

# On the Real-Time Semantic Segmentation of Aphid Clusters in the Wild

Raiyan Rahman<sup>1</sup>, Christopher Indris<sup>1</sup>, Tianxiao Zhang<sup>2</sup>  
Kaidong Li<sup>2</sup>, Brian McCornack<sup>3</sup>, Daniel Flippo<sup>4</sup>, Ajay Sharda<sup>4</sup>, Guanghui Wang<sup>1\*</sup>

<sup>1</sup>Department of Computer Science, Toronto Metropolitan University, Toronto, Canada

<sup>2</sup>Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, USA

<sup>3</sup>Department of Entomology, Kansas State University, Manhattan, USA

<sup>4</sup>Department of Biological and Agricultural Engineering, Kansas State University, Manhattan, USA

## Abstract

*Aphid infestations can cause extensive damage to wheat and sorghum fields and spread plant viruses, resulting in significant yield losses in agriculture. To address this issue, farmers often rely on chemical pesticides, which are inefficiently applied over large areas of fields. As a result, a considerable amount of pesticide is wasted on areas without pests, while inadequate amounts are applied to areas with severe infestations. The paper focuses on the urgent need for an intelligent autonomous system that can locate and spray infestations within complex crop canopies, reducing pesticide use and environmental impact. We have collected and labeled a large aphid image dataset in the field, and propose the use of real-time semantic segmentation models to segment clusters of aphids. A multiscale dataset is generated to allow for learning the clusters at different scales. We compare the segmentation speeds and accuracy of four state-of-the-art real-time semantic segmentation models on the aphid cluster dataset, benchmarking them against nonreal-time models. The study results show the effectiveness of a real-time solution, which can reduce inefficient pesticide use and increase crop yields, paving the way towards an autonomous pest detection system.*

## 1. Introduction

The challenge of meeting the growing global demand for food under various environmental conditions has become increasingly significant. It is estimated that 37% of crops worldwide are lost to pest damage, with 13% attributed to insects. This loss includes staple crops such as rice, wheat, and maize, which directly impacts global food security. Additionally, commodity crops like banana and coffee have a significant impact on national economies, as shown in [5].



Figure 1. Examples of dataset images and their corresponding ground truth masks. The first, second, and third columns represent image patches at Scale-1, Scale-2, and Scale-3, respectively. The top row shows the aphid clusters while the bottom row shows their corresponding ground truth masks overlaid on top of them.

The increasing demand for food has led to the widespread use of chemical pesticides, which have become a critical component of crop production for maximizing yields. As a result, the global pesticide market was valued at approximately \$53.7 billion in 2015, and the economic value of global pesticide imports has been increasing by 7% annually since 2000 [15]. Farmers typically apply uniform and continuous pesticide treatment across entire fields once a pest infestation reaches a specific treatment threshold. However, this approach is excessive as pest incidence and severity are only present in a fraction of the field. This highlights the urgent need for an intelligent autonomous system that can accurately locate and spray infestations within complex crop canopies, minimizing pesticide use and reducing environmental impact.

The topic of automatic pest detection has been extensively studied, with significant challenges arising due to the small size of pests and their ability to blend into their natural settings. Previous studies have focused on counting individual pests [1, 2, 16], while not putting as much focus on the pressing issue of identifying infestations. In recent

\*Corresponding author: Guanghui Wang (wangcs@torontomu.ca)

years, through the development of small autonomous vehicles such as drones, researchers have been able to study the viability of autonomous pest-scouting vehicles with camera feeds for the eventual goal of exploring crop canopies and identifying pest infestations. While work has been done on the detection and segmentation of aphids [2, 9, 16, 21], none have focused on this real-time aspect which is crucial for an autonomous system. With the use of real-time semantic segmentation models, we explore the feasibility of accurate detection of pests in real-time as these vehicles with their feeds move throughout the fields.

In this paper, we focus on segmenting clusters of aphids in real-time using semantic segmentation models. We use high-resolution images captured from a sorghum field to create a multiscale dataset that allows for learning the clusters at different scales. We train our real-time models and compare their performance against state-of-the-art nonreal-time models, evaluating them across a wide range of metrics and segmentation speeds. To verify the effectiveness of different scales, we also train our models on each individual scale and compare their performance to the multiscale version of the dataset. Our goal is to identify the best model for automated pest infestation segmentation by exploring the relationship between accuracy and speed. Based on our results, Fast-SCNN and Small HRNet are two promising models that achieve high accuracy while also providing real-time segmentation speeds. The generated dataset can be accessed at: <https://doi.org/10.7910/DVN/N3YJXG>.

## 2. Related Works

Computer vision techniques have been extensively used in the research of automated pest detection. Barbedo [1] demonstrated an automatic method of detecting and counting pests on leaves with simple backgrounds. It was suggested that designing more effective features would further improve segmentation performance. A more robust aphid identification model was developed by [9] using histogram-oriented gradient features and a support vector machine. They utilized a maximally stable extremal region descriptor to allow the model to focus on pest regions rather than backgrounds. However, these methods required carefully designed features to describe and detect the targets. Thus, there was a movement towards designing models that could learn effective features from the data itself.

Convolutional neural networks (CNNs) are one of the most effective feature extraction and classification methods, achieving numerous state-of-the-art results in various image analysis tasks, such as classification [3, 10], object detection [8], segmentation [6, 12], object counting [14], and tracking [22]. However, the detection of small insects can be especially challenging and even the state-of-the-art models have difficulty detecting individual aphids to satisfactory results. A modified U-Net architecture was proposed in [2]

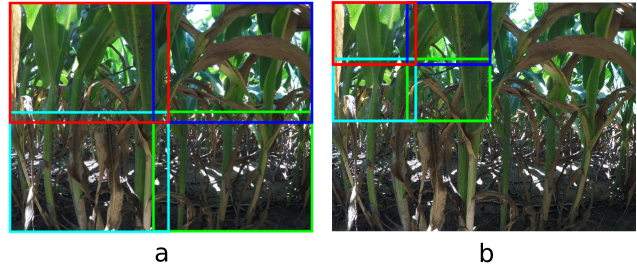


Figure 2. Example of two of the different scales that were used in training. Image (a) shows the original high-resolution image of resolution  $3647 \times 2736$  with patches at the  $0.525W \times 0.525H$  scale, where  $W$  and  $H$  refer to the width and height of the original image. At this scale, the original image will yield 4 patches. Image (b) shows how the patches were generated at  $0.263W \times 0.263H$  relative to the original image. This scale will yield 16 patches from the original image. In each case, adjacent patches were taken with an overlap of 10%.

to segment and count aphid nymphs on leaves. Although the model achieved high precision and recall and was generic to different species of pests, it was trained on ideal images with black backgrounds rather than the natural field settings in which such a detection system would be necessary. In these real-world settings, most aphids are clustered together on leaves as well as partially covered by adjacent foliage, making their individual detection amidst the dense clusters very difficult.

TD-Det [16] was proposed for aphid detection with two core designs, a transformer-feature pyramid network (TFPN) and a multi-resolution training method (MTM). Most of these existing methods attempt to identify individual aphids for counting and usually take place in ideal lighting conditions. The study by Zhang *et al.* [21] addressed the need for a dataset of pests within their natural habitat, creating a new aphid cluster detection dataset where images affected by aphids were manually selected and annotated. Instead of individual aphids, they were annotated in clusters to assess the overall infestation levels across the field. Bounding boxes were created based on these aphid clusters and the performance of four object detection models was benchmarked on square image patches of 400 pixels. However, the detection bounding box is not an ideal measure for infestation levels. As the importance of real-time segmentation systems is becoming more prevalent, the feasibility of such when it comes to identifying pests is ripe for study.

## 3. Dataset

Using a three-camera GoPro rig at different heights, a variety of images were captured from a sorghum field over two seasons to gather images of aphid clusters. Most images did not contain aphids as they are localized in specific regions of the field, so images without aphids were filtered

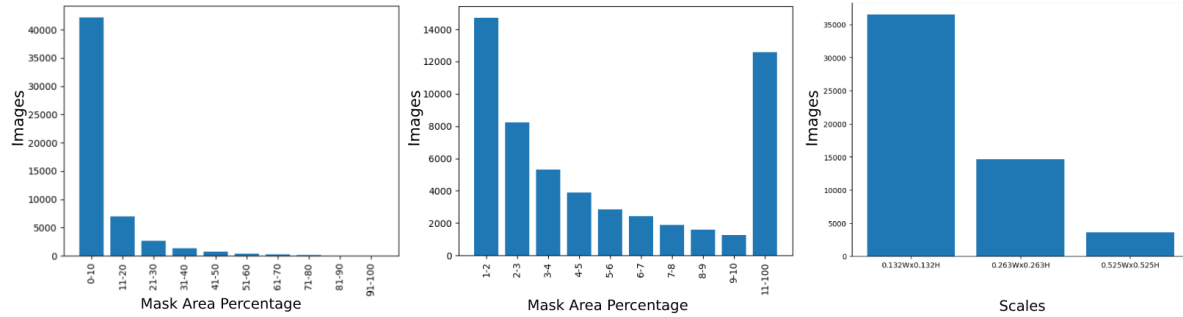


Figure 3. Histograms showing the mask area percentage across the images as well as the number of images per scale. The chart on the left shows the percentage of aphid cluster masks using intervals of 10%. As most take place in the interval of 0% to 10%, the chart in the center further breaks that interval down for better analysis. On the right, we can see the comparison between the number of images at each scale.

out, leaving 5,447 images. These were manually labeled by trained researchers to create segmentation masks, resulting in 59,767 labeled aphid cluster images. To ensure sufficient representation of aphid clusters, images with less than 1% of pixels containing aphid clusters were filtered out, leaving a total of 54,742 images in the dataset.

In fields that are often plagued by aphid infestations, these pests tend to group closely together rather than being distributed individually. Therefore, to accurately assess the extent of infestation, it is more useful and efficient to monitor for clusters of aphids. However, it is necessary to set a proper threshold to ensure that a sufficient number of aphids are present to constitute a cluster. The study [21] defined any group of six or more aphids located close to one another as an aphid cluster. This approach ensures that if some aphids were too sparsely distributed, they would not be considered a significant economic threat. The aphid clusters in our dataset were also considered to be those with six or more aphids to meet the same threshold.

The original collected images consisted of very high-resolution images where most masks made up only 0.015% of the images. The study [21] demonstrated the feasibility of utilizing square patches of 400 pixels with 50% overlap between adjacent patches for the purpose of cluster detection. While this may aid models to better learn to detect the clusters, for the purpose of real-time segmentation on high-resolution images the models must be more robust to detecting aphid clusters at different scales. Thus, we propose to generate patches at different scales to allow the model to be able to better generalize to aphids and images captured at different scales. Patches were generated at scales of  $0.132W \times 0.132H$  (Scale-1),  $0.263W \times 0.263H$  (Scale-2), and  $0.525W \times 0.525H$  (Scale-3), where W and H refer to the width and height of the original image, respectively. An example of what these scales look like is shown in Fig. 2. In each case, 10% overlap was used for adjacent patches for the completeness of masks. Patch generation was performed after fold separation to prevent data leakage across

them. All three scales were combined into the multiscale dataset and separated into 10 sets of images to perform 10-fold cross validation.

The final dataset contained 35,140, 13,311, and 6,291 images taken from cameras at the top, middle, and bottom heights respectively and 54,742 images in total. From these images, there were 36,478, 14,628, and 3,636 images from Scale-1, Scale-2, and Scale-3, respectively. The percentage of aphid clusters in each image was also analyzed and can be seen in Fig. 3. The large majority of these clusters are very small with an average cluster percentage of 2.45%.

#### 4. Real-Time Segmentation

The development of real-time segmentation has been a significant factor in the increasing use of autonomous platforms, such as self-driving cars, robots, and drones. These platforms rely on the ability to make accurate predictions quickly, which is made possible through the low latency of real-time segmentation models. In 2016, the Efficient Neural Network (ENet) was introduced, which used efficient convolutions and a compact architecture to enable real-time semantic segmentation [11]. This model was a significant breakthrough and inspired the development of other efficient and compact models for real-time computer vision applications.

Bilateral Segmentation Network (BiSeNet) [19] proposed to combine low-level features and high-level semantic information to find a good balance of speed and segmentation performance, achieving 68.4% mIoU on high-resolution input from the Cityscapes test dataset at speeds of 105 FPS. A new version of this architecture, BiSeNetV2, was proposed in 2020 [18] and improved upon both the performance and the segmentation speed of its predecessor. This network managed to achieve 72.6% mIoU on high-resolution input from the Cityscapes test set with faster speeds of 156 FPS. Fast-SCNN [13] is another popular real-time segmentation model which achieves good segmentation speeds for high-resolution image data while also be-

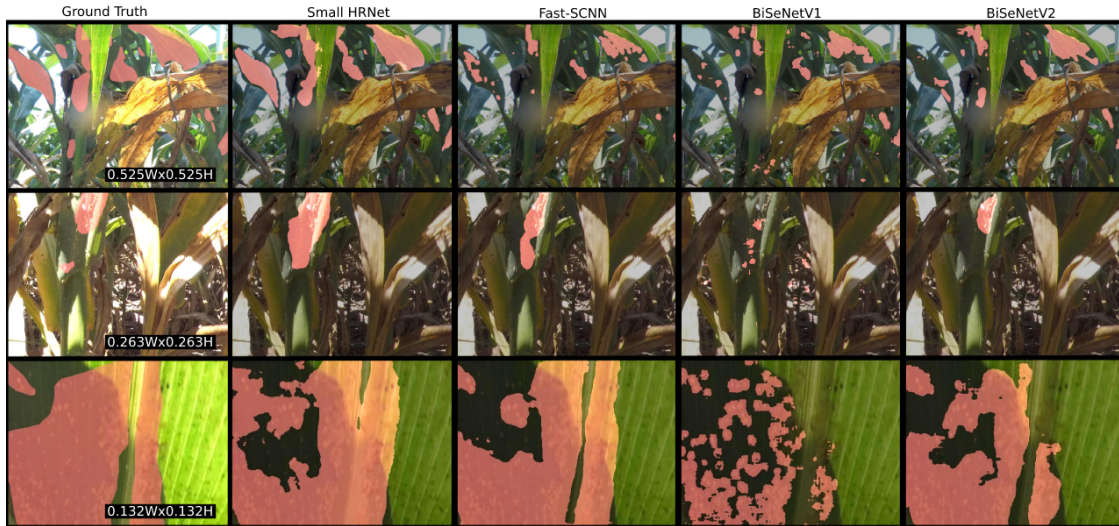


Figure 4. Comparison between the ground truths and the segmentation results of the real-time segmentation models. The results of Small HRNet, Fast-SCNN, BiSeNetV1, and BiSeNetV2 are shown in the second, third, fourth, and fifth columns respectively. The three rows show segmentation results at Scale-3 (top), Scale-2 (middle), and Scale-1 (bottom).

ing suitable for embedded devices with low computational memory. The model achieves this by learning low-level features from multiple resolution branches simultaneously. On the Cityscapes dataset, Fast-SCNN is able to achieve an accuracy of 68% mIoU at 123.5 FPS. In 2019, High-Resolution Network (HRNet) was introduced by [17] as a strong backbone for many computer vision problems where high-resolution representations are essential. It maintains high resolution throughout the whole process, resulting in semantically rich representations that are more spatially aware. A simpler version of this network, Small HRNet, managed to achieve real-time segmentation speeds at the slight cost of accuracy. It did so by using fewer layers and smaller width [17, 20]. The original HRNet achieved an accuracy of 80.9% on the Cityscapes test set while its smaller variant achieved 73.86% mIoU at the cost of faster speeds.

BiSeNetV1, BiSeNetV2, Fast-SCNN, and Small HRNet are all models that achieve real-time speeds across different datasets while also working with high-resolution images to obtain excellent performance. With the overall objective of an image analysis system for rapid pest identification and subsequent pest severity quantification at high spatial resolution, it is necessary for an approach that takes these into consideration. Thus, these four models were chosen for our problem of real-time aphid cluster segmentation using high-resolution images. By selecting a real-time segmentation solution, the ability to quickly segment aphid clusters in their natural setting is enhanced, allowing for efficient use of real-time information in autonomous systems. However, the real-time segmentation of pests in natural settings has not been extensively studied. This paper aims to draw

attention to this challenge and demonstrate the potential effectiveness of such a system.

## 5. Experiments

*Pre-Processing:* The original dataset images were of very high resolution ( $3647 \times 2736$ ), from which image patches were generated at three different scales,  $0.132W \times 0.132H$ ,  $0.263W \times 0.263H$ , and  $0.525W \times 0.525H$ . These were then combined together and subsequently split into 10 sets for 10-fold cross validation.

As a pre-processing step, all images were resized to  $1024 \times 768$  prior to training. The images were also normalized according to the mean and standard deviation of the dataset. It was found that as most masks only consisted of a small percentage of the image, there is a large class imbalance between the background and aphid clusters. To account for this, the class weights were calculated from the pixels in the image masks as follows:

$$W_{Aphid\ Cluster} = \frac{Total\ Pixels}{Aphid\ Cluster\ Pixels} \quad (1)$$

$$W_{Background} = \frac{Total\ Pixels}{Background\ Pixels} \quad (2)$$

*Training Set Up:* The models were all implemented with PyTorch in Python using the MMSegmentation framework [4]. They were trained on a Linux server with 4 NVIDIA Tesla P100 GPUs with 48G memory in total.

*Training Pipeline:* Stochastic Gradient Descent (SGD) was used with a learning rate of 0.001, a momentum of 0.9, and a weight decay of 0.0005. The models were trained

Model	mIoU	mDice	mPrecision	mRecall	FPS
HRNet-Small	<b>71.62 ± 0.47</b>	<b>81.15 ± 0.36</b>	<b>80.82 ± 1.20</b>	<b>81.64 ± 0.65</b>	31.57
Fast-SCNN	71.25 ± 0.59	80.87 ± 0.50	80.46 ± 1.47	81.21 ± 0.67	<b>91.66</b>
BiSeNetV1	59.94 ± 0.54	69.22 ± 0.70	72.39 ± 1.55	67.12 ± 1.40	56.19
BiSeNetV2	65.72 ± 0.53	75.58 ± 0.55	77.47 ± 1.34	74.06 ± 1.05	53.70

Table 1. Real-time segmentation results from 10-fold cross-validation. **Bold** indicates top result.

Model	mIoU	mDice	mPrecision	mRecall	FPS
PSPNet	<b>75.34</b>	<b>84.33</b>	82.55	<b>86.37</b>	6.46
DeepLabV3	75.05	84.06	<b>83.21</b>	83.92	3.86
HRNet	73.82	83.07	82.21	83.98	<b>17.27</b>
FCN	72.43	81.88	81.43	82.35	6.43

Table 2. Nonreal-time segmentation results. **Bold** indicates top results.

with a batch size of 2 for 160,000 iterations. To calculate the loss during training, we utilized the Cross-Entropy Loss and Dice loss with weights of 1 and 2 respectively.

*Model Evaluation:* To evaluate our models, we used the intersection over union (IoU) metric, one of the foundational evaluation metrics in detection and segmentation, as well as the Dice score, a very popular evaluation metric for segmentation. These metrics are defined as follows:

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \quad (3)$$

$$Dice = \frac{2 \times \text{Area of Overlap}}{\text{Total Area}} \quad (4)$$

We also calculate the Precision and Recall of the models, and they are defined as follows:

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

To compare the speed with which the models are able to segment images to detect aphid clusters, we will be evaluating the frames per second (FPS) of each model.

## 6. Results

### 6.1. Training Real-Time Models

In our experiments, four state-of-the-art real-time semantic segmentation models were used to perform 10-fold cross-validation. We initially separated the original high-resolution images into separate groups and then generated patches from them at the three scales. We combined these

generated patches into a single dataset, allowing the models to generalize to what aphid clusters look like at multiple scales. We then proceeded to train our models on this combined multiscale dataset. The performance of these models is shown in Table 1. The results show the mean intersection over union (mIoU), mean dice score (mDice), mean precision (mPrecision), mean recall (mRecall), and the speed in frames per second (FPS). As we can see, there is a usual trade-off between accuracy and speed. Small HRNet achieved the best accuracy while having a much slower speed compared to the other models at 31.57 FPS, albeit still in real-time. On the other hand, Fast-SCNN achieved excellent above real-time segmentation speeds of 91.66 FPS while maintaining a comparably good accuracy, only slightly lower than Small HRNet. The BiSeNetV1 model achieved the worst accuracy out of the chosen models but was able to achieve a respectable 56.19 FPS. BiSeNetV2 improved upon the performance of its predecessor while taking a slight blow to its speed. From these results, we have a clear distinction of the best-performing models with Small HRNet and Fast-SCNN. They are able to achieve the highest accuracy while also keeping above the 30 FPS real-time threshold.

### 6.2. Comparison Against Nonreal-Time Models

To provide adequate context regarding the segmentation accuracy obtained, four popular image segmentation models were also trained with the same hyperparameters. These models include the original version of HRNet, Pyramid Scene Parsing Network (PSPNet), DeepLabV3, and Fully Convolutional Network (FCN). These models are quite frequently used for many complicated semantic segmentation tasks and are chosen to allow us to analyze how accuracy and more complicated network structures would impact segmentation speed. These models were trained on the same generated dataset containing images at multiple scales. Their performance is shown in Table 2. It is evident that the best performance was obtained by PSPNet. The performance of HRNet is also shown to be slightly better than its small counterpart, with just over half the speed of Small HRNet at 17.27 FPS. DeepLabV3 is able to achieve very similar performance to PSPNet while FCN, a relatively older network, achieves a lower accuracy compared to the other nonreal-time models but is able to still outperform

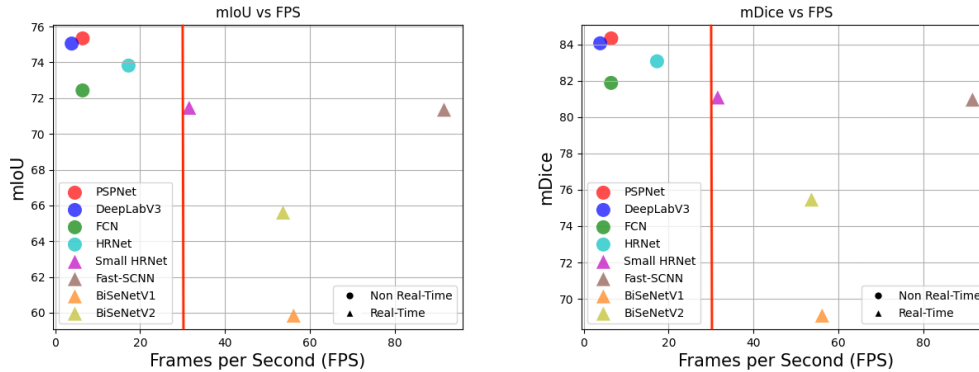


Figure 5. Scatter plots showing the accuracy vs the segmentation speed in frames per second (FPS). The plot on the left shows the accuracy as mean intersection over union while the right shows it as mean Dice score. The nonreal-time models are marked with circles while the real-time segmentation models are marked with triangles. The red line in each plot indicates 30 FPS which is considered the real-time speed threshold.

Model	0.132H×0.132W				0.263H×0.263W				0.525H×0.525W			
	mIoU	mDice	mIoU(M)	mDice(M)	mIoU	mDice	mIoU(M)	mDice(M)	mIoU	mDice	mIoU(M)	mDice(M)
HRNet-Small	69.91	79.63	65.11	74.84	69.73	78.95	65.14	74.84	68.33	77.21	59.41	68.1
Fast-SCNN	69.88	79.6	65.58	75.29	70.07	79.25	67.16	76.93	68.82	77.73	65.52	75.38
BiSeNetV1	59.77	68.97	57.60	66.07	64.64	73.64	60.39	69.53	65.02	73.57	55.75	63.29
BiSeNetV2	65.51	75.32	60.49	69.68	66.69	75.79	61.23	70.39	66.09	74.81	57.78	66.02

Table 3. Results of the real-time segmentation models across the individual scales. For each scale, mIoU and mDice give the accuracy when tested only on images at their respective scale while mIoU(M) and mDice(M) denote the accuracy when tested on the combined multiscale version of the dataset.

all the real-time models. The segmentation speed of each model is also noted. While these models may achieve superior accuracy, they are unsuitable for our need for real-time segmentation, unlike the real-time models. Thus, through small sacrifices in accuracy, we are able to achieve real-time segmentation.

### 6.3. Comparison of Different Multiple Scales

In the study *et al.* [21], square patches of 400 pixels were used to create the aphid clusters dataset for the training of object detection. This was due to the original images having a very small percentage of them containing aphid clusters. In this paper, we wanted to study the effectiveness of using multiple scales to have a generalizable representation of aphid clusters for semantic segmentation. We trained our real-time segmentation models on each of the individual generated scales to see how well they would perform not only on their own scales but on images of other scales as well through the combined dataset. The performance of these models is shown in Table 3. For each scale, we show the accuracy for images at their own respective scale in terms of mIoU and mDice in the first two columns. The next two columns labeled mIoU(M) and mDice(M) show the accuracy on the combined dataset containing multiple

scales. For Small HRNet and Fast-SCNN, we see a decrease in the overall accuracy across the scales as well as a significant decrease in accuracy when applied to the combined dataset. For the two BiSeNet models, we see a rise in accuracy at their respective scales showing that they are able to learn individual scales better. This is especially true for BiSeNetV1 which is also able to achieve a similar result at the 0.263 scale as it is able to achieve on the experiments using the combined dataset from Table 1. However, for every other model, there is still a noticeable degradation in the accuracy for each scale when applied to the combined dataset. Thus, the use of the multiscale dataset has contributed to an increase in performance for almost all models.

### 6.4. Segmentation Results

Figure 4 shows the segmentation results in comparison to the ground truths. The segmentation results of Small HRNet matched the ground truth mask most closely. It is able to detect most clusters that are sparsely located throughout the crop canopies, partially occluded, and also hidden by shadows. Small HRNet is also able to more accurately handle complicated boundaries. Fast-SCNN, on the other hand, creates tighter boundaries around the clusters and is able to avoid segmenting areas within and around the cluster

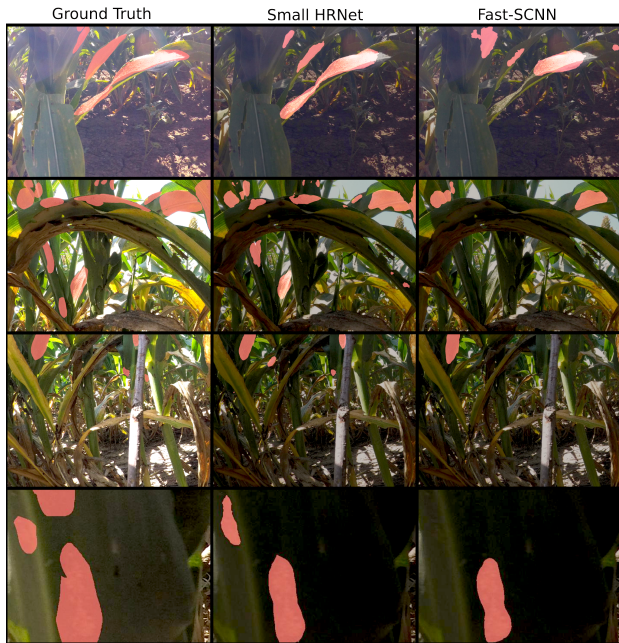


Figure 6. Further segmentation results from Small HRNet and Fast-SCNN alongside the ground truth images. From the different lighting conditions, blur, and cluster sizes we can make a better comparison between these two better-performing models.

that are only leaves without any aphids. In contrast to this, BiSeNetV1 is able to detect the general area of the aphid clusters but it also erroneously confuses other similarly textured parts of crops and flowers as clusters. BiSeNetV2 improves upon this and is able to create larger segmentation boundaries around the clusters and avoids segmenting other similarly textured parts of the crops. Both BiSeNetV1 and BiSeNetV2 are not as robust at handling different lighting conditions as Small HRNet and Fast-SCNN.

## 7. Discussion

Throughout the original collected images, aphid clusters usually only take up a very small percentage of each high-resolution image. It is important to ensure that models are robust to what these aphid clusters look like at different scales, at high resolutions with very small features, as well as at lower resolutions where larger features are able to be made out. Through the training of models with a variety of scales, we ensure they are able to generalize to changes in scale more effectively as seen through the increase in performance of the combined dataset compared to the individual scaled datasets.

From our results, apart from Fast-SCNN, there is a clear trade-off between accuracy and speed. This is especially evident in the case of HRNet and Small HRNet. By reducing the depth and the width and updating its stem to consist of two  $3 \times 3$  convolutions with a stride of 2 as stated in [7, 20],

Small HRNet is able to improve upon the speed from 17.27 FPS to 31.57 FPS with only a small decrease in the accuracy. Fast-SCNN managed to achieve excellent speeds through the downsampling of images and the efficient sharing of computed low-level features. However, the simplicity of its architecture holds its performance back slightly behind Small HRNet.

Both Small HRNet and Fast-SCNN are able to perform very well with similar segmentation results. To demonstrate some key differences between them, Figure 6 shows some segmentation results of the two better-performing models where they had a big difference in their segmentation results. From the examples in this figure, we can see that smaller clusters are better detected by Small HRNet. Small HRNet is also able to better handle different lighting conditions such as darker regions underneath the canopies. However, Fast-SCNN is able to create tighter boundaries around the clusters compared to the much larger ones from Small HRNet. While this is better for smaller clusters or those that do not have many complicated shapes, it often misses surrounding smaller aphid subclusters that are present but separated from the main cluster body. Fast-SCNN is also able to better handle blurry images containing clusters.

Considering accuracy, both Small HRNet and Fast-SCNN are good candidates for the ideal choice. While the former has a slightly better performance, when considering the efficient nature of Fast-SCNN, its ability to find tighter boundaries of smaller clusters, and its very high framerate, it distinguishes itself as the ideal choice for the purpose of real-time aphid cluster segmentation.

## 8. Conclusion

Automated pest detection has become an increasingly popular research problem in recent years. In this study, we have collected high-resolution images and manually labeled clusters of aphids to better assess infestation severity in natural settings. We have created a dataset for semantic segmentation using three different scales to enhance generalization and trained four real-time segmentation models to evaluate their performance relative to state-of-the-art and popular semantic segmentation models on high-resolution input. Our experiments showed a trade-off between segmentation accuracy and speed due to the complexity of the network structure. Fast-SCNN emerged as the optimal choice, with real-time segmentation speed and efficient performance. Small HRNet also showed promise with slightly better accuracy, albeit at a cost of speed. Our findings suggest that automated pest control systems can use these real-time models to efficiently and accurately aid farmers in pesticide application, minimizing crop exposure to pesticides. These promising results motivate further research in automated pest control, bringing us closer to a more efficient and sustainable agricultural system.

## References

- [1] Jayme Garcia Arnal Barbedo. Using digital image processing for counting whiteflies on soybean leaves. *Journal of Asia-Pacific Entomology*, 17(4):685–694, 2014. 1, 2
- [2] Jian Chen, Yangyang Fan, Tao Wang, Chu Zhang, Zhengjun Qiu, and Yong He. Automatic segmentation and counting of aphid nymphs on leaves using convolutional neural networks. *Agronomy*, 8(8):129, 2018. 1, 2
- [3] Xiangyu Chen, Ying Qin, Wenju Xu, Andrés M Bur, Cuncong Zhong, and Guanghui Wang. Improving vision transformers on small datasets by increasing input information density in frequency domain. In *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2022. 2
- [4] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. <https://github.com/open-mmlab/mms Segmentation>, 2020. 4
- [5] Cambria Finegold, Jeffrey Ried, Katherine Denby, and Sarah Gurr. Global burden of crop loss. *Gates Open Res*, 3, 2019. 1
- [6] Lei He, Jiwen Lu, Guanghui Wang, Shiyu Song, and Jie Zhou. Sosd-net: Joint semantic object segmentation and depth estimation from monocular images. *Neurocomputing*, 440:251–263, 2021. 2
- [7] Christopher J Holder and Muhammad Shafique. On efficient real-time semantic segmentation: a survey. *arXiv preprint arXiv:2206.08605*, 2022. 7
- [8] Kaidong Li, Mohammad I Fathan, Krushi Patel, Tianxiao Zhang, Cuncong Zhong, Ajay Bansal, Amit Rastogi, Jean S Wang, and Guanghui Wang. Colonoscopy polyp detection and classification: Dataset creation and comparative evaluations. *Plos one*, 16(8):e0255809, 2021. 2
- [9] Tao Liu, Wen Chen, Wei Wu, Chengming Sun, Wenshan Guo, and Xinkai Zhu. Detection of aphids in wheat fields using a computer vision technique. *Biosystems Engineering*, 141:82–93, 2016. 2
- [10] Wenchi Ma, Xuemin Tu, Bo Luo, and Guanghui Wang. Semantic clustering based deduction learning for image recognition and classification. *Pattern Recognition*, 124:108440, 2022. 2
- [11] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*, 2016. 3
- [12] Krushi Bharatbhai Patel, Fengjun Li, and Guanghui Wang. Fuzzynet: A fuzzy attention module for polyp segmentation. In *NeurIPS’22 Workshop on All Things Attention: Bridging Different Perspectives on Attention*. 2
- [13] Rudra PK Poudel, Stephan Liwicki, and Roberto Cipolla. Fast-scnn: Fast semantic segmentation network. *arXiv preprint arXiv:1902.04502*, 2019. 3
- [14] Usman Sajid and Guanghui Wang. Towards more effective prm-based crowd counting via a multi-resolution fusion and attention network. *Neurocomputing*, 474:13–24, 2022. 2
- [15] Annie Shattuck. Generic, growing, green?: The changing political economy of the global pesticide complex. *The Journal of Peasant Studies*, 48(2):231–253, 2021. 1
- [16] Yue Teng, Rujing Wang, Jianming Du, Ziliang Huang, Qiong Zhou, and Lin Jiao. Td-det: A tiny size dense aphid detection network under in-field environment. *Insects*, 13(6):501, 2022. 1, 2
- [17] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020. 4
- [18] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. *International Journal of Computer Vision*, 129:3051–3068, 2021. 3
- [19] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 325–341, 2018. 3
- [20] Changqian Yu, Bin Xiao, Changxin Gao, Lu Yuan, Lei Zhang, Nong Sang, and Jingdong Wang. Lite-hrnet: A lightweight high-resolution network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10440–10450, 2021. 4, 7
- [21] Tianxiao Zhang, Kaidong Li, Xiangyu Chen, Cuncong Zhong, Bo Luo, Ivan Grijalva Teran, Brian McCornack, Daniel Flippo, Ajay Sharda, and Guanghui Wang. A new dataset and comparative study for aphid cluster detection. In *2nd AAAI Workshop on AI for Agriculture and Food Systems*, 2023. 2, 3, 6
- [22] Tianxiao Zhang, Xiaohan Zhang, Yiju Yang, Zongbo Wang, and Guanghui Wang. Efficient golf ball detection and tracking based on convolutional neural networks and kalman filter. *arXiv preprint arXiv:2012.09393*, 2020. 2