Journal of Agricultural and Crop Research
Vol. 8(11), pp. 260-271, November 2020
doi: 10.33495/jacr_v8i11.20.180
ISSN: 2384-731X
Research Paper

Data augmentation method for strawberry flower detection in non-structured environment using convolutional object detection networks

Umme Fawzia Rahim* • Hiroshi Mineno

Graduate School of Integrated Science and Technology, Shizuoka University, 3-5-1 Johoku, Naka-ku, Hamamatsu, Shizuoka 432 – 8081, Japan.

*Corresponding author. E-mail: fawzia@minelab.jp. Tel: +81 080 5492 3240.

Accepted 25th August, 2020.

Abstract. Deep learning has demonstrated significant capabilities for learning image features and presents many opportunities for agricultural automation. Deep neural networks typically require large and diverse training datasets to learn generalizable models. However, this requirement is challenging for applications in agricultural automation systems, since collecting and annotating large amount of training samples from filed crops and greenhouses is an expensive and complicated process due to the large diversity of crops, growth seasons and climate changes. This research proposed a new method for augmenting training dataset using synthesized images that preserves the background context and texture of the data object. A synthetic dataset of 1800 images was generated using a reference dataset and applying image processing techniques. As reference dataset 100 and for evaluating detection performance 230 real images of strawberry flowers were collected in greenhouses. Experimental results demonstrated that the suggested method provides improved performance when applied to the state-of-the-arts convolutional object detectors including Faster R-CNN, SSD, YOLOv3 and CenterNet for the task of strawberry flower detection in non-structured environment. The YOLOv3 w/darknet53 model achieved 46.84% boost in performance with average precision (AP) improved from 39.20% to 86.04% when applied augmentation using synthetic dataset. The AP of Faster R-CNN w/resnet50, SSD w/resnet50 and FPN and CenterNet w/hourglass52 models improved by 15.71, 18.42 and 22.24%, respectively. The Faster R-CNN w/resnet50 model provided most significant strawberry flower detection performance with AP 90.84%, which is higher than SSD w/resnet50 and FPN, YOLOv3 w/darknet53 and CenterNet w/hourglass52 models (88.56%, 86.04 % and 83.82%, respectively).

Keywords: Flower detection, deep convolutional neural network, data augmentation, synthetic dataset.

INTRODUCTION

In recent years, automation in agriculture has motivated by concerns over increasing demand of productivity and quality of food production whilst decreasing the pressure on resources required (Bac *et al.*, 2014). Intelligent agriculture has become a popular concept (Tyagi, 2016), and crop imaging has turned out to be an important means of collecting crop growth information (Zhao *et al.*, 2016). Detecting objects using off-the-shelf RGB cameras and computer vision in field conditions is a key requirement for automating many tasks in agriculture such as automatic harvest of fruits and vegetables, water and nutrition control and management, artificial pollination and yield estimation. Varying illumination conditions, complex and cluttered background, the camera's viewing angle and distance, and other factors in fields and greenhouses can have certain impacts on target detection in agricultural scene (Kapach *et al.*, 2012). Therefore, any algorithm that depends on parameters which are hand-tuned to a specific crop features is at risk of being overly specified.

Unlike hand-engineered computer vision pipelines, deep convolutional neural networks (LeCun *et al.*, 2015) combine multiple convolutional layers and down sampling techniques to learn a hierarchical representation of the data, and therefore, the network can adapt to be invariant to various types of diversities in the image data. State-of-the-art computer vision systems based on deep convolutional neural networks can deal with variations in lighting conditions, pose, shape, large inter-class diversity and occlusions (He *et al.*, 2016; Krizhevsky *et al*., 2012; Simonyan and Zisserman, 2014), essential features needed for robust detection of objects in complex agricultural environment. However, the migration from hand-engineered computer vision pipelines to deep learning comes with some limitations. While convolutional neural networks (CNNs) have the representational capability to learn complex models, the success of these representations relies on the quality and quantity of the training samples. In most computer vision-based applications where CNNs show a significant progress over hand-engineered methods, such as image segmentation, classification, and object detection in a scene, the size of the training dataset is typically on the order of tens of thousands to tens of millions of images (Deng *et al*., 2009). This allows for much diversity in the training samples, and very robust learned models as a consequence. The collection and labeling of large amount of data from filed crops and greenhouses is an expensive and complicated process due to the large variety of crops, growth seasons, climate change and phenotype diversity. The cost of generating new data and the limitations of naturally generated datasets has motivated the use of an alternative source of data to train deep networks for object detection task for applications in intelligent agriculture and farm automation.

Recent surveys on plant phenotyping emphasize the importance of data augmentation and synthetic data for training networks (Kamilaris and Prenafeta-Boldú, 2018). Synthetic data modeling and graphical rendering play an important role in plant phenotyping and genotyping (Douarre *et al*., 2018). In Dyrmann *et al.* (2016), the network was entirely trained on synthetic images generated using manually segmented maize plants and weeds pasted on bare soil images. They achieved high pixel accuracy in classifying maize and weeds when tested on real images. However, manual segmentation of weeds and maize from raw images is labor intensive. Ubbens *et al.* (2018) employed graphical modeling and introduced a parametric version of L-systems for modeling synthetic rosettes. They argue that images of real and synthetic plants are significantly interchangeable during training a neural network. However, their model is applicable only in structured environment. Ward *et al*. (2018), applied domain randomization to produce synthesize Arabidopsis images. Although they applied random camera positions and lighting to generate the images, and randomized leaf positions in a unit sphere. The main drawbacks of this method are image textures that have a cartoon-like appearance, and it does not handle background complexity.

Several researchers employed basic augmentation methods such as rotations (Namin *et al*., 2018), cropping and flipping/mirroring (Dcunha *et al*., 2017), scaling (De Brabandere *et al.*, 2017) and color transformation (Dias *et al*., 2018); and achieved improved performance in classification, target detection and instance segmentation task in agriculture. These transformations provide limited number of augmentations, therefore, cannot help much in conditions when a small number of training samples are available.

Accurate and robust flower detection is the key step to ensure reliable yield estimation and for the development of optimized plant management system for use in applications of intelligent agriculture. Despite the importance of flower detection in intelligent agricultural systems and farm automation, there has been limited progress in automatic flower detection in non-structured agricultural environment. Most of the existing works based on simple color thresholding technique (Adamsen *et al.*, 2000; Hočevar *et al.*, 2014; Oppenheim *et al*., 2017; Thorp and Dierig, 2011) have their applicability hindered especially by variable illumination conditions and occlusions by other flowers or foliage. Recently two works have applied CNNs for the task of cotton flower (Xu *et al*., 2018) and apple flower (Dias *et al*., 2018) detection in outdoor field and orchards. Inspired by successful researches using CNNs in several computer vision tasks, previous work by Rahim and Mineno (in press) employed the Faster R-CNN to detect and count tomato flowers in greenhouses and produced good results.

This study proposed a method of data augmentation preserving texture of target object (strawberry flower) and background context as close as possible to imaged flowers in real greenhouse scenes. Using a small number of real images of strawberry flowers and plants, segmented strawberry flowers, geometric transformations and image processing techniques, a large diverse set of synthetic images of strawberry flowers in dense cultivation in greenhouse was generated. Among these, some techniques can be considered global, like color thresholding, rotations and scaling, while some are tailored specifically for the particular dataset, for example, number of flowers and their size variations in images. Using the synthesized images alongside real training data, this work demonstrated the applicability of the proposed method to boost performances of modern convolutional object detection networks including Faster

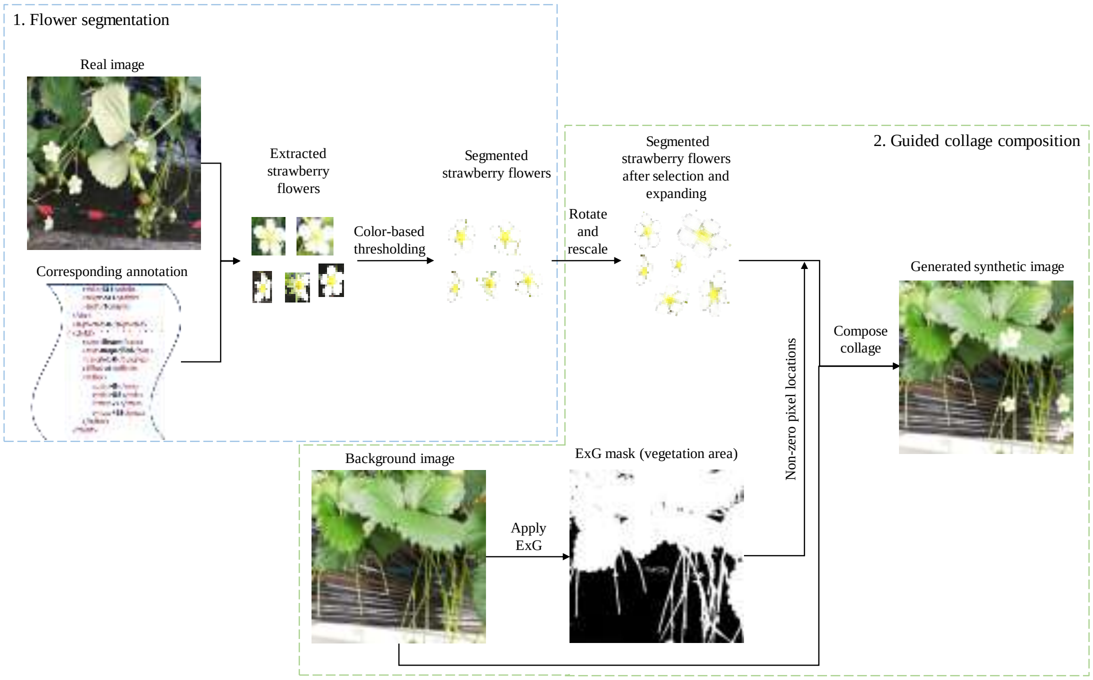

1. Flower segmentation
Real image
Corresponding annotation
Extracted strawberry flowers
Color-based thresholding
Segmented strawberry flowers
Rotate and rescale
Segmented strawberry flowers after selection and expanding
2. Guided collage composition
Generated synthetic image
Compose collage
Non-zero pixel locations
Background image
Apply ExG
ExG mask (vegetation area)

Figure 1. Synthetic image generation pipeline.

R-CNN (Ren *et al.*, 2015), SSD (Liu *et al.*, 2016), YOLOv3 (Redmon and Farhadi, 2018) and CenterNet (Duan *et al.*, 2019) in accurately detecting strawberry flower in non-structured environment.

MATERIALS AND METHODS

This work seeks to demonstrate that realistic synthetic images can be used in conjunction with real data to train deep CNNs to accurately detect strawberry flowers in the images.

Empirical reference dataset

Empirical data was a foundation for two objectives. First, it was used as a source of segmented strawberry flowers and background images and as a reference to create realistic conditions e.g., flower numbers and sizes in the images to generate the synthetic dataset. Second, to provide training dataset and an evaluation test set for object detection networks that use the synthetic dataset for boosting performance.

The empirical image dataset was acquired using a smartphone camera (Huawei P20 Lite) with 4608 × 3456 pixels resolution under daylight conditions from different distances and angles. The greenhouses are located in Shizuoka, Japan. The first collection was taken in the research and development greenhouses owned by Shizuoka Prefectural Agriculture and Forestry Research Institute on the 5th of February 2020 and contained 100 images of strawberry flowers and 65 images of strawberry plants without any flower. The second collection, which contained 180 images of strawberry flowers was taken on the 19th of February 2020 from the same institute (different greenhouses). The third collection, which contained another 50 strawberry flower images was captured in a commercial greenhouse on the 24th of February 2020. All image collections included various cultivars of strawberry flowers in dense cultivation. The images from the second and third collections were used for evaluation purpose and not included in training set. The entire empirical dataset was labeled manually using LabelImg graphical image annotation tool. Most visible flowers in the image were labeled by rectangle bounding boxes. Very small and blurry flowers were disregarded and were not labeled.

Synthetic image generation pipeline

The overall synthetic image generation pipeline can be divided into two key steps: (1) flower segmentation, (2) guided collage composition (Figure 1). The final output of this pipeline is synthetic strawberry flower images in dense cultivation those simulate real greenhouse environment as close as possible and their corresponding annotations.

Flower segmentation

In order to preserve the texture of target object in the

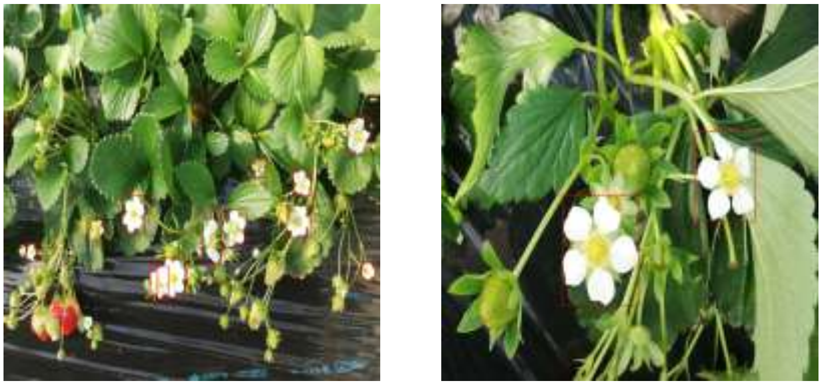


Figure 2. Real strawberry flowers with manual annotations.

synthetically generated images, segmented strawberry flowers from the real images were used to compose new images. Each annotated flower in the raw image were segmented from its background. The raw data consisted of 100 high resolution RGB images of strawberry flowers with high quality manual annotation (rectangle bounding box per object) of most visible flowers (Figure 2). Using corresponding annotation information all the box regions containing a flower were extracted from the raw image and then color-based thresholding was applied to each extracted region to segment strawberry flowers from its background. The segmentation was performed over the HSV color space as it has been proven to be useful in many color-based algorithms for segmentation (Aquino *et al*., 2015; Bairwa and Agrawal, 2014; Kaur and Porwal, 2015; Plebe and Grasso, 2001). Segmentation was done based on the fact that strawberry flowers are white with green yellow carpel and stamen.

Determination of threshold values in a segmentation process is a crucial part, since each pixel segmented out of the image is considered as non-flower pixel and is not taken into consideration in the following steps, even if it a flower's pixel and vice versa. Hence, in order to determine the threshold values over the H (hue), S (saturation) and V (value) components correctly, a sampling program was written using python OpenCV 3.4.1, and parameters were then chosen empirically. For white flower parts (corolla), the low threshold value of the hue component chosen was 0 and the high threshold was chosen to be 255, the low threshold value of the saturation component chosen was 0 and the high threshold was chosen to be 75 and the low threshold value of the value component chosen was 168 and the high threshold was set to 255, since these thresholds segment white corollas well with low noise. For green yellow flower parts (carpel and stamen), the low threshold value of the hue component chosen was 22 and the high threshold was chosen to be 34, the low threshold value of the saturation component chosen was 65 and the high threshold was set to 255 and the low threshold value of the value component chosen was 115 and the high threshold was set to 255, since these thresholds segment green yellow flower parts relatively well with minimum noise. Next, the two segmented regions are joined using an OR operator so the whole flower is segmented. Finally, backgrounds of all the segmented flowers were made transparent.

Guided collage composition

The guided collage generation is composed of carefully positioning segmented strawberry flowers on top of selected background images. From the original set of segmented strawberry flowers, we selected a set of 'suitable' flowers, by two criteria: (a) clear appearance (without/minimum noise) and (b) not occluded by other flowers or foliage. The resulting set consists of ~100 flowers (out of ~550 original flowers) (Figure 3). Occluded flowers were discarded because in the real environment it is uncommon to see cut flowers and the partial appearance of flowers is a side effect of the collage procedure. Scaling and rotation operations were applied on the selected set of flowers in order to expand flower number and create more variations in appearance. Each flower was rescaled to in a random range between 18 to 145 pixels (chosen based on the size of smallest and largest segmented flower) in larger dimension, preserving the aspect ratio. The rotation angle of each segmented flower was randomly selected in the range of 0 to 359 degrees. The expanded set consists of ~500 images.

As background for the synthetic images we used a set of 65 images of strawberry plants in greenhouse in dense cultivation without any visible strawberry flowers (Figure 4) and their mirrored images. The rationale for selection of background is based on our intention of simulation of real greenhouse environment as close as possible and preserving background complexity.

In the next step, Excess Green (ExG) vegetation indices (Woebbecke *et al*., 1995) was applied on the background image to distinguish pixels of strawberry plants from non-plant background. The ExG indexing process produced a binary mask with 1 indicating pixels that fell within the plant area and 0 otherwise. This mask

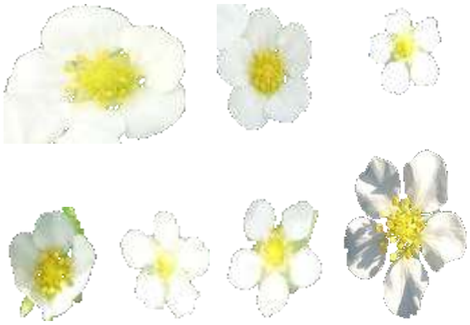


Figure 3. Examples of segmented strawberry flowers used for guided collage composition.

Figure 4. Examples of strawberry plant images used as backgroung for composing synthetic image.

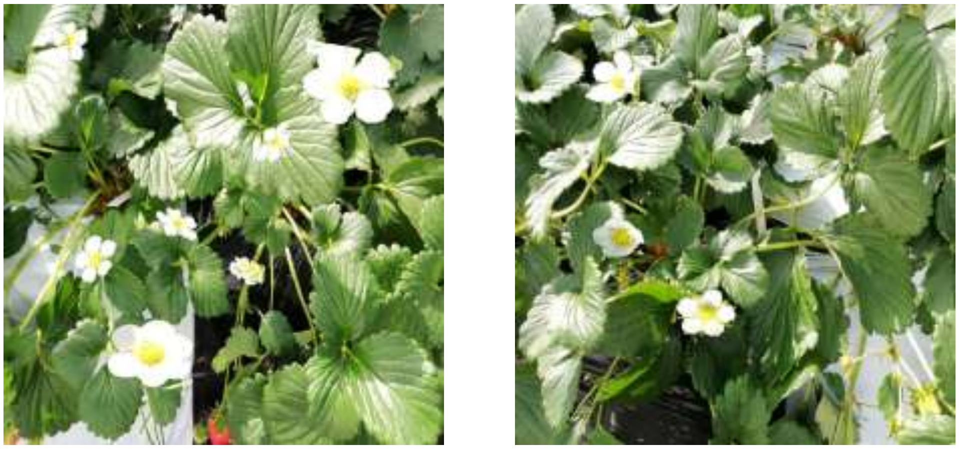


Figure 5. Examples of generated synthetic images used to augment training set.

guides the collage composition process and prevents positioning segmented strawberry flowers on non-plant area on the background image.

Finally, the collage was created by pasting 2 to 4 large flowers (larger dimension >72 pixels) or 3 to 12 small (larger dimension <= 72 pixels) flowers (random number in arbitrary but fixed range) at random non-zero pixel locations (plant area) on top of the background image (512 × 512). The location of each flower was randomly selected inside plant area, and the only restriction was that the flower center remains outside the area of a previously positioned flower, allowing maximum 50% overlap due to our deliberate intention of accurately detecting flowers under up to 50% overlap condition. Some samples of generated images are shown in Figure 5. Parallel to image composition, we generated corresponding annotation for every synthetic image.

Convolutional object detection networks

In this augmentation experiment, four state-of-the-arts convolutional object detection networks were employed for the task of strawberry flower detection in order to evaluate the applicability of synthetic images for improving their performance.

Faster R-CNN

The Faster Region-based Convolutional Neural Network (Faster R-CNN) object detection system (Ren *et al*., 2015) composed of two modules: 1) a Region Proposal Network (RPN) which proposes regions that may contain objects and 2) a classification module which classifies the individual regions and regress a bounding box around the object. Images are propagated through a feature extractor (e.g. Resnet50) and generated high dimensional feature map is feed into the RPN network. The RPN produces up to a predefined number of box proposals. In the next stage, these box proposals are used to crop those features which would correspond to the relevant objects from the same feature map. Finally, individual feature maps are propagated through subsequent two sibling fully connected layers (the classification module), in order to predict an object class and associated finest bounding box. The detection time depends on the number of region proposals generated by RPN.

In this study, Faster R-CNN with Resnet50 (He *et al*., 2016) was adopted for the task of strawberry flower identification in images captured in greenhouses. The Tensorflow implementation of Faster R-CNN of object detection API (Huang *et al*., 2017) was used.

YOLO

The You Only Look Once (YOLO) object detection framework (Redmon *et al*., 2016) unifies target classification and localization into a regression problem. A YOLO network does not require RPN, and it produces bounding box coordinates and probabilities of each class directly through regression. The network splits each image in the training set into S × S grids. If the center of the object's ground truth falls in a grid, then the grid is responsible for detecting the existence of that object. Each grid predicts the location of bounding boxes, their confidence scores, and class conditional probabilities. The confidence score indicates the likelihood that the grid contains an object.

The YOLOv2 (Redmon and Farhadi, 2017) adopts the idea of the "anchor box" in Faster R-CNN and uses k-means clustering method to generate suitable priori bounding boxes. It also introduces batch normalization, a high-resolution classifier, dimension clusters, direct location prediction, fine-grained features and multi-scale training methods that enormously increases the detection accuracy compared with YOLO. YOLOv3 (Redmon and Farhadi, 2018) is an improved version of YOLOv2. It uses multi-scale prediction to detect the final target, and its network structure is more complex than YOLOv2. YOLOv3 predicts bounding boxes on different scales, and multi-scale prediction makes YOLOv3 more effective for detecting small targets than YOLOv2. In this work, YOLOv3 with Darknet-53 was used for the task of strawberry flower detection.

SSD

The Single Shot Detector (SSD; Liu *et al*., 2016) is one of the primary endeavors at using convolutional neural network's pyramidal feature hierarchy for efficient detection of objects of various sizes. Like YOLO, SSD detection also happens in one stage, a deep convolutional network directly predicts object classes and anchor boxes without requiring a second stage per-proposal classification operation. SSD uses multi-scale convolutional bounding boxes and turn-outs are connected to multiple feature maps. Lower resolution layers are used to detect larger scale objects while higher resolution layers capture smaller scale objects. This study used SSD with resnet50 and feature pyramid network (FPN; Lin *et al.*, 2017) for the purpose of strawberry flower detection. The Tensorflow implementation of SSD by object detection API (Huang *et al.*, 2017) was adapted.

CenterNet

Most effective object detectors, such as the previously mentioned Faster R-CNN, SSD and YOLOv3, enumerate a nearly exhaustive amount of bounding box proposals and classify each of them. CenterNet (Duan *et al.*, 2019) introduces a different approach - it models an object as a single point - the center point of its bounding box. At first, a region proposal is obtained by a pair of corner points. Then, the network verifies whether the proposal is indeed an object by detecting if there's a center key point of the same class appearing in the central region of the proposal. In this study, CenterNet with an Hourglass-52 (Zhou *et al.*, 2019) backbone was used for strawberry flower detection task.

Evaluation of guided collage composition algorithm

To evaluate the efficiency of the guided collage composition

Table 1. Initialization parameters for the detection networks.

Detection networks	Size of input image	Batch size	Momentum optimizer value	Initial learning rate	Decay
Faster R-CNN	512 × 512	1	0.9	0.0003	Manual
SSD		12			0.8
YOLOv3	416 × 416	6		0.0001	0.96
CenterNet	512 × 512	5		0.001	Manual

algorithm, two different sets of synthetic images were generated. First set of synthetic images was generated using guided collage composition algorithm and second set was generated by positioning selected segmented strawberry flowers at random locations on top of background image which we call random collage composition.

Evaluation metrics

This study used the following indicators for evaluating the performance of convolutional object detection models for strawberry flower detection and the effectiveness of augmentation using synthetic images.

Precision, recall and average precision

The correctness of an identified strawberry flower object is determined by the intersection-over-union (IoU) overlap with the corresponding ground truth bounding box (Girshick *et al*., 2015). The IoU overlap is defined as follows:

$$IoU = \frac{Area(ground\ truth\ box\ \cap\ detected\ bounding\ box)}{Area(ground\ truth\ box\ \cup detected\ bunding\ box)}$$

A predicted bounding box is considered as true positive (*TP*) if its IoU overlap with a ground truth bounding box is greater than a certain threshold. Otherwise, the predicted bounding box is determined as false positive (*FP*). When the ground truth bounding box has no matches with the predicted bounding box, it is considered as false negative (*FN*). The default value of the IoU overlap threshold is 0.5 (Manning *et al*., 1999), which was used in this study. Based on these definitions, the precision and recall are calculated (Manning *et al*., 1999):

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

Based on the precision and recall score precision-recall curve is obtained by: (1) ordering all strawberry flower detections according to their confidence score, (2) matching detections to ground truth starting from highest confidence score until a recall $\tilde{r}$ higher than recall level r is reached, (3) calculating precision values based on each recall level r and (4) interpolating the precision p_{interp} by the maximum precision that can be achieved for a recall level *r* as defined by (Everingham *et al*., 2010) as:

$$p_{interp}(r) = \max_{\tilde{r}\,:\,\tilde{r}\ \geq r} \mathrm{p}(\tilde{r})$$

where $r \in$ {0, 0.1, 0.2, … , 1}.

The average precision (AP) was utilized to quantify the detection performances of different strawberry flower detection models. The standard average precision metrics, AP@).5IoU, is an overall measure of the performance of an object detector concerning a specific class of an object detection. According to Everingham *et al.* (2010), AP is calculated as the arithmetic mean of the precision-recall curve:

$$AP = \frac{1}{11} \sum_{r\, \in\{0.0,\ldots,1.0\}} p_{interp}\,(r)$$

Detection time

The average detection times for several convolutional object detection models were compared in this study, and the real-time performance of these models was analyzed.

RESULTS AND DISCUSSION

The detection networks were trained and tested on GPU (Nvidia GeForce GTX 1080 Ti) with a machine having Intel ® core i7-9700k 3.60 GHz processor and 64 GB RAM under the deep learning development framework of TensorFlow. The network initialization parameters are shown in Table 1.

Dataset description

A synthetic image dataset of 1800 images were generated using the method as described in synthetic

Table 2. Statistics of the entire dataset.

Data type	Empirical images	Synthesized images	Total
Training	100	1800	1900
Test	230		230

Table 3. AP scores of different strawberry flower detection models.

Models	AP	
	Empirical dataset (%)	After augmentation by synthetic images (%)
Faster R-CNN w/resnet50	75.13	90.84
SSD w/resnet50 and FPN	70.14	88.56
YOLOv3 w/darknet53	39.20	86.04
CenterNet w/hourglass52	61.58	83.82

image generation pipeline. This dataset was used to augment training dataset that contains 100 empirical strawberry flower images.

In this study, a series of experiments with the trained Faster R-CNN, SSD, YOLOv3 and CenterNet models were conducted with the test images to verify the effectiveness of the generated synthetic images. Test set contains 230 empirical images of strawberry flower with resolution 4608 × 3456 pixels. Table 2 shows the statistics of the entire dataset.

Influence of augmentation by synthetic images

In order to verify the effectiveness of the augmentation method proposed in this study, the Faster R-CNN w/resnet50, SSD w/resnet50 and FPN, YOLOv3 w/darknet53 and CenterNet w/hourglass52 neural networks were trained using a dataset that contains only empirical images and another expanded dataset that contains both empirical and synthetic images of strawberry flowers. The AP scores of the corresponding models are shown in Table 3.

Based on the detection results, augmenting the empirical set using synthetic images improved performance of all the models remarkably for the task of strawberry flower detection in non-structured agricultural environment. Empirical data on Faster R-CNN with resnet50 achieved AP 75.13%. When augmentation using synthetic images was applied, the AP increased to 90.84%. The AP of SSD with resnet50 and FPN and CenterNet with hourglass models improved by 18.42% and 22.24%, respectively. The YOLOv3 with darknet53 model achieved the most significant boost in performance, improving the AP by > 46% (from 39.20% to 86.04%). Figure 6 graphically visualizes the impact of the number of synthetic images on the strawberry flower detection performance of different models. From the results, one can draw the conclusion that the performance of the all four detection models improves as the number of synthetic images increases.

Efficiency of guided collage composition algorithm

The efficiency of guided collage composition algorithm was analyzed by comparing the performance of the data augmentation by synthesized images generated using guided collage composition with data augmentation by synthesized images generated using random collage composition. The AP score of the models trained using image set expanded by applying guided collage composition is higher (by >1.9%) than that of the models trained using image set expanded by applying random collage composition (Figure 7). Considering the results (Figure 7), guided collage composition simulates the real environment well and capable of generating more realistic synthetic images than random collage.

Comparison of different detection networks

In this section, the strawberry flower detection performances of Faster R-CNN, SSD, YOLOv3 and CenterNet are analyzed. The precision-recall curves and the average detection time (per image with resolution 4608 × 3456) of different strawberry flower detection models are shown in Figure 8 and in Table 4, respectively. It was observed that YOLOv3 w/darknet53 and SSD w/resnet50 and FPN models were faster, while Faster R-CNN w/resnet50 was slower but more accurate model. All the models provided high precision (>0.91) for strawberry flower detection, however, the recall was low (<0.85) for CenterNet with hourglass52.

With high precision (>0.92) and recall (>0.91), Faster R-CNN with resnet50 turned out to be the most accurate and robust model for strawberry flower detection in non-structured agricultural environment. The significant performance of Faster R-CNN reflects RPN's ability to produce good quality object proposals.

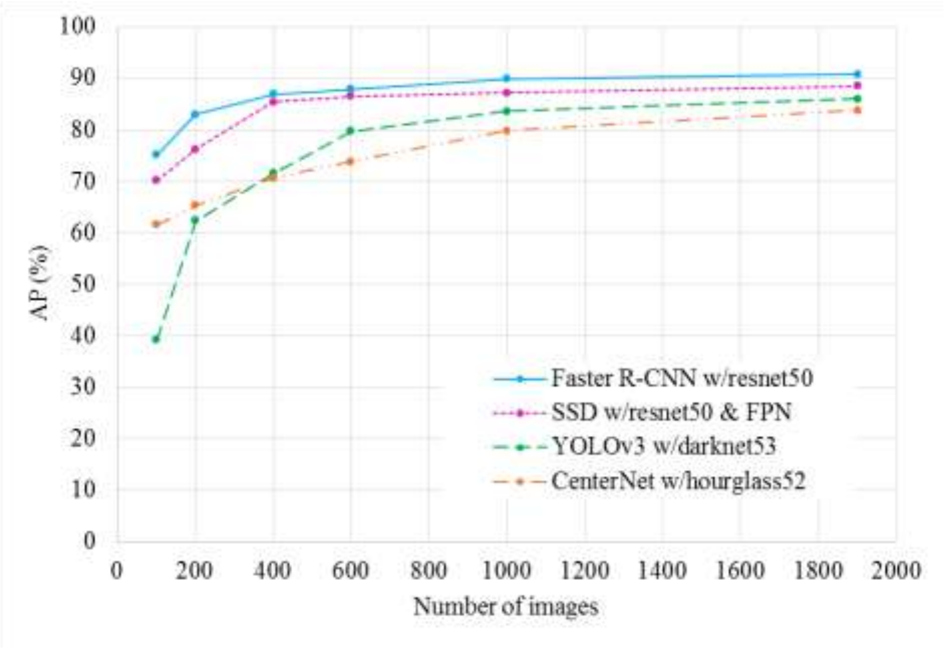
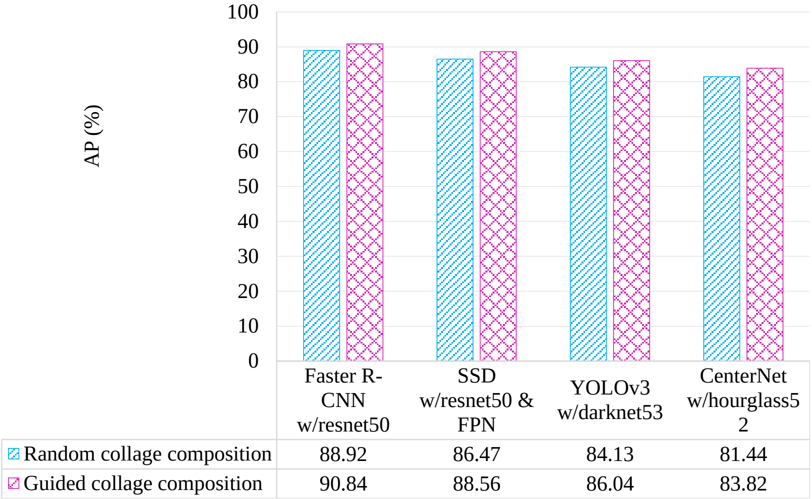


Figure 6. AP of different models with different number of synthetic images.



	Faster R-CNN w/resnet50	SSD w/resnet50 & FPN	YOLOv3 w/darknet53	CenterNet w/hourglass52
▨ Random collage composition	88.92	86.47	84.13	81.44
☒ Guided collage composition	90.84	88.56	86.04	83.82

Figure 7. Guided collage composition vs. Random collage composition.

The YOLOv3 w/darknet53 and SSD w/resnet50 and FPN provided faster detection (15.96 ms and 16.53 ms, respectively) compared to Faster R-CNN w/resnet50 (20.87ms) and CenterNet w/hourglass52 (18.25 ms) as they do target detection in one pass without requiring second stage per-proposal classification in Faster R-CNN and proposal refinement in CenterNet.

The SSD w/resnet50 and FPN model achieved best balance between detection performance and detection speed (Figure 9). Therefore, SSD w/resnet50 and FPN could be a suitable model for applications where real-time detection of flowers is required such as artificial pollination using a drone. On the other hand, Faster R-CNN w/resnet50 could be a suitable model for applications where accurate detection of all flowers in the scene is needed such as yield estimation.

Figure 10 visualizes some strawberry flower detections on images from our test set, showing side-by-side comparisons of four detection models. It can be observed that the Faster R-CNN w/resnet50 model detected almost all visible strawberry flowers in the images correctly and could address challenging issues, such as different illumination, overlapping, occlusions and very small targets (Figure 10a to d). The SSD w/resnet50 and FPN and YOLOv3 w/darknet53 models performed well under overlapped and occlusion conditions (Figure 10e, f and

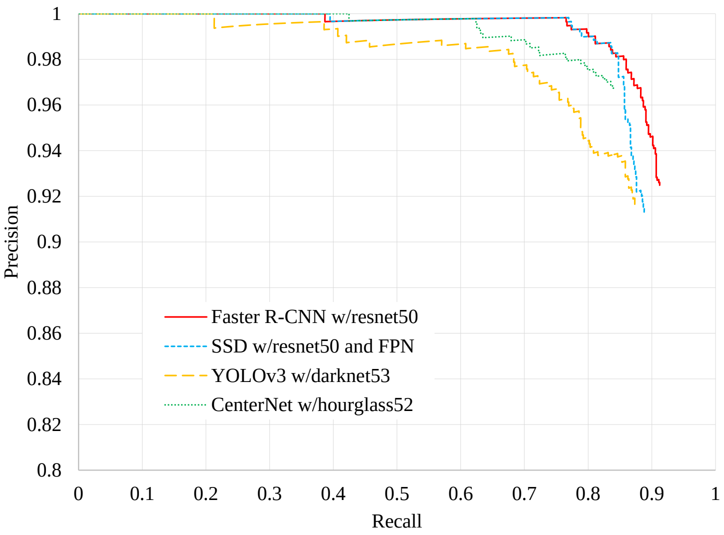


Figure 8. Precision-recall curve of several strawberry flower detection models.

Table 4. Average detection time of several strawberry flower detection models.

Models	Faster R-CNN w/resnet50	SSD w/resnet50 and FPN	YOLOv3 w/darknet53	CenterNet w/hourglass52
Average detection time (ms)	20.87	16.53	15.96	18.25

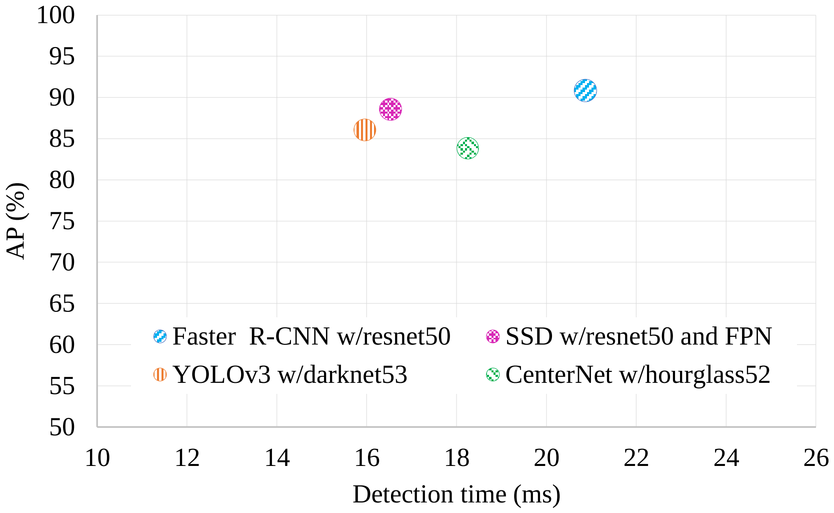


Figure 9. AP and average detection time of different strawberry flower detection models.

Figure 10i, j, respectively), however, SSD only showed its weakness in detecting very small flowers (Figure 10g, h) since SSD uses layers already deep down into the convolutional network to detect objects. The CenterNet w/hourglass52 model demonstrated poor performance under occlusions (Figure 10n).

CONCLUSION

In this study, synthetically generated images of strawberry flowers were applied to improving performance of strawberry flower detection using convolutional neural networks in non-structured agricultural environment. Experimental results

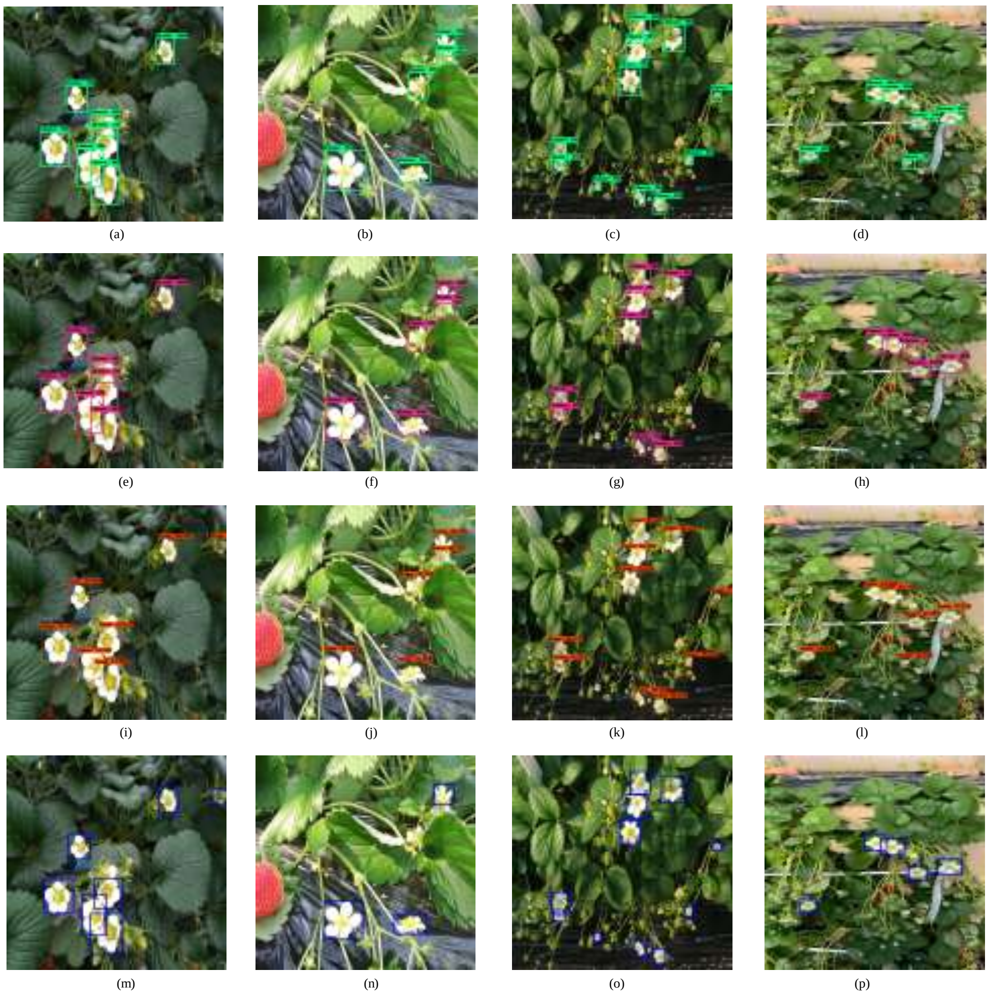


Figure 10. Examples of strawberry flower detections by several models. (a-d) detections by Faster R-CNN with reanet50, (e-h) detections by SSD with resnet50 and FPN, (i-l) detections by YOLOv3 with darknet53, (m-p) detections by CenterNet with hourglass52.

demonstrated that suggested data augmentation method enhances network performances remarkably indicating its' ability in generating large and diverse set of realistic synthetic images. This modest contribution will serve to motivate further examination of integrating synthetic data with real world botanical scenes for developing agricultural automation systems and different plant phenotyping tasks.

Our second contribution is the experimental comparison of performance of some modern convolutional object detectors. This will serve as a guideline for practitioners to select an appropriate method when extending object detection in various application of intelligent agriculture.

ACKNOWLEDGEMENT

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors thank Mr. Ohishi Naoki and Mr. Yahaguchi Motoki (Shizuoka Prefectural Agricultural and Forestry Research Institute) for providing an environment for data collection.

REFERENCES

Adamsen FJ, Coffelt TA, Nelson JM, Barnes EM, Rice RC (2000). Method for using images from a color digital camera to estimate flower number. Crop Sci. 40:704-709.

Aquino A, Millan B, Gutiérrez S, Tardáguila J (2015). Grapevine flower estimation by applying artificial vision techniques on images with uncontrolled scene and multi-model analysis. Comput. Electron. Agric. 119:92-104.

Bac CW, van Henten EJ, Hemming J, Edan Y (2014). Harvesting robots for high-value crops: State- of- the- art review and challenges ahead. J. F. Robot. 31:888-911.

Bairwa N, Agrawal NK (2014). Counting of flowers using image processing. Int. J. Eng. Res. Technol. 3:775-779.

Dcunha S, Das J, Qu C (2017). Counting Apples and Oranges with Deep Learning : https://doi.org/10.1109/LRA.2017.2651944.

De Brabandere B, Neven D, Van Gool L (2017). Semantic instance segmentation with a discriminative loss function. arXiv Prepr. arXiv1708.02551.

Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009). Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition. Ieee, pp. 248-255.

Dias PA, Tabb A, Medeiros H (2018). Apple flower detection using deep convolutional networks. Comput. Ind. 99, 17–28. https://doi.org/10.1016/j.compind.2018.03.010.

Douarre C, Schielein R, Frindel C, Gerth S, Rousseau D (2018). Transfer learning from synthetic data applied to soil–root segmentation in x-ray tomography images. J. Imaging 4, 65.

Duan K, Bai S, Xie L, Qi H, Huang Q, Tian Q (2019). Centernet: Keypoint triplets for object detection, in: Proceedings of the IEEE International Conference on Computer Vision. pp. 6569-6578.

Dyrmann M, Mortensen AK, Midtiby HS, Jørgensen RN (2016). Pixel-wise classification of weeds and crops in images by using a Fully Convolutional neural network. Int. Conf. Agric. Eng. p. 6.

Everingham M, Van Gool L, Williams CKI, Winn J, Zisserman A (2010). The pascal visual object classes (voc) challenge. Int. J. Comput. Vis. 88:303-338.

Girshick R, Donahue J, Darrell T, Malik J (2015). Region-based convolutional networks for accurate object detection and segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 38:142-158.

He K, Zhang X, Ren S, Sun J (2016). Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770-778.

Hočevar M, Širok B, Godeša T, Stopar M (2014). Flowering estimation in apple orchards by image analysis. Precis. Agric. 15:466-478. https://doi.org/10.1007/s11119-013-9341-6.

Huang J, Rathod V, Sun C, Zhu M, Korattikara A, Fathi A, Fischer I, Wojna Z, Song Y, Guadarrama S (2017). Speed/accuracy trade-offs for modern convolutional object detectors, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7310-7311.

Kamilaris A, Prenafeta-Boldú FX (2018). Deep learning in agriculture: A survey. Comput. Electron. Agric. 147, 70–90. https://doi.org/10.1016/j.compag.2018.02.016.

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. Int. J. Comput. Vis. Robot. 3:4-34.

Kaur R, Porwal S (2015). An optimized computer vision approach to precise well-bloomed flower yielding prediction using image segmentation. Int. J. Comput. Appl. p. 119.

Krizhevsky A, Sutskever I, Hinton GE (2012). Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems. pp. 1097-1105.

LeCun Y, Bengio Y, Hinton G (2015). Deep learning. Nature. 521:436-444.

Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2017). Feature pyramid networks for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2117-2125.

Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, Berg AC (2016). Ssd: Single shot multibox detector, in: European Conference on Computer Vision. Springer, pp. 21-37.

Mannin CD, Manning CD, Schütze H (1999). Foundations of statistical natural language processing. MIT press.

Namin ST, Esmaeilzadeh M, Najafi M, Brown TB, Borevitz JO (2018). Deep phenotyping : deep learning for temporal phenotype / genotype classification. Plant Methods. pp. 1-14. https://doi.org/10.1186/s13007-018-0333-4.

Oppenheim D, Edan Y, Shani G (2017). Detecting Tomato Flowers in Greenhouses Using Computer Vision. Int. J. Comput. Electr. Autom. Control Inf. Eng. 11:104-109.

Plebe A, Grasso G (2001). Localization of spherical fruits for robotic harvesting. Mach. Vis. Appl. 13:70-79.

Rahim UF, Mineno H (in press). Tomato Flower Detection and Counting in Greenhouses Using Faster Region-based Convolutional Neural Network. Journal of Image and Graphics.

Redmon J, Divvala S, Girshick R, Farhadi A (2016). You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 779-788.

Redmon J, Farhadi A (2018). YOLOv3: An Incremental Improvement.

Redmon J, Farhadi A (2017). YOLO9000: Better, faster, stronger. Proc. 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017 2017-Janua, pp. 6517-6525. https://doi.org/10.1109/CVPR.2017.690.

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.

Simonyan K, Zisserman A (2014). Very deep convolutional networks for large-scale image recognition. arXiv Prepr. arXiv1409.1556.

Thorp KR, Dierig DA (2011). Color image segmentation approach to monitor flowering in lesquerella. Ind. Crops Prod. 34:1150-1159. https://doi.org/10.1016/j.indcrop.2011.04.002.

Tyagi AC (2016). Towards a second green revolution. Irrig. Drain. 65:388-389.

Ubbens J, Cieslak M, Prusinkiewicz P, Stavness I (2018). The use of plant models in deep learning: An application to leaf counting in rosette plants. Plant Methods 14:1-10. https://doi.org/10.1186/s13007-018-0273-z.

Ward D, Moghadam P, Hudson N (2018). Deep leaf segmentation using synthetic data. arXiv Prepr. arXiv1807.10931.

Woebbecke DM, Meyer GE, Von Bargen K, Mortensen DA (1995). Color indices for weed identification under various soil, residue, and lighting conditions. Trans. ASAE 38:259-269.

Xu R, Li C, Paterson AH, Jiang Y, Sun S, Robertson JS (2018). Aerial Images and Convolutional Neural Network for Cotton Bloom Detection. Front. Plant Sci. 8:1-17. https://doi.org/10.3389/fpls.2017.02235.

Zhao Y, Gong L, Huang Y, Liu C (2016). A review of key techniques of vision-based control for harvesting robot. Comput. Electron. Agric. 127:311-323.

Zhou X, Wang D, Krähenbühl P (2019). Objects as points. arXiv Prepr. arXiv1904.07850.