# Semi-Supervised Object Detection for Agriculture

**Gabriel Tseng[1*], Krisztina Sinkovics[2], Tom Watsham[2], David Rolnick[1], Thomas Walters[2]**

[1]McGill University and Mila – Quebec AI Institute
[2]Small Robot Company

## Abstract

Machine learning is a key component of precision agriculture, by allowing plant-level insights to be inferred at scale. However, the labelled data necessary to train these algorithms is expensive to acquire, making methods that leverage unlabelled data – such as semi-supervised object detection (SSOD) – of particular interest. Current SSOD methods have been designed specifically for Flickr-based datasets and may not be appropriate for the unique challenges encountered in agricultural contexts, limiting their usefulness in practice. In this paper, we offer innovations in designing, testing, and deploying SSOD in the real world. We compile a challenging new dataset of semi-supervised agricultural images, on which existing SSOD methods largely fail. We present a Python package to more easily test SSOD methods in such real-world domains. Finally, we introduce two components to the standard SSOD pipeline which demonstrably improve performance on our dataset. All code is available at `https://github.com/SmallRobotCompany`.

## Introduction

Precision agriculture is an important tool for sustainability as well as productivity. It can reduce farmer inputs, such as nitrogen fertilizer (Waldrop et al. 2004; Plant et al. 2000), and allow soil to better operate as a carbon stock reserve by reducing tillage (Angers and Eriksen-Hamel 2008). While machine learning plays a critical role in precision agriculture (Sharma et al. 2020), labelled data can be difficult and expensive to acquire, making semi-supervised learning techniques particularly relevant to this domain.

Semi-Supervised Object-Detection consists of learning to identify objects from a training set consisting of labelled and unlabelled images. SSOD research is typically evaluated against the COCO (Lin et al. 2014a) and Pascal VOC (Everingham et al. 2010) datasets where only a randomly sampled subset of the images have labels provided (Sohn et al. 2020).

While the COCO and Pascal VOC datasets both use images from Flickr and are similar in many respects, real-world agricultural datasets often present a very different set of challenges and attributes. These include: i) relatively few classes (much less than Flickr-based datasets), with many agricultural datasets consisting of a single class (Assunção et al. 2022; Bargoti and Underwood 2017; Roy and Bhaduri 2022; David et al. 2021), ii) a restricted geographic area, with some datasets covering a single field (Assunção et al. 2022) or a small number of them (Bargoti and Underwood 2017), iii) unique challenges, including high levels of occlusion (Lawal 2021) intra-class variance (Roy and Bhaduri 2022) and poor image quality, and iv) a higher object density; for example, the Global Wheat Head Dataset (David et al. 2021) has 39.1 objects per image on average, compared to 7.3 objects per image in COCO (Zhou et al. 2021b).

The small size of labelled object detection datasets in agricultural contexts make SSOD methods tailored for agricultural contexts especially important to investigate. The main contributions of this paper are:

- The introduction of the `smallSSD` dataset, a semi-supervised object detection dataset for agriculture consisting of over 100,000 images.
- Calibrated Teacher-Student Learning, a semi-supervised object detection method tailored for agriculture.
- Python packages for the `smallSSD` dataset and for real-world SSOD research more broadly, moving beyond Flickr-based datasets.

## Related Work

**Semi Supervised Object Detection** Pseudo-labelling methods – where a trained model is used to generate labels for otherwise unlabelled images – are commonly used in semi-supervised object detection (Sohn et al. 2020; Liu et al. 2021; Xu et al. 2021). Research in this area heavily builds on the pseudo-labelling framework introduced by (Sohn et al. 2020); subsequent work has investigated how the teacher-model may be improved (Liu et al. 2021; Zhou et al. 2021a), how the pseudo-labels can be better selected before being passed to the student (Li et al. 2021; Xu et al. 2021) and how alternative architectures may perform (Wang et al. 2022). However, we note again that these methods are evaluated only on Flickr-based datasets, and do not consider the nuances of agricultural contexts.

**Object Detection in Agriculture** Current approaches to object detection in agricultural contexts (Sa et al. 2016; Bargoti and Underwood 2017) focus on fully-supervised learning, specifically by finetuning models pretrained on

[*]Work done while at the Small Robot Company

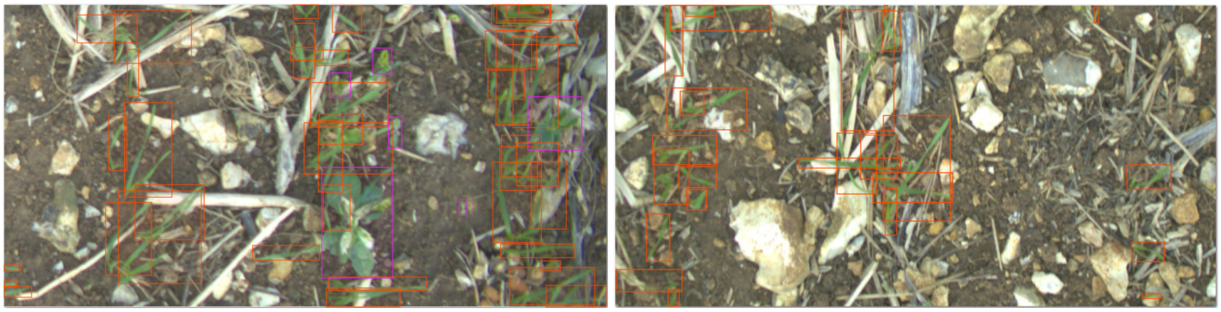

Figure 1: Sample annotated images. Orange boxes represent wheat & purple boxes represent weeds.

ImageNet (Deng et al. 2009a). A drawback of this fully-supervised approach is that – particularly in the case of grain crops, which we are considering – annotation is challenging even when the annotators have received extensive training, and (as a result) obtaining high quality annotations is difficult and time consuming (Patrício and Rieder 2018).

Focusing on the task of object-detection of grains, the largest scale public dataset – the Global Wheat Detection dataset – consists of fewer than 5,000 labelled images. For comparison, this is less than 3% the size of the COCO object detection dataset, which consists of over 200,000 labelled images (Lin et al. 2014a). Therefore, the creation of large scale open source agricultural semi-supervised object detection datasets – and the development of methods on top of such datasets – is critical.

## The `smallSSD` dataset

We introduce the `smallSSD` dataset[1] – a semi-supervised object detection dataset for agriculture. This dataset consists of images taken from above, of which a subset have been labelled with bounding boxes, collected by the Small Robot Company's "Tom" autonomous field survey robot during winter wheat surveys on a farm in South of England. The images were taken in 8 trial plots in a field with varying drill rates and fertilizer and herbicide application, thus capturing a diversity of field conditions. All images are RGB and are `1200 × 600` pixels.

Annotation was conducted by a team of annotators trained by agronomists to recognize and mark up individual crops and weeds across different growth stages of plants. The annotation process includes a regular feedback loop from a field specialist. The quality of labels has been validated by the agronomists. Figure 1 shows examples of images with "wheat" and "weed" annotations.

This dataset is split into three parts: i) a labelled training set, ii) a labelled test set and iii) an unlabelled training set. The unlabelled dataset consists of **100,032** unlabelled images. The labelled dataset consists of **960** images, with the crops and the weeds in the images labelled using bounding boxes. We split this labelled dataset into a training set consisting of **804** (**84%**) images, and an evaluation set consisting of **156** (**16%**) images.

[1]Released under Creative Commons Attribution-NonCommercial 4.0 International license

## Calibrated Teacher-Student Learning

In keeping with standard practice in SSOD, we consider models trained with a student-teacher training scheme. We begin by training a model in a fully supervised fashion, after (Liu et al. 2021) ("burn-in"). We then use the weights learned in a fully-supervised fashion to initialize a teacher and student model. We use the predictions of the teacher model to supplement the real labels in the dataset, and jointly train the student model on these pseudo-labels and on the true labels. The teacher is updated using exponential mean averaging (EMA), as in (Liu et al. 2021; Xu et al. 2021).

We now highlight some of the key aspects of our method and how they differ from other common ingredients in teacher-student methods for object detection:

**Augmentations**: We find that certain augmentations – which are otherwise commonly used in semi-supervised object detection – can be extremely detrimental in agricultural contexts. Motivated by (Balestriero, Bottou, and LeCun 2022), we hypothesize that properties of agricultural datasets (including `smallSSD`) – such as the homogeneous colour of positive classes (green wheat leaves and green broad leaf weeds) and the very high resolution of the images mean that colour and scale-altering augmentations penalize the model. We therefore only apply two sets of augmentations to the model: i) mosaicing (Bochkovskiy, Wang, and Liao 2020), which consists of combining 4 images into a single one, and ii) random horizontal and vertical flips.

**Weak teacher-ensembling** Although previous work has proposed ensembling teacher models to improve teacher predictions (Zhou et al. 2021a), this has required training multiple student models. We propose using test-time augmentations to create ensembles of predictions with a single teacher-student model. Specifically, we use 3 augmentations (in addition to no augmentation) when generating the pseudo-labels for each unlabelled image: i) a horizontal flip, ii) a vertical flip, iii) both a vertical and horizontal flip.

We combine the predictions for each augmentation using Weighted Boxes Fusion (WBF) (Solovyev, Wang, and Gabruseva 2021), yielding the pseudo-labels ultimately used to train the student model. Specifically, for each box pre-

| Model | Backbone | Fully Supervised mAP | Fully Supervised $\text{mAP}_{50}$ | Fully Supervised $\text{mAP}_{75}$ | Semi-Supervised mAP | Semi-Supervised $\text{mAP}_{50}$ | Semi-Supervised $\text{mAP}_{75}$ |
|---|---|---|---|---|---|---|---|
| SSD | VGG-16 | 0.133 | 0.383 | 0.062 | 0.151 | 0.420 | 0.075 |
| RetinaNet | ResNet-50 | 0.295 | 0.630 | 0.239 | 0.302 | 0.630 | 0.253 |
| YOLOv4 | CSPDarknet53 | 0.322 | 0.645 | 0.296 | 0.347 | 0.649 | 0.335 |
| FasterRCNN | ResNet-50 | 0.355 | 0.689 | 0.334 | 0.365 | 0.697 | 0.347 |

Table 1: Baseline results for fully-supervised and semi-supervised benchmarks.

dicted by a single model in the ensemble, WBF iterates through boxes predicted by other models and finds those with an IOU above a threshold (in our case, 0.7). The ultimate model predictions are an average of each of those boxes, weighted by the model's confidence. This is similar to the co-rectify method described in (Zhou et al. 2021a), but does not require index-alignment in the proposals (allowing for geometric augmentations).

**Per-class calibration**: In teacher-student setups, a critical hyperparameter is the threshold $\tau$ used to decide which of the teacher's predictions will be used to train the student. Previous works use a fixed threshold (Liu et al. 2021; Sohn et al. 2020). However, in an agricultural context, the calibration of the model is critical; specifically, the ability of the model to output a correct average plant count over the entire field (or over sub-regions of the field). We therefore dynamically select this parameter $\tau$ on a per-class basis, so that the average number of class-predictions by the teacher reflect the distribution of labels in the training data.

For each class $c_i \in C$, we compute the average number of objects with the label $c_i$ in the labelled training set, $n_{c_i}$. We then select a class specific threshold $\tau_{c_i}$ such that the number of objects detected by the ensembled teacher-models, $\hat{n}_{c_i} = f(\tau_{c_i})$, for a sample of unlabelled images is close to $n_{c_i}$:

$$\tau_{c_i} = \arg\min_{\tau_{c_i}} |\lambda n_{c_i} - f(\tau_{c_i})|, \quad (1)$$

for some scaling factor $\lambda$. In our experiments, we set $\lambda = \frac{4}{3}$. We use an unlabelled random sample with the same size as the labelled trainig set when calculating $\tau_{c_i}$, and recalculate this threshold every epoch. We use $\tau_{c_i}$ to filter boxes during the WBF ensembling described above.

# Experimental Results

## Baselines

As baselines, we evaluate a variety of model architectures, encompassing both single-stage and multi-stage object detectors (described in Table 1). For all models except YOLOv4 (Bochkovskiy, Wang, and Liao 2020) we fine-tune models pre-trained on ImageNet (Deng et al. 2009b) and implemented in the `torchvision` framework. For YOLOv4 we used a CSPDarknet53 (Wang et al. 2020) backbone pre-trained on COCO(Lin et al. 2014b).

We evaluate all models by measuring the Mean Average Precicision (mAP) (Everingham and Winn 2011) on the evaluation set. Specifically, we consider mAP with an IOU threshold of $0.5$ ($\text{mAP}_{50}$), $0.75$ ($\text{mAP}_{75}$) and using a stepped range $[0.50, ..., 0.95]$ with intervals of size $0.05$.

**Fully Supervised Models** For fully supervised training, we ignored all unlabelled data (and only used the labelled training set). We split the training dataset into a training and validation set (using an 80:20 ratio), and used the validation set for early stopping with a patience of 10 (monitored against validation mAP). We used an Adam optimizer (Kingma and Ba 2015) and reduced the learning rate by a factor of 10 when the validation mAP did not decrease for 2 epochs. Models were trained for between 21 (YOLOv4) and 64 epochs (SSD). For this baseline, we apply the random flip augmentation but not the mosaicing. The results of the fully-supervised models are shown in Table 1.

**Semi-Supervised Pseudo-Labelling** To measure the change in performance when the unlabelled data is considered, we additionally implement a pseudo-labelling (Lee et al. 2013) baseline. This pseudo-labelling baseline consists of taking a trained-fully supervised model (trained as described above), and using it to generate predictions on 2000 randomly sampled unlabelled images. This baseline is similar to the approach introduced in (Sohn et al. 2020).

We then added these 2000 images to the labelled dataset, and finetuned the model on the aggregate dataset. The results of these pseudo-labelling experiments are also shared in Table 1.

## Calibrated Teacher-Student Learning

We evaluate our proposed algorithm by testing whether it improves the best-performing baseline model – Faster R-CNN. We train the fully supervised model in the same manner as the baselines described above, with the addition of the mosaicing augmentation.

Our results – comparing the proposed method to the baselines – are presented in Table 2. Overall, this method outperforms the baselines across all three metrics (mAP, $\text{mAP}_{50}$ and $\text{mAP}_{75}$), with a particular increase in $\text{mAP}_{75}$.

**Ablations** To better understand the contributions of different parts of the algorithm, we conduct an ablation study in Table 3. Specifically, we run the algorithm without the calibrated threshold ("T") and without an ensembled teacher ("E"), reporting the % change between the fully and semi-supervised methods without these components. When we do not calibrate the threshold, we use a fixed threshold of $0.7$ as in Liu et al. (2021). We find that while each component in isolation yields an improvement over the baseline it is the combination of both components which leads to the overall improvement.

| | | mAP | | $\text{mAP}_{50}$ | | $\text{mAP}_{75}$ | |
|---|---|---|---|---|---|---|---|
| Baseline | Full | $34.587 \pm 0.004$ | +4.54% | $68.287 \pm 0.006$ | +2.10% | $33.520 \pm 0.004$ | +5.92% |
| | Semi | $36.157 \pm 0.002$ | | $69.723 \pm 0.002$ | | $33.523 \pm 0.003$ | |
| Calibrated Teacher Student | Full | $35.180 \pm 0.001$ | +5.27% | $67.367 \pm 0.013$ | +3.88% | $31.983 \pm 0.003$ | +10.05% |
| | Semi | $\mathbf{37.033} \pm 0.001$ | | $\mathbf{69.983} \pm 0.000$ | | $\mathbf{35.197} \pm 0.001$ | |

Table 2: Results for the fully supervised and semi-supervised Faster R-CNN models. These results are averaged from 3 runs with different random seeds, with standard error reported and the best results **highlighted**.

| T | E | mAP | $\text{mAP}_{50}$ | $\text{mAP}_{75}$ |
|---|---|---|---|---|
| ✓ | | $4.61 \pm 0.13$ | $1.89 \pm 0.24$ | $6.82 \pm 0.23$ |
| | ✓ | $4.87 \pm 0.14$ | $-0.43 \pm 0.44$ | $11.12 \pm 0.22$ |
| ✓ | ✓ | $5.27 \pm 0.04$ | $3.88 \pm 1.31$ | $10.04 \pm 0.25$ |

Table 3: Ablations for the calibrated teacher-student method. This table reports % increases in the semi-supervised model over the fully supervised model. Results are an average of 3 runs with standard error reported.

## Accessibility of the method and data

A significant goal of this work is to encourage semi-supervised object detection research beyond Flickr-based datasets. To this end, we release two python packages alongside this work, one to encourage exploration and utilization of the dataset and one for broader semi-supervised object detection research.

### The `smallSSD` python package

The pip-installabe `smallSSD` python package exposes the `smallSSD` dataset as a `Dataset` object inspired by the `torchvision` datasets, allowing a straightforward integration of the `smallSSD` data into the PyTorch (Paszke et al. 2019) computer-vision ecosystem:

```
from torch.utils.data import DataLoader
from smallssd.data import LabelledData
from smallssd.data import UnlabelledData

loader = DataLoader(LabelledData())
u_loader = DataLoader(UnlabelledData())
```

These datasets return images as PyTorch tensors and (in the labelled case) annotations as dictionaries of bounding boxes and labels (as expected by PyTorch object detection models). Returning datasets in a `torchvision` compatible format means all `torchvision` tools (e.g. for visualization) can be leveraged. Finally, **the dataset automatically downloads the data if it isn't already present**.

### A python package for SSOD research

We additionally introduce `smallteacher`, a pip installable python package built on top of PyTorch-Lightning (Falcon et al. 2020) to encourage easily-adoptable research in real-world semi-supervised object detection. In particular, this code-base aims to decouple semi-supervised object detection code from the COCO and Pascal VOC datasets.

Unlike other codebases which require data to be stored in a certain format, this codebase only requires torchvision-like object-detection `Dataset` objects to be provided by the user. Given this dataset object, three main classes are exposed by the codebase:

**A `DataModule`** which upon initialization accepts labelled and unlabelled datasets. This data module checks that the provided datasets have the right outputs.

```
dm = DataModule(
    train_ds, val_ds, test_ds,
    unlabelled_train_ds
)
```

**A Fully Supervised Object Detection pipeline** which can be used to train a variety of model architectures (currently, Faster-RCNN, SSD, YOLO and RetinaNet) to identify objects from a dataset:

```
model = FullySupervised(
    model_base="FRCNN", num_classes=2
)
```

**A Teacher Student pipeline** which implements the best practices described above (as well as additional options motivated by prior research):

```
model = SemiSupervised(
    model_base="FRCNN", num_classes=2
)
```

Where both models can be fit using a PyTorch-Lightning Trainer:

```
trainer.fit(model, dm)
```

This codebase is highly configurable, allowing different practices to be quickly tested against datasets. Our hope is that the simplicity of this library – which allows a teacher-student model to be trained on any dataset in only a few lines of code – will allow for research into student-teacher methods for object detection beyond Flickr datasets.

## Conclusion

In conclusion, we present a novel agricultural semi-supervised object detection dataset, `smallSSD`, and a semi-supervised object detection algorithm tailored to this agricultural use case. We additionally present two python packages, one to explore the dataset and one for general semi-supervised object detection research, in the hope that future semi-supervised object detection research will look beyond the currently investigated Flickr-based datasets.

## References


Angers, D.; and Eriksen-Hamel, N. 2008. Full-inversion tillage and organic carbon distribution in soil profiles: A meta-analysis. *Soil Science Society of America Journal*.

Assunção, E. T.; Gaspar, P. D.; Mesquita, R. J. M.; Simões, M. P.; Ramos, A.; Proença, H.; and Inacio, P. R. M. 2022. Peaches Detection Using a Deep Learning Technique–A Contribution to Yield Estimation, Resources Management, and Circular Economy. *MDPI Climate*.

Balestriero, R.; Bottou, L.; and LeCun, Y. 2022. The effects of regularization and data augmentation are class dependent. *arXiv preprint arXiv:2204.03632*.

Bargoti, S.; and Underwood, J. 2017. Deep fruit detection in orchards. In *IEEE international conference on robotics and automation (ICRA)*. IEEE.

Bochkovskiy, A.; Wang, C.-Y.; and Liao, H.-Y. M. 2020. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

David, E.; Serouart, M.; Smith, D.; Madec, S.; Velumani, K.; Liu, S.; Wang, X.; Pinto, F.; Shafiee, S.; Tahir, I. S.; et al. 2021. Global wheat head detection 2021: an improved dataset for benchmarking wheat head detection methods. *Plant Phenomics*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009a. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition (CVPR)*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009b. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

Everingham, M.; Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vision*.

Everingham, M.; and Winn, J. 2011. The pascal visual object classes challenge 2012 (voc2012) development kit. *Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep*.

Falcon, W.; Borovec, J.; Wälchli, A.; Eggert, N.; Schock, J.; Jordan, J.; Skafte, N.; Ir1dXD; Bereznyuk, V.; Harris, E.; Murrell, T.; Yu, P.; Præsius, S.; Addair, T.; Zhong, J.; Lipin, D.; Uchida, S.; Bapat, S.; Schröter, H.; Dayma, B.; Karnachev, A.; Kulkarni, A.; Komatsu, S.; Martin.B; SCHIRATTI, J.-B.; Mary, H.; Byrne, D.; Eyzaguirre, C.; cinjon; and Bakhtin, A. 2020. PyTorchLightning/pytorch-lightning.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

Lawal, M. O. 2021. Tomato detection based on modified YOLOv3 framework. *Scientific Reports*.

Lee, D.-H.; et al. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, 896.

Li, H.; Wu, Z.; Shrivastava, A.; and Davis, L. S. 2021. Rethinking pseudo labels for semi-supervised object detection. In *AAAI Conference on Artificial Intelligence*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014a. Microsoft coco: Common objects in context. In *European conference on computer vision (ECCV)*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014b. Microsoft COCO: Common Objects in Context. In Fleet, D.; Pajdla, T.; Schiele, B.; and Tuytelaars, T., eds., *Computer Vision – ECCV 2014*, 740–755. Cham: Springer International Publishing.

Liu, Y.-C.; Ma, C.-Y.; He, Z.; Kuo, C.-W.; Chen, K.; Zhang, P.; Wu, B.; Kira, Z.; and Vajda, P. 2021. Unbiased Teacher for Semi-Supervised Object Detection. In *International Conference on Learning Representations (ICLR)*.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*.

Patrício, D. I.; and Rieder, R. 2018. Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Computers and electronics in agriculture*.

Plant, R.; Pettygrove, G.; Reinert, W.; et al. 2000. Precision agriculture can increase profits and limit environmental impacts. *California Agriculture*.

Roy, A. M.; and Bhaduri, J. 2022. Real-time growth stage detection model for high degree of occultation using DenseNet-fused YOLOv4. *Computers and Electronics in Agriculture*.

Sa, I.; Ge, Z.; Dayoub, F.; Upcroft, B.; Perez, T.; and McCool, C. 2016. Deepfruits: A fruit detection system using deep neural networks. *sensors*, 16(8): 1222.

Sharma, A.; Jain, A.; Gupta, P.; and Chowdary, V. 2020. Machine learning applications for precision agriculture: A comprehensive review. *IEEE Access*.

Sohn, K.; Zhang, Z.; Li, C.-L.; Zhang, H.; Lee, C.-Y.; and Pfister, T. 2020. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*.

Solovyev, R.; Wang, W.; and Gabruseva, T. 2021. Weighted boxes fusion: Ensembling boxes from different object detection models. *Image and Vision Computing*.

Waldrop, M. P.; Zak, D. R.; Sinsabaugh, R. L.; Gallo, M.; and Lauber, C. 2004. Nitrogen deposition modifies soil carbon storage through changes in microbial enzymatic activity. *Ecological applications*.

Wang, C.-Y.; Liao, H.-Y. M.; Wu, Y.-H.; Chen, P.-Y.; Hsieh, J.-W.; and Yeh, I.-H. 2020. CSPNet: A new backbone that can enhance learning capability of CNN. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR) workshops*.

Wang, P.; Cai, Z.; Yang, H.; Swaminathan, G.; Vasconcelos, N.; Schiele, B.; and Soatto, S. 2022. Omni-DETR: Omni-Supervised Object Detection with Transformers. In *Pro-*

*ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Xu, M.; Zhang, Z.; Hu, H.; Wang, J.; Wang, L.; Wei, F.; Bai, X.; and Liu, Z. 2021. End-to-end semi-supervised object detection with soft teacher. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Zhou, Q.; Yu, C.; Wang, Z.; Qian, Q.; and Li, H. 2021a. Instant-teaching: An end-to-end semi-supervised object detection framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Zhou, Y.-C.; Hu, Z.-Z.; Yan, K.-X.; and Lin, J.-R. 2021b. Deep learning-based instance segmentation for indoor fire load recognition. *IEEE Access*.