional proposals on the detection accuracy was also investigated. Figure 4 shows that the performance of the network was almost saturated after generating 300 proposals. Although there was a slight improvement at 400 proposals, the calculations took a longer time. The maximum detection rate was 86.1% by using 400 proposals.

Comparison of the region-based CNN methods

In this section, the strawberry flower detection performances of the R-CNN, the Fast R-CNN, and the Faster R-CNN on the strawberry flower dataset were compared. The top 2000 proposals generated by the selective search method were used for the R-CNN and the

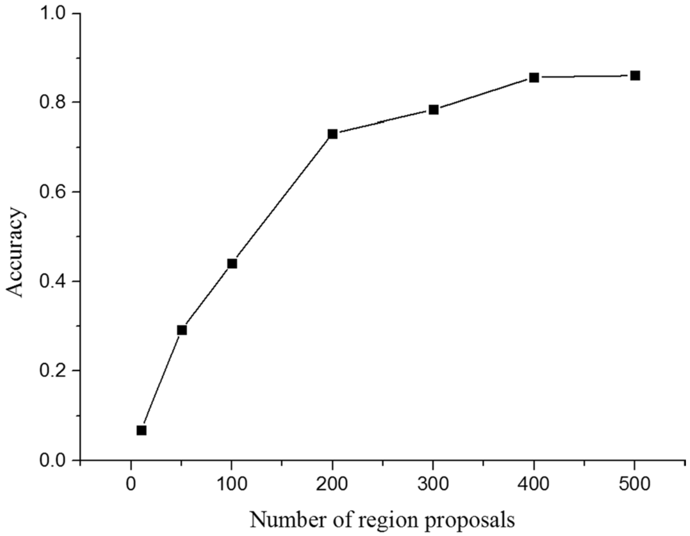

Fig. 4 The influence of the number of regional proposals used in training the detection accuracy of the Faster R-CNN model

Fast R-CNN. The R-CNN model was trained end-to-end for both classification and regression (Yang et al. 2015). The precision-recall metric (Tang et al. 2016) was used to estimate the quality of strawberry flower detection. The precision-recall curve showed the trade-off between precision and recall for different thresholds. The high precision was related to a low false positive rate, and the high recall was related to a low false negative rate. The high scores indicated that the classification model obtained accurate results and the majority of the positive results. As shown in Fig. 5, as the threshold of the recall rates increased, the corresponding precision rates of the Faster R-CNN became much higher than those of the other two algorithms (the R-CNN and Fast R-CNN). The overall performances of the algorithms were measured with the mean average precision (mAP) score (Davis and Goadrich 2006), which is the average precision at the ranks where the recall changed. The geometric interpretation of the mAP score is the area below the curve. A large area below the precision-recall curve indicated the overall superior performance of the algorithm with the higher mAP score. In other words, a curve that is above another curve had a better performance level. The Faster R-CNN-based model achieved the highest mAP score of 0.861 on the test strawberry flower image dataset, as shown in Table 1. The Faster R-CNN fused the RPN and Fast R-CNN into a single network by sharing their convolutional features with an attention mechanism to guide the unified network as to where to exactly orient and sharply cut down on the number of invalid bounding boxes, which finally improved the accuracy of the objection detection algorithm (Chu et al. 2018). The compared results illustrated that the improvement of the Faster R-CNN model for strawberry flower detection was substantial. More detailed features were abstracted effectively from the original images by using the Faster R-CNN compared with the other two algorithms.

The RPN technology adopted by the Faster R-CNN, which shared the full-image convolutional features with the detection network to detect the RoI in the image, reduced the running time of the detection networks and reached the target for real-time object detection (Chu et al. 2018). The Faster R-CNN routines, which could perform at an

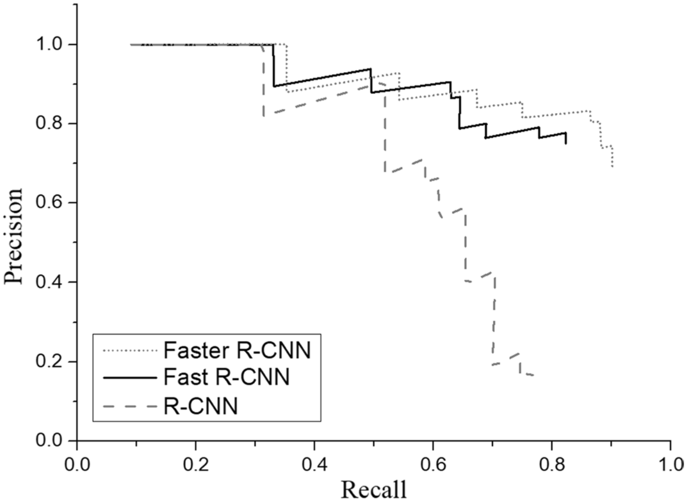

Fig. 5 Precision recall curves of the detection results for the R-CNN, Fast R-CNN, and Faster R-CNN models

Table 1 Detection results of three different R-CNN models with a selective search (SS) or a regional proposal network (RPN) in terms of mean average precision (mAP), execution time and frame per second (FPS)

Method	Proposals		MAP	Time (s)	FPS
R-CNN	SS	2000	0.634	7.226	0.138
Fast R-CNN	SS	2000	0.767	2.709	0.369
Faster R-CNN	RPN	400	0.861	0.118	8.475

8.475 frames per second (FPS) detection rate, ran much faster than the R-CNN and Fast R-CNN routines, which performed at detection rates of 0.138 FPS and 0.369 FPS, respectively, as summarized in Table 1. It was clear that the performance of the Faster R-CNN model exceeded those of the R-CNN and Fast R-CNN models.

As shown in Fig. 6, more instances are illustrated for comparing detection performance using the three different models of the R-CNN, Fast R-CNN and Faster R-CNN. Each column, from left to right, in Fig. 6 shows the detection results of the three models. Figure 6a shows that there were four strawberry flowers in the image, but the R-CNN detected only two of them. The other two small strawberry flowers at the positions of ‘a’ and ‘b’ were undetected. The Fast R-CNN had a similar result. As shown in Fig. 6d, the two small strawberry flower instances were also undetected by the Fast R-CNN model. However, as shown in Fig. 6g, the Faster R-CNN successfully detected all four flower instances, including the two small ones previously undetected by the other two models. Figure 6b shows that there was only one flower in the image, but the R-CNN incorrectly detected the bright sunlight spot as a flower. However, Fig. 6e, h both show that the Fast R-CNN and Faster R-CNN detectors produced correct

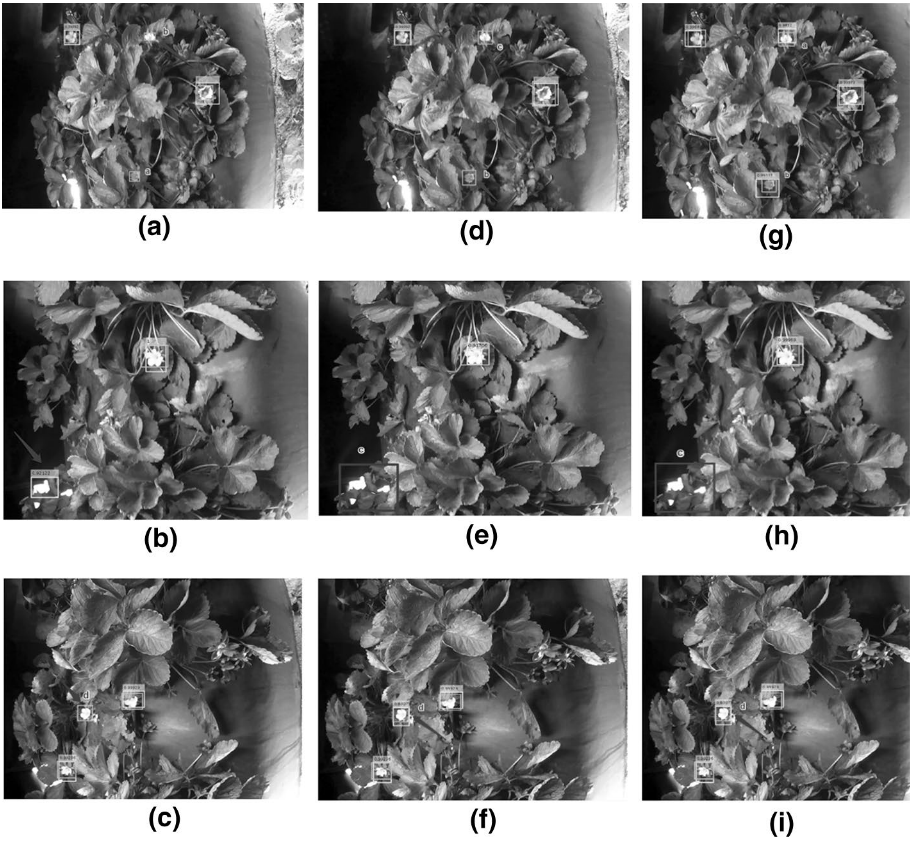

Fig. 6 Comparison of the detection results of three different models: R-CNN (**a–c**), Fast R-CNN (**d–f**) and Faster R-CNN (**g–i**)

detection results. Figure 6c shows that the R-CNN model had difficulty identifying the connected two flower objects; however, Fig. 6f, i shows that all three flowers were correctly detected by the Fast R-CNN detector and Faster R-CNN detector, respectively.

The qualitative strawberry flower detection results are illustrated in Fig. 7. It can be observed that the Faster R-CNN model could address challenging issues, such as different illumination and overlap in Fig. 7a, e, a shelter in Fig. 7b, complex backgrounds in Fig. 7c and different flower orientations in Fig. 7d. In Fig. 7c at the 'a' position, the strawberry flower was covered by the opposite side of the leaves, and there were not enough data for the opposite leaf side when training the model; however, it was finally identified by the Faster-RCNN model, which demonstrated the robustness of the algorithm. In Fig. 7, some failure cases of the Faster R-CNN model on the strawberry flower testing dataset are shown. In Fig. 7d at position 'b', one tiny strawberry flower was not identified by the algorithm because it was too small compared to the other instances in the dataset and its orientation. In the future work, the integration of more cues and instances for training the network will be needed to detect the missing flowers to increase the detection accuracy.

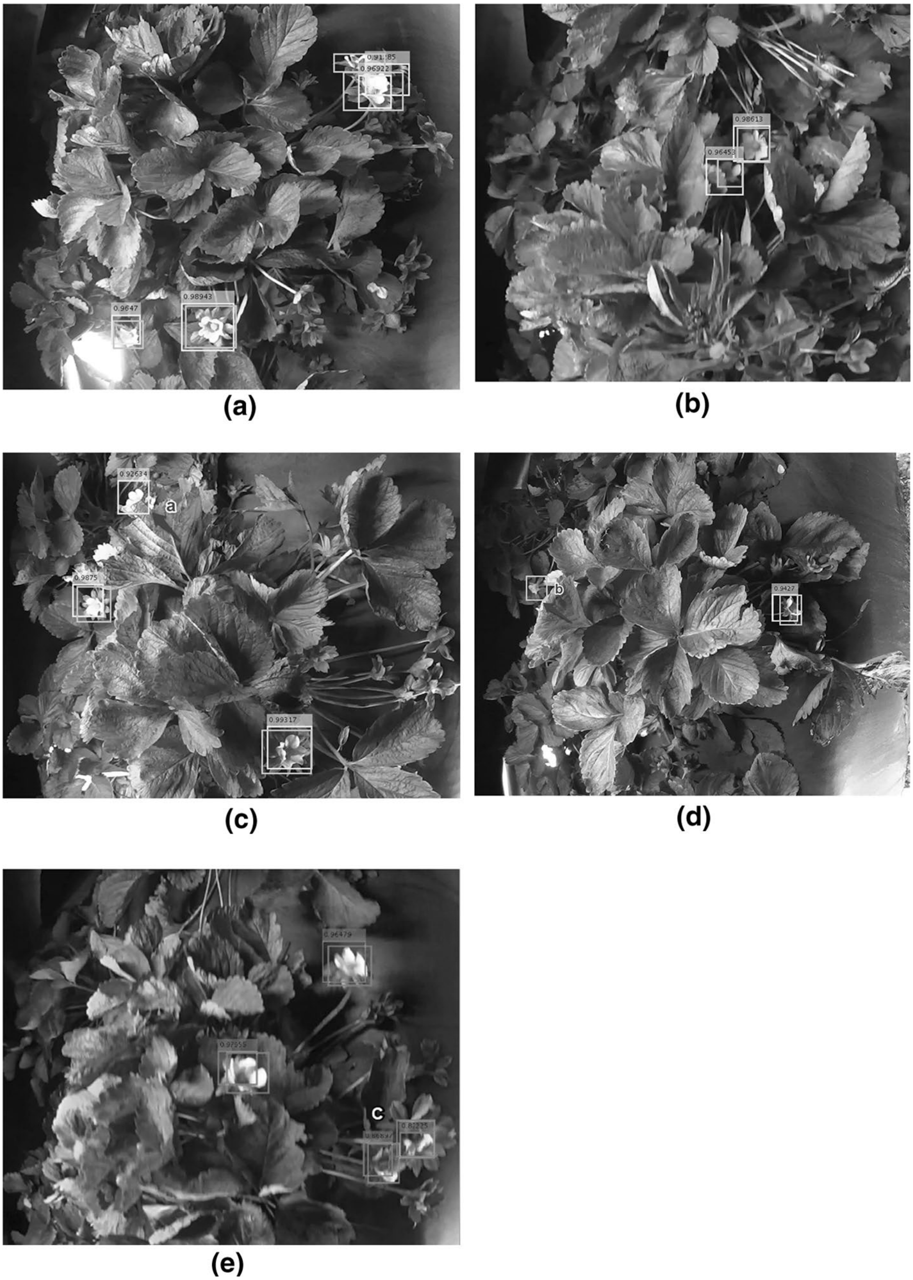

Fig. 7 Examples of strawberry flower detection results in various instances using the faster R-CNN model

Conclusions

In this paper, a state-of-the-art deep-level region-based visual representation architecture was proposed for the detection of strawberry flowers in an outdoor field. The proposed algorithm was able to efficiently complete the detection task even if the strawberry flowers were under a shadow, obscured by foliage and stems, or overlapped by other strawberry flowers. The transfer learning technology was used to quickly transfer the learned features to the detection application by using fewer training images. It was much faster and easier to train the network by fine-tuning it through transfer learning than to train the network with random initialization. The experimental results showed that the Faster R-CNN effect came from the RPN module. Due to sharing between the convolution layer RPN and the Fast R-CNN detector module, the Faster R-CNN model could use the RPN within the multiple convolution layers without an additional computational burden. The proposed algorithm achieved a superior detection accuracy (mean average precision) of 86.1%. The developed deep-level region-based artificial intelligent strawberry flower visual representation and recognition system in such a setting has the potential to offer much value to farmers. With accurate knowledge of the individual strawberry flower locations in the field, strawberry yield estimation and prediction would be possible, which is important and beneficial for growers to efficiently utilize their labour resources and increase yield and profit.

Acknowledgements This study was supported by the National Natural Science Foundation of China (Grant No. 31601227, 31501221), Natural Science Foundation of Jiangsu Province (Grant No. BK20161310, BK20140467), Jiangsu Government Scholarship for Overseas Studies (Grant No. JS-2015-065), Yancheng Agricultural Science and Technology Guidance Program (Grant No. YKN2015019, YKN2015021), Yancheng Institute of Technology Breeding Programs (Grants No. KJC2014006, KJC2014007), and the Florida Strawberry Research and Education Foundation in USA (Grant No. AGR00008886).

References

Arel, I., Rose, D. C., & Karnowski, T. P. (2010). Deep machine learning-a new frontier in artificial intelligence research [research frontier]. *IEEE Computational Intelligence Magazine, 5*(4), 13–18.

Bac, C., Hemming, J., & Van Henten, E. (2013). Robust pixel-based classification of obstacles for robotic harvesting of sweet-pepper. *Computers and Electronics in Agriculture, 96,* 148–162.

Bargoti, S., & Underwood, J. P. (2017). Image segmentation for fruit detection and yield estimation in apple orchards. *Journal of Field Robotics, 34*(6), 1039–1060.

Bengio, Y. (2012). Deep learning of representations for unsupervised and transfer learning. In *Proceedings of ICML workshop on unsupervised and transfer learning* (pp. 17–36).

Chan, T. H., Jia, K., Gao, S., Lu, J., Zeng, Z., & Ma, Y. (2015). PCANet: A simple deep learning baseline for image classification. *IEEE Transactions on Image Processing, 24*(12), 5017–5032.

Chaudhry, S., & Chandra, R. (2017). Face detection and recognition in an unconstrained environment for mobile visual assistive system. *Applied Soft Computing, 53,* 168–180.

Choi, D., Lee, W., Ehsani, R., Schueller, J., & Roka, F. (2016). Detection of dropped citrus fruit on the ground and evaluation of decay stages in varying illumination conditions. *Computers and Electronics in Agriculture, 127,* 109–119.

Chu, W., Liu, Y., Shen, C., Cai, D., & Hua, X. S. (2018). Multi-Task Vehicle Detection With Region-of-Interest Voting. *IEEE Transactions on Image Processing, 27*(1), 432–441.

Cireşan, D. C., Giusti, A., Gambardella, L. M., & Schmidhuber, J. (2013). Mitosis detection in breast cancer histology images with deep neural networks. *International conference on medical image computing and computer-assisted intervention* (pp. 411–418). Heidelberg: Springer.

Dairi, A., Harrou, F., Senouci, M., & Sun, Y. (2018). Unsupervised obstacle detection in driving environments using deep-learning-based stereovision. *Robotics and Autonomous Systems, 100,* 287–301.

Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and ROC curves. In *Proceedings of the international conference on machine learning (ICML).*

Dollár, P., & Zitnick, C. L. (2015). Fast edge detection using structured forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 37*(8), 1558–1570.

Erhan, D., Szegedy, C., Toshev, A., & Anguelov, D. (2014). Scalable object detection using deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2147–2154).

Girshick, R. (2015). Fast R-CNN. In *The IEEE international conference on computer vision (ICCV)* (pp. 1440–1448).

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2016). Region-based convolutional networks for accurate object detection and segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 38*(1), 142–158.

He, K., Zhang, X., Ren, S., & Sun, J. (2014). Spatial pyramid pooling in deep convolutional networks for visual recognition. *European conference on computer vision* (pp. 346–361). Heidelberg: Springer.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 37*(9), 1904–1916.

Kapach, K., Barnea, E., Mairon, R., Edan, Y., & Ben-Shahar, O. (2012). Computer vision for fruit harvesting robots—State of the art and challenges ahead. *International Journal of Computational Vision and Robotics, 3*(1–2), 4–34.

Kuo, W., Hariharan, B., & Malik, J. (2015). Deepbox: Learning objectness with convolutional networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 2479–2487).

Lin, P., Chen, Y., & He, Y. (2012). Identification of broken rice kernels using image analysis techniques combined with velocity representation method. *Food and Bioprocess Technology, 5*(2), 796–802.

Lin, P., Li, X., Chen, Y., & He, Y. (2018). A deep convolutional neural network architecture for boosting image discrimination accuracy of rice species. *Food and Bioprocess Technology, 11*(2), 1–9.

Liu, G., Gousseau, Y., & Xia, G. S. (2016). Texture synthesis through convolutional neural networks and spectrum constraints. In *2016 IEEE 23rd international conference on pattern recognition (ICPR)* (pp. 3234–3239).

Nuske, S., Achar, S., Bates, T., Narasimhan, S., & Singh, S. (2011). Yield estimation in vineyards by visual grape detection. In *2011 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 2352–2358).

Nuske, S., Wilshusen, K., Achar, S., Yoder, L., Narasimhan, S., & Singh, S. (2014). Automated visual yield estimation in vineyards. *Journal of Field Robotics, 31*(5), 837–860.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems* (pp. 91–99).

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(6), 1137–1149.

Rothe, R., Guillaumin M., & Gool, L. V. (2014). Non-maximum suppression for object detection by passing messages between Windows. In *Asian conference on computer vision 2014* (pp. 290–306).

Sendik, O., & Cohen-Or, D. (2017). Deep correlations for texture synthesis. *ACM Transactions on Graphics (TOG), 36*(5), 161.

Shin, H. C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., et al. (2016). Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Transactions on Medical Imaging, 35*(5), 1285–1298.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556.

Szegedy, C., Reed, S., Erhan, D., Anguelov, D., & Ioffe, S. (2014). Scalable, high-quality object detection. arXiv preprint arXiv:14121441.

Tang, B., He, H., Baggenstoss, P. M., & Kay, S. (2016). A Bayesian classification approach using class-specific features for text categorization. *IEEE Transactions on Knowledge and Data Engineering, 28*(6), 1602–1606.

Uijlings, J. R., Van de Sande, K. E., Gevers, T., & Smeulders, A. W. (2013). Selective search for object recognition. *International Journal of Computer Vision, 104*(2), 154–171.

Underwood, J. P., Hung, C., Whelan, B., & Sukkarieh, S. (2016). Mapping almond orchard canopy volume, flowers, fruit and yield using LiDAR and vision sensors. *Computers and Electronics in Agriculture, 130,* 83–96.

Wang, Q., Nuske, S., Bergerman, M., & Singh, S. (2013). Automated crop yield estimation for apple orchards. *Experimental robotics* (pp. 745–758). Heidelberg: Springer.

Wen, Y., Zhang, K., Li, Z., & Qiao, Y. (2016). A discriminative feature learning approach for deep face recognition. In *European conference on computer vision* (pp. 499–515). Cham: Springer.

Yamamoto, K., Guo, W., Yoshioka, Y., & Ninomiya, S. (2014). On plant detection of intact tomato fruits using image analysis and machine learning methods. *Sensors, 14*(7), 12191–12206.

Yang, S., Luo, P., Loy, C. C., & Tang, X. (2015). From facial parts responses to face detection: A deep learning approach. In *Proceedings of the IEEE international conference on computer vision* (pp. 3676–3684).

Zitnick, C. L., & Dollár, P. (2014). Edge boxes: Locating object proposals from edges. *European conference on computer vision* (pp. 391–405). Cham: Springer.

Zou, X., Zou, H., & Lu, J. (2012). Virtual manipulator-based binocular stereo vision positioning system and errors modelling. *Machine Vision and Applications, 23*(1), 43–63.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.