Received: 23 January 2023 | Accepted: 26 June 2023

DOI: 10.1002/ppj2.20079

The Plant Phenome Journal OPEN ACCESS

ORIGINAL ARTICLE

Self-supervised learning improves classification of agriculturally important insect pests in plants

Soumyashree Kar[1] | **Koushik Nagasubramanian**[2] | **Dinakaran Elango**[1] | **Matthew E. Carroll**[1] | **Craig A. Abel**[3] | **Ajay Nair**[4] | **Daren S. Mueller**[5] | **Matthew E. O'Neal**[5] | **Asheesh K. Singh**[1] | **Soumik Sarkar**[2] | **Baskar Ganapathysubramanian**[2] | **Arti Singh**[1]

[1]Department of Agronomy, Iowa State University, Ames, IA, USA

[2]Department of Mechanical Engineering, Iowa State University, Ames, IA, USA

[3]USDA, Agricultural Research Service, Corn Insects and Crop Genetics Research Unit, Ames, IA, USA

[4]Department of Horticulture, Iowa State University, Ames, IA, USA

[5]Department of Plant Pathology, Entomology and Microbiology, Iowa State University, Ames, IA, USA

Correspondence
Arti Singh, Department of Agronomy, Iowa State University, Ames, IA, USA.
Email: arti@iastate.edu

Assigned to Associate Editor Weizhen Liu.

Funding information
NSF, Grant/Award Number: 1952045; CPS Frontier, Grant/Award Number: 1954556; Agricultural Research Service; National Institute of Food and Agriculture, Grant/Award Numbers: 2019-67021-29938, 2021-67021-35329, 2022-67013-37120; U.S. Department of Agriculture, Grant/Award Number: IOW04714

Abstract

Insect pests cause significant damage to food production, so early detection and efficient mitigation strategies are crucial. There is a continual shift toward machine learning (ML)-based approaches for automating agricultural pest detection. Although supervised learning has achieved remarkable progress in this regard, it is impeded by the need for significant expert involvement in labeling the data used for model training. This makes real-world applications tedious and oftentimes infeasible. Recently, self-supervised learning (SSL) approaches have provided a viable alternative to training ML models with minimal annotations. Here, we present an SSL approach to classify 22 insect pests. The framework was assessed on raw and segmented field-captured images using three different SSL methods, Nearest Neighbor Contrastive Learning of Visual Representations (NNCLR), Bootstrap Your Own Latent, and Barlow Twins. SSL pre-training was done on ResNet-18 and ResNet-50 models using all three SSL methods on the original RGB images and foreground segmented images. The performance of SSL pre-training methods was evaluated using linear probing of SSL representations and end-to-end fine-tuning approaches. The SSL-pre-trained convolutional neural network models were able to perform annotation-efficient

Abbreviations: BYOL, Bootstrap Your Own Latent; CNN, convolutional neural network; DL, deep learning; FAW, fall armyworm; FN, false negatives; FP, false positives; HTP, high throughput phenotyping; IA-IP22, IA insect-pest dataset 22; IPM, integrated pest management; ML, Machine Learning; NNCLR, Nearest Neighbor Contrastive Learning of Visual Representations; RGB, red, green, and blue; sgd, stochastic gradient descent; SSL, self-supervised learning; TN, true negatives; TP, true positives.

Soumyashree Kar and Koushik Nagasubramanian contributed equally.

classification. NNCLR was the best performing SSL method for both linear and full model fine-tuning. With just 5% annotated images, transfer learning with ImageNet initialization obtained 74% accuracy, whereas NNCLR achieved an improved classification accuracy of 79% for end-to-end fine-tuning. Models created using SSL pre-training consistently performed better, especially under very low annotation, and were robust to object class imbalances. These approaches help overcome annotation bottlenecks and are resource efficient.

1 | INTRODUCTION

Insect pests cause yield losses of up to 40% globally, with estimated revenue losses of $220 billion (Gullino et al., 2021). Insect populations are influenced by temperature and other environmental conditions, so future climate change is predicted to affect insect-pest outbreaks (Liebhold & Bentz, 2011; Skendžić et al., 2021). Resistant varieties and integrated pest management (IPM) strategies are effective methods to control insect pests. IPM and other management interventions require the timely identification of different insect pests, which also reduces the usage of excessive pesticides. Therefore, developing tools to identify diverse insects would benefit both farmers and the broader science community. The lack of species-specific visual features (due to extreme visual similarities between various insects), very specific and short activity duration, mobile nature, and the propensity to hide under the leaves in clusters, and so on often lead to misidentification and make insect-pest detection an extremely challenging problem (Zhong et al., 2018). Timely pest detection would lower production costs and adverse environmental impacts and help contribute to better human health and food safety (Hao et al., 2020).

High-throughput phenotyping (HTP) tasks have been one of the successful applications of machine learning (ML) and computer vision in the past decade including plant stress phenotyping (Singh et al., 2016; 2021a). Since 2016, deep learning (DL)-based methods have been successfully deployed in a variety of applications to extract plant traits, such as pod counting (Riera et al., 2021), crop yield (Shook et al., 2021), weed detection (Bah et al., 2018; dos Santos Ferreira et al., 2017; Osorio et al., 2020; Razfar et al., 2022), insect identification (Ahmad et al., 2022; Bereciartua-Pérez et al., 2022; Li et al., 2021), disease detection (Ghosal et al., 2018; Kulkarni, 2018; Mohanty et al., 2016; Rairdin et al., 2022; Rangarajan et al., 2018), nutrient deficiency detection (Azimi et al., 2021; Bahtiar et al., 2020; Barbedo, 2019; Waheed et al., 2022; Yi et al., 2020), and root nodules (Jubery et al., 2021). Although conventional DL-based supervised classification and object detection are powerful models, they require large volumes of labeled data (Singh et al., 2018). The DL architecture enables the extraction of a suite of features from images using a multilayer neural network, such as ConvNet or ResNet (Li et al., 2019). Therefore, several studies have reported the comparative performance of multiple DL architectures with respect to conventional supervised methods in classifying crop insect pests (Tetila et al., 2020; Thenmozhi & Srinivasulu Reddy, 2019; Xia et al., 2018). The three most reported convolutional neural network (CNN) models for insect-pest classification are versions of VGGNet, ResNet, and MobileNet, albeit some studies have reported 98% accuracy in classifying multiple crop insect pests by fine-tuning models like GoogLeNet (Chen et al., 2020). The latter is, however, both resource- and time-intensive, hence not very common in this domain (Liu & Wang, 2021; Nanni et al., 2022). Considering the challenges in insect-pest classification tasks, large insect datasets have been published, for example, the IP102 dataset (Wu et al., 2019), and iNaturalist plant–insect interaction data (Gazdic & Groom, 2019). However, despite utilizing a combined deep-CNN and saliency-based approach and being trained on such datasets, models fail to perform desirably due to small inter-class and large intra-class variation in a multi-class pest detection problem (Singh et al., 2021a; Tetila et al., 2020).

Supervised DL methods provide promising results with very high classification accuracies; however, the amount of labeling needed to achieve desired accuracies is very high, making their applicability infeasible in many real-world cases (Tetila et al., 2020). Therefore, there is a pressing need to build a DL-based classification framework to address the issue of inter- and intra-class variabilities with limited annotation. In agriculture or other domains where data labeling is difficult, costly, time-consuming, or complex, there is a need to overcome the challenges of limited annotation, so that a robust DL method classification framework can be created. In this context, a state-of-the-art self-supervised learning (SSL) approach has been developed that learns useful latent representations from input data without human annotations. The efficiency of employing an SSL approach over the conventional supervised methods has been shown in diverse domains, for example, diagnosis from medical imaging (Masood et al., 2015; Shurrab & Duwairi, 2022), autonomous navigation systems (Kahn et al., 2021), seismic imaging (Wang et al., 2020), and plant phenotyping (Nagasubramanian et al., 2022).

The SSL approach is built on a set of latent features, and therefore, carrying out downstream tasks gets very convenient with the significantly reduced amount of labeled data while performance is comparable with that of supervised learning (Caron et al., 2021; Grill et al., 2020; Nagasubramanian et al., 2021). An integral aspect of SSL that enables the learning of latent and complex high-level features from non-labeled data is augmentation. Tuning different augmentation parameters allows the backbone architecture to learn the underlying distortion-agnostic representations (Misra & van der Maaten, 2020). Thus, the pre-trained models obtained via SSL could be fine-tuned on annotated examples for target transfer-learning tasks. This becomes even more applicable where HTP is routinely utilized or deployed as a large trove of data is created, and classification tasks are the goal (Agastya et al., 2021; Margapuri & Neilsen, 2021; Nagasubramanian et al., 2021; Singh et al., 2021a).

Our main objectives were to develop an efficient classification model for economically important 22 insect classes in field and horticultural crops in Iowa, generate insight into real-world challenges faced in processing a large dataset for DL, present strategies to handle imbalanced dataset in various insect-pest classes using SSL, and solve fine-grained inter- and intra-class classification problems. As real-field images of insect pests are confounded with larger and complex backgrounds compared to the foreground, we hypothesize that image segmentation can aid with better latent representations from the foreground that can improve the overall classification performance. Therefore, this work focuses on demonstrating the role of efficient pre-training of the SSL methods for a significant reduction in the need for human annotation, and comparative performance assessment on both raw and segmented images. Further, we show the ability of foreground-aware SSL in addressing the abovementioned challenges and improved model performance. In this context, we present a novel insect-pest dataset (IA-IP22) collected from several fields in Iowa, comprising 22 insect-pest classes and 14,665 images collected using smartphones. Using this useful dataset, we investigated the efficacy of SSL in classifying 22 insect-pest classes via a meticulous DL-based classification framework that involves comparative assessment across 3 SSL algorithms, Nearest-Neighbor Contrastive Learning of Visual Representations (NNCLR) (Dwibedi et al., 2021), Bootstrap Your Own Latent (BYOL) (Grill et al., 2020), and Barlow Twins (Zbontar et al., 2021). The SSL and conventional transfer learning methods were employed to address the inter- and intra-class variabilities using raw and segmented images. In both methods, segmented images produced better results. For instance, with just 3% training data, the NNCLR self-supervised learner could classify the segmented images with 70.87% accuracy, whereas with raw images, the method was just 58.59% accurate. Additionally, compared to supervised learning, SSL with segmented images yielded visible performance gains. Such findings will be applicable to crop production and plant breeding (Singh et al., 2021b).

Core Ideas

- Insect pests cause significant damage to food production.
- Early detection and mitigation of insect pests are crucial in managing economic threshold level.
- We developed a self-supervised learning (SSL) model to identify insect pests with minimal annotations.
- SSL models greatly improve the identification and classification tasks.
- Entropy-masking-based segmentation aids SSL effectiveness.

2 | MATERIALS AND METHODS

2.1 | Dataset

Although multiple insect-pest datasets have been reported, including open source (Gazdic & Groom, 2019; Wu et al., 2019), we emphasized real-world settings and did not include any images sourced from the internet, to make the application easier in real-life settings that farmers and agronomists will encounter in agriculture. Further, the available insect datasets are mostly limited in the total image count, for example, 200 images (Venugoban & Ramanan, 2014), 1440 images (Xie et al., 2015), 5000 images (Tetila et al., 2020), and are sometimes crop-specific (Tetila et al., 2020; Venugoban & Ramanan, 2014). Limited size and variability in a dataset constrain the training of DL models in satisfactorily capturing the complex features for the detection or classification of the insect pests, which inherently possess significant inter- and intra-class variabilities (Wu et al., 2019).

To create a novel insect-pest dataset with practical applications, a team of agronomists visited several fields in the state of Iowa (IA), United States with the objective to collect insect-pest images of common species present in different crops. For real-world applicable images, smartphones (Android and iOS) were used to take photos throughout the day with a team of five people who collected images over the course of several weeks in July–August 2021. This ensured varied image specification, image variation across people/camera, and time of the day. Due to the presence of different insect species in varying numbers, we have an imbalance dataset in different insect species classes, which was desirable for the objective of this research project. Additional variation was created due to the imaging of insects in various crops, leaf, or stem in the background, different zooms while taking images, and variation in types of insect species present. It was noticed that the insects appeared at the top of the canopy mostly during the early morning or evening hours, when the temperature and environmental conditions were mild. This characteristic in insect sightings was also reported by Tetila et al. (2020). However, some insects like the Japanese beetles could be found in clusters throughout the day. We did not experience any challenge in collecting sufficient images for 21 of 22 insect species. However, fall armyworm (FAW) (*Spodoptera frugiperda*) was difficult to collect images because those were rarely sighted compared to the other insects. Hence, the larvae were first reared and grown in the lab and then imaged with varying background conditions to get a sizable number of FAW images.

To incorporate variability in the dataset, the images were also taken from varying camera angles with an intent to serve as a natural augmentation technique in training the models. Thus, the mentioned insect-pest dataset includes both between- and within-species variability in terms of type, size, shape, and overall visual features. All these phenotyping efforts led to the creation of "IA insect-pest dataset 22," that is, IA-IP22, which comprises 14,665 images across 22 insects (Figure 1), and the number of insects per class varies from 95 to 1653 (Figure 2). As few insects were extremely tiny to photograph, a very close-range 5× zoomed mode was primarily used; however, the zoom level differed based on the insect type and their location on the plant canopy. In the following section, the challenges faced with and the methodology for handling such data are demonstrated.

2.2 | Challenges in classifier training

Using this dataset is challenging from the ML perspective due to the following reasons:

i. Several classes had large intra-class variability in size, shape, color, patterns, and texture.
ii. Insects from different classes looked very similar, that is, very small inter-class variability.
iii. The dataset was highly class imbalanced.
iv. There was a large background compared to the insect or the foreground.
v. Due to varying illumination conditions in a day, shadow effects were also found.
vi. Many images consisted of multiple instances of the same insect, in cases where insects were found clustered on a leaf or flower, creating an impression of overlapping objects.
vii. Insects from different classes were found together in the same image.

These variabilities (Figure 3) not only make the classification task challenging but also make the dataset unique, because it unravels the opportunities for solving complex real-world computer vision problems (Singh et al. 2021c).

2.3 | Description of the SSL methods

SSL methods differ based on the augmentation approaches and the loss function definitions, which control the selection of the constraints and the way an optimal solution is achieved. The three SSL methods leveraged in this study are briefly described below.

2.4 | BYOL

BYOL is a distillation-based SSL method that does not rely on negative samples, unlike contrastive methods. It rather works on two same architecture networks, the online and the target network. The online network is tasked with learning the representations for an augmented view of an image, then predicting the representations of the target network trained on another augmentation of the same image. Although the online network gets updated as per the prediction errors, the target network weights are also simultaneously updated with the moving averages of the online network weights. Thus, BYOL enables self-supervision by learning interactively from two encoder networks (Grill et al., 2020).

2.5 | Barlow Twins

Barlow Twins also leverages two identical networks to learn image features, like BYOL. However, in the Barlow Twins method, embeddings from both the networks trained on different augmentations of the same images are cross-correlated. The model is optimized by making the cross-correlation matrix close to identity, such that the learned embeddings are distortion-agnostic providing maximized information. The objective function thus tries to minimize redundancy between the representations learned from the networks and works on a simpler concept than BYOL (Zbontar et al., 2021).

2.6 | NNCLR

NNCLR exploits a contrastive learning approach to finding positives from other samples closest in the latent space

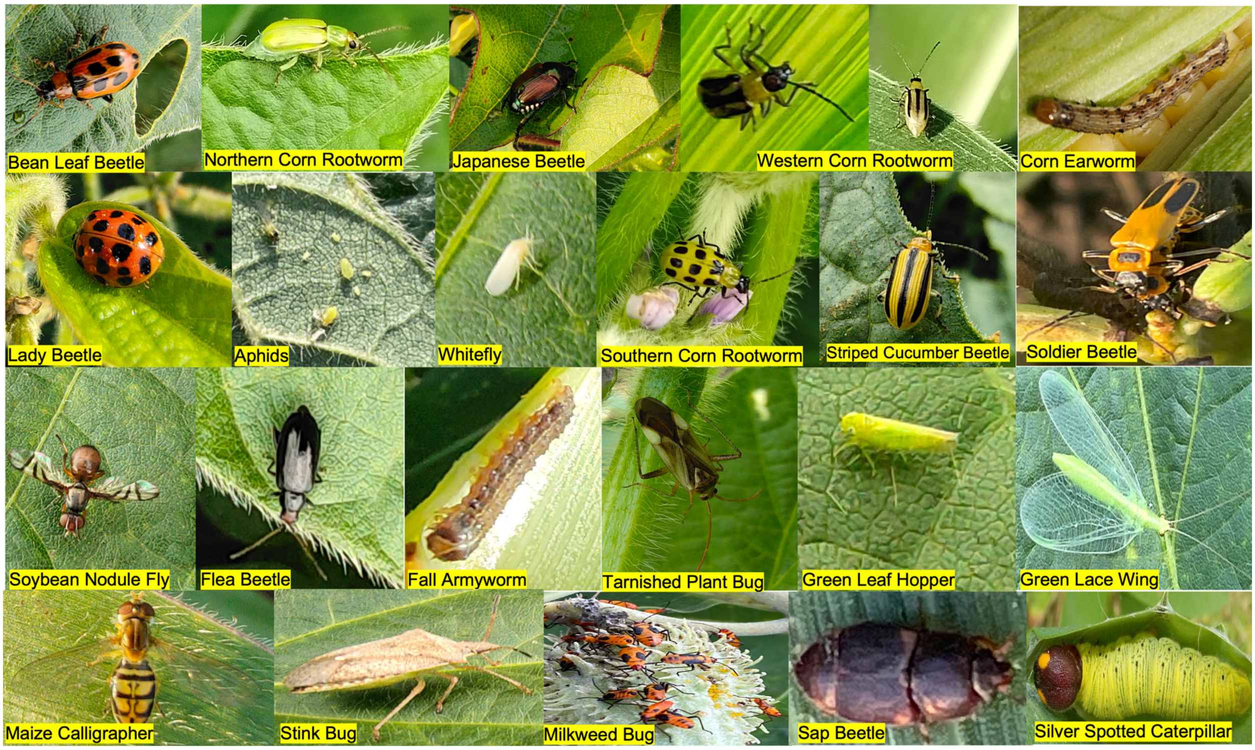


FIGURE 1 An illustration of some of the insect pest images collected from Iowa State University research fields in Iowa, USA. These represent the variety, type, and quality of the collected images.

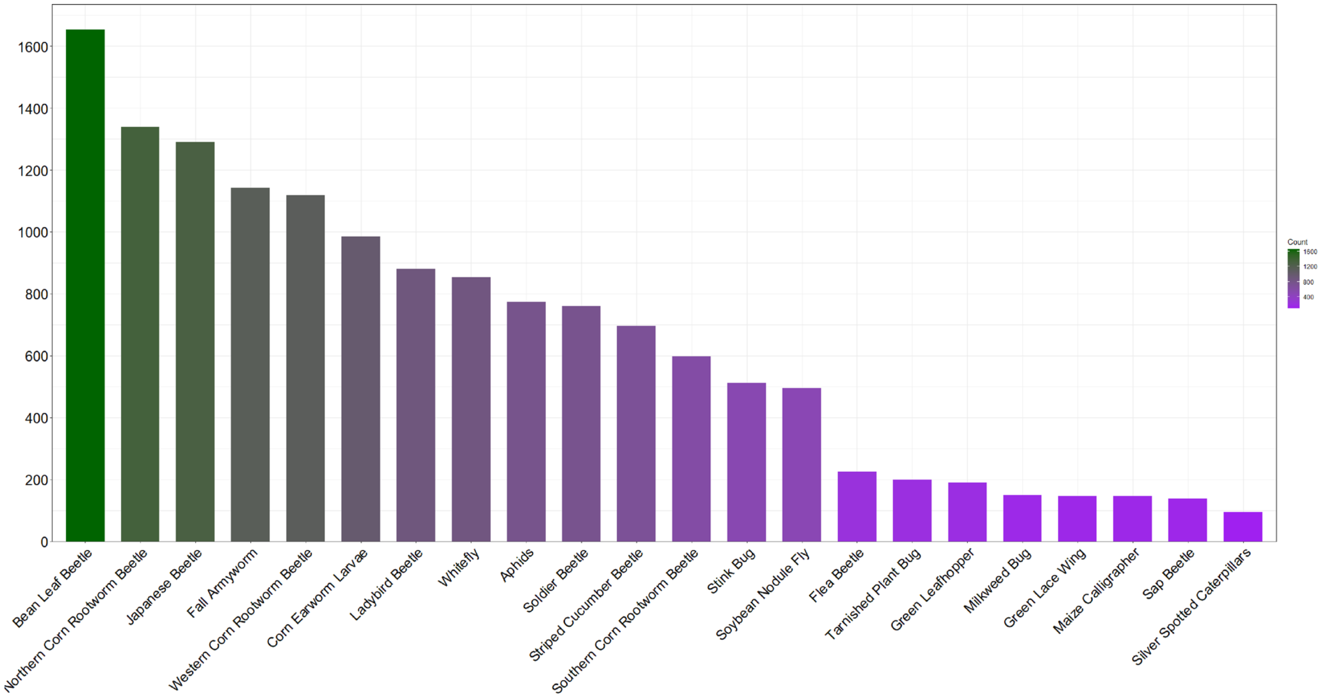


FIGURE 2 Plot representing the count of insect per class, arranged in descending order (top to bottom).

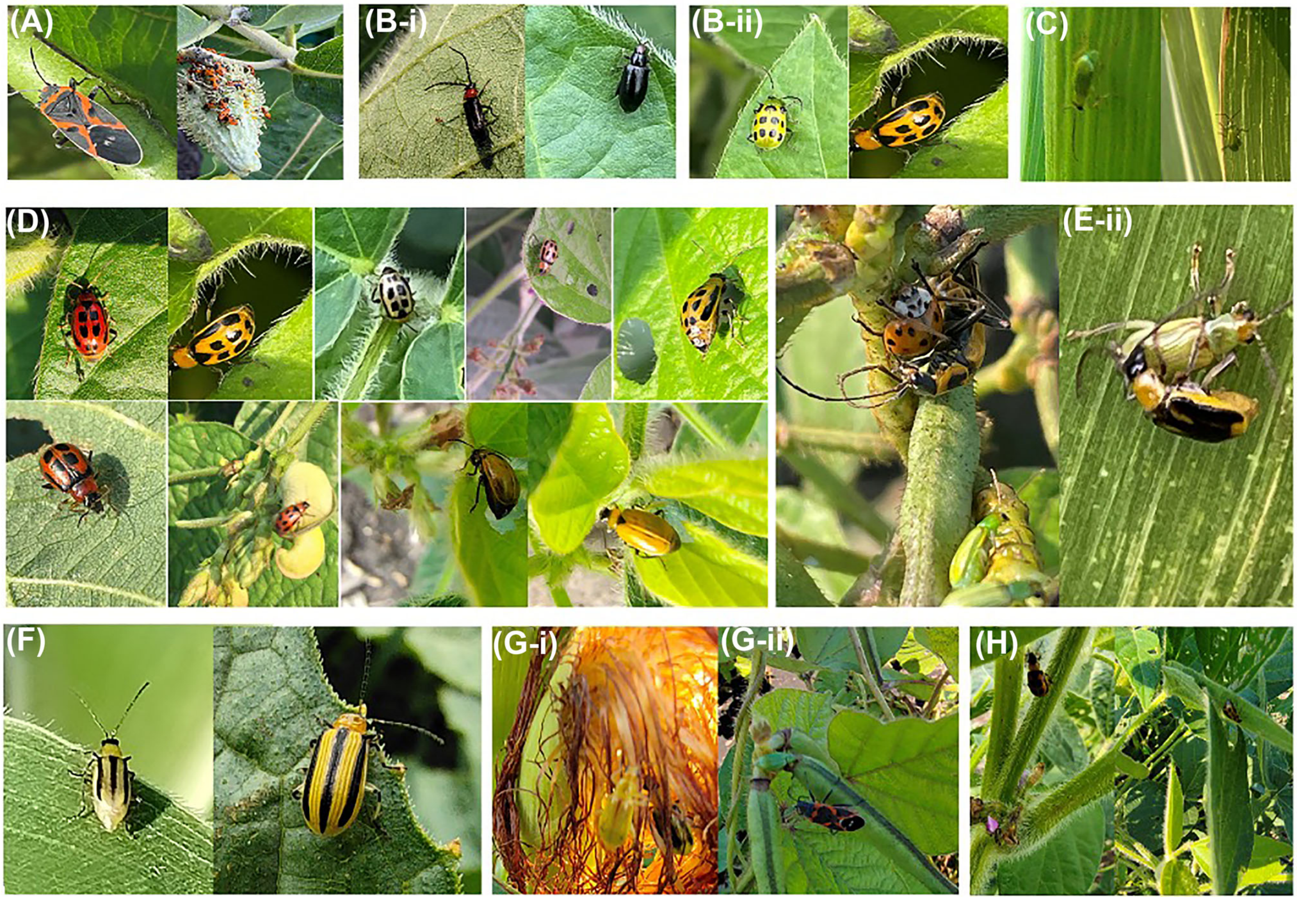


FIGURE 3 (A) Single (left) and multiple (right) instances of the same insect, milkweed bug; (B) two examples of similar-looking insects from different classes—(B-i) black soldier beetle (left) and sap beetle (right) and (B-ii) southern corn rootworm (left) and bean leaf beetle (right); (C) two examples of camouflaging background effect with an instance of a northern corn rootworm in each; (D) intra-class variability in the same insect class, bean leaf beetle; (E) multiple insect classes in the same image—(E-i) a lady beetle, one soldier beetle and two northern corn rootworms and (E-ii) a northern and a western corn rootworm; (F) visual similarity between western corn rootworm (left) and striped cucumber beetle (right); (g) instances of both noisy background and multiple insects in the same image—(G-i) northern and western corn rootworm and (G-ii) northern corn rootworm and milkweed bug; (H) background and illumination effects on the foreground, an instance of bean leaf beetle in each.

than using augmentations of the same image. This enables increasing semantic variability compared to the latter. The networks thus learn beyond a single discriminative instance in providing better invariance to different viewpoints, deformations, and even intra-class variations. This not only makes the method less reliant on complex data augmentations but also helps with significant improvement in performance in downstream tasks.

2.7 | Workflow

The classification framework comprises three major steps: data pre-processing and extraction of segmented images, deriving latent representations through different self-supervised procedures, and finally, classification. Considering the complexity of the images, SSL performance on raw and segmented images was compared via linear evaluation of the representations learned in both cases. Subsequently, supervised fine-tuning was performed to compare supervised versus self-supervised results. In this process, two backbone architectures, ResNet-18 and -50 (RN 18/50), were examined for different sampling strategies, random, random-augmented, diverse, and diverse-augmented, with label fractions of the sample varying from 0.1%, 0.3%, 0.5%, and 100%. All the experiments were repeated three times, and the average results from each method were used to compare between SSL and SL performances. Thus, this paper primarily aims to examine the minimum amount of training data needed to obtain at least 80% classification accuracy, and how efficiently SSL helps in handling class imbalance. The detailed methodology is illustrated in Figure 4.

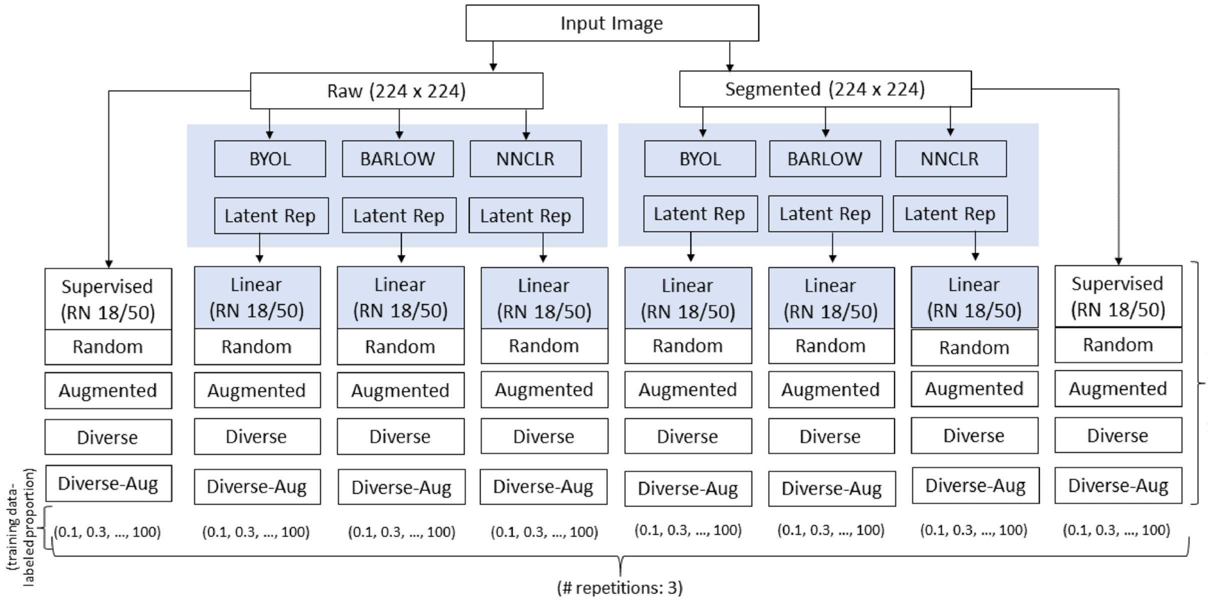


FIGURE 4 Detailed methodology flowchart, representing the two input sets, raw/segmented, the backbone architectures ResNet-18 and 50 (RN 18/50), sampling strategies—random, random-augmented, diverse, and diverse-augmented, and the labeled fractions of sample varying from 0.1%, 0.3%, 0.5%, to 100%.

2.8 | Data pre-processing (raw and segmented) and pre-training setup

The dataset was first cleaned up by removing the duplicate and empty images. Then it was partitioned in an approximately 70:15:15 ratio yielding 10,725 training images, 2081 validation, and 1859 test images. All the images were resized to 224 × 224 dimensions for processing efficiency, and then, training samples were labeled with increasing proportion, that is, [0.1, 0.3, 0.5, 0.7, 1, 3, 5, 7, 10, 30, 50, 70, 100]. This approach would eventually help identify the amount of training data ideally required for reasonable SSL performance. In this study, four different sampling strategies were adopted: (a) random, (b) diverse (by selecting diverse samples from the latent space of the encoder output (Bortolato et al., 2022), (c) random-augmented, and (d) diverse-augmented. Although the two former training sample sets (a, b) included imbalanced classes, the latter two (c, d) were augmented via over-sampling for ensuring balanced classes. Again, this strategy was adopted to test the impact of an imbalanced dataset on SSL results. Thus, there were four training sets that differed in the sampling strategy. The entire training set was replicated, and each image was segmented, to create the segmented training samples, such that both the raw and segmented training data contained the same images. The classification framework was then parallelly employed for subsequent analysis of any difference in performance. This study hypothesizes a possible improvement in the performance of downstream tasks if much of the noisy background is first removed via segmentation before executing the pre-training methods. The visual difference between the raw and the segmented images is shown (Figure 5) in BGR format, the default format used by the OpenCV library employed for the image pre-processing operations in this study. In the segmented images, much of the background is removed; however, essential visual patterns in the foreground are retained, for example, the bean leaf beetle, and the aphid images are desirably segmented despite very high similarity between the foreground and background.

For segmentation, the local entropy-based (Hržić et al., 2019) masking approach was leveraged to segment an image based on the level of complexity contained in a given neighborhood, defined by the structuring element, disk radius. The entropy filter first detects subtle variations in the local gray level distribution in the defined neighborhood and captures the inherent properties of the transition regions. Image binarization was then performed using a threshold of 0.8 to obtain the mask. On applying the resultant mask to the grayscale image, only those portions of the image that exceeded the threshold were retained. The resultant entropy-masked image was then converted back to the color image format, which now represented the foreground, which was segmented from the background. In this process, for each insect class, the foreground-object texture was selectively segregated using entropy, by varying the disk radius from 5 to 20. Satisfactory segmentation results were empirically achieved for a disk radius of 20 for southern corn rootworm and flea beetle; for stink bug, northern corn rootworm, and flea beetle, it was 15,

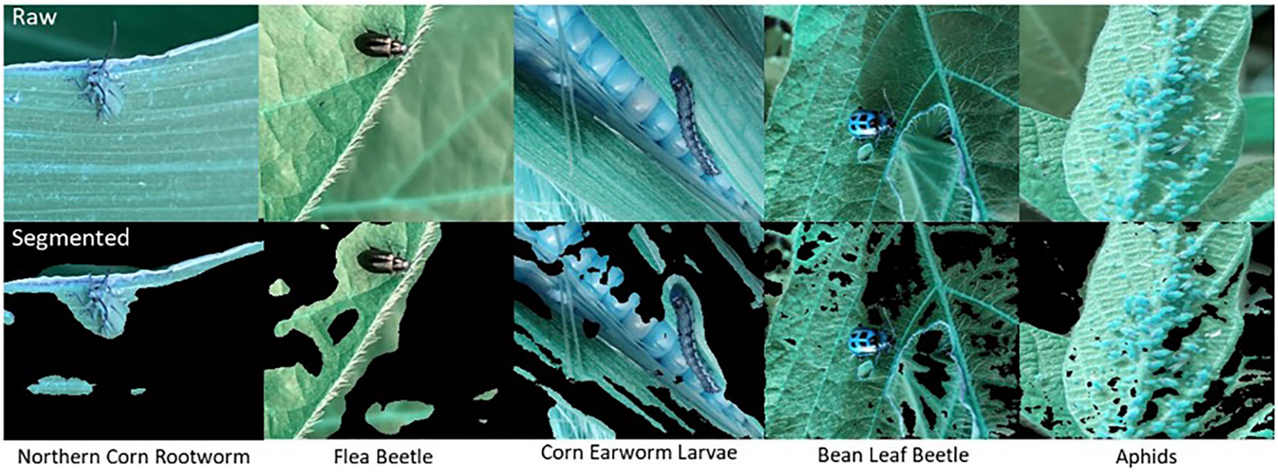


FIGURE 5 Examples of raw (top row) and corresponding segmented (bottom row) images are shown (in BGR format) for specific insect classes, northern corn rootworm, flea beetle, corn earworm larvae, bean leaf beetle, and aphids.

and for the remaining insect classes, 5. Similarly, the threshold for masking was also empirically chosen to be 0.8. This segmentation method was adopted because it takes image texture into account rather than color variations and is simpler and remarkably faster than other reported methods like the Simple Linear Iterative Clustering super pixel segmentation (Stutz, 2015). Thus, once the datasets were prepared, pre-training was performed for 800 epochs by employing SSL methods described above. Two backbone architectures were compared during pre-training, ResNet-18 and ResNet-50, initialized with ImageNet weights (Krizhevsky et al., 2017). The hyperparameters were fine-tuned for each of the methods (Table 1), and the model checkpoint with the lowest training and validation loss was saved for the downstream task. For training optimization, the stochastic gradient descent optimizer was used for each of the experiments, and the models were trained using ReLU activations in the convolutional and dense layers.

2.9 | Linear probing versus end-to-end fine-tuning

We used two different types of evaluation for the SSL methods as shown in Figure 6. To evaluate the transfer of representations, a popular evaluation protocol is to freeze the backbone model and train a linear classifier on the final layer representation (Kolesnikov et al., 2019) as shown in Figure 6a. This method is used to understand the effectiveness of SSL representations for downstream classification. Here, we froze the ResNet backbone model and used the representation from the final layer of the model to train a linear classifier. A linear classifier with 512 nodes was used for the ResNet-18 model, and a linear classifier with 2048 nodes was used for the ResNet-50 model. We used different label fractions of training sets (0.1%, 0.3%, 0.5%, 0.7%, 1%, 3%, 5%, 7%, 10%, 30%, 50%, 70%, and 100%) for the classifier. All the linear probing experiments were repeated three times. We also evaluated the SSL model initializations as shown in Figure 6b. For this, we fine-tuned the model end-to-end using supervised learning. We used different label fractions of training sets (0.1%, 0.3%, 0.5%, 0.7%, 1%, 3%, 5%, 7%, and 10%) for fine-tuning the classifier. Unlike the linear probing evaluation, here we focus on accessing performance when there is a limited budget for labeling (set to 10% of the dataset). All the fine-tuning experiments were repeated three times, and the average results from each method were used to compare between SSL and SL performances.

2.10 | Performance metrics

We calculate the multi-class classification accuracy from the confusion matrix: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). TP and TN are the samples that were correctly classified by the model and are shown on the main diagonal of the confusion matrix. FP and FN are the samples that were incorrectly classified by the model. From these values, the classification accuracy, precision, recall, and $F1$-score are calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

TABLE 1 The values of hyperparameters tuned during pre-training of each self-supervised learning (SSL) model.

Hyperparameter	BYOL	NNCLR	Barlow Twins
num_crops_per_aug	[1, 1, 6]	[1, 1]	[1, 1]
Brightness	[0.4, 0.4, 0.4]	[0.4, 0.4]	[0.4, 0.4]
Contrast	[0.4, 0.4, 0.4]	[0.4, 0.4]	[0.4, 0.4]
Saturation	[0.2, 0.2, 0.2]	[0.2, 0.2]	[0.2, 0.2]
Hue	[0.2, 0.2, 0.2]	[0.1, 0.1]	[0.4, 0.4]
color_jitter_prob	[0.8, 0.8, 0.8]	[0.8, 0.8]	[0.8, 0.8]
gray_scale_prob	[0.0, 0.0, 0.0]	[0.2, 0.2]	[0.2, 0.2]
horizontal_flip_prob	[0.5, 0.5, 0.5]	[0.5, 0.5]	[0.5, 0.5]
gaussian_prob	[0.1, 0.2, 0.3]	[1.0, 0.1]	[1.0, 0.1]
solarization_prob	[0.0, 0.2, 0.4]	[0.0, 0.2]	[0.2, 0.4]
crop_size	[128, 128, 64]	[224, 224]	[128, 128]
min_scale	[0.08, 0.08, 0.08]	[0.08, 0.08]	[0.08, 0.08]
max_scale	[1.0, 1.0, 1.0]	[1.0, 1.0]	[1.0, 1.0]
batch_size	128	128	64
Lr	0.02	0.02	0.01
classifier_lr	0.1	0.3	0.3
weight_decay	1.00E − 05	1.00E − 05	0.0001
Optimizer	sgd	sgd	sgd

Note: The list is as per the hyperparameters provided in the solo-learn library (da Costa et al., 2022).

Abbreviations: BYOL, Bootstrap Your Own Latent; NNCLR, Nearest Neighbor Contrastive Learning of Visual Representations; sgd, stochastic gradient descent.

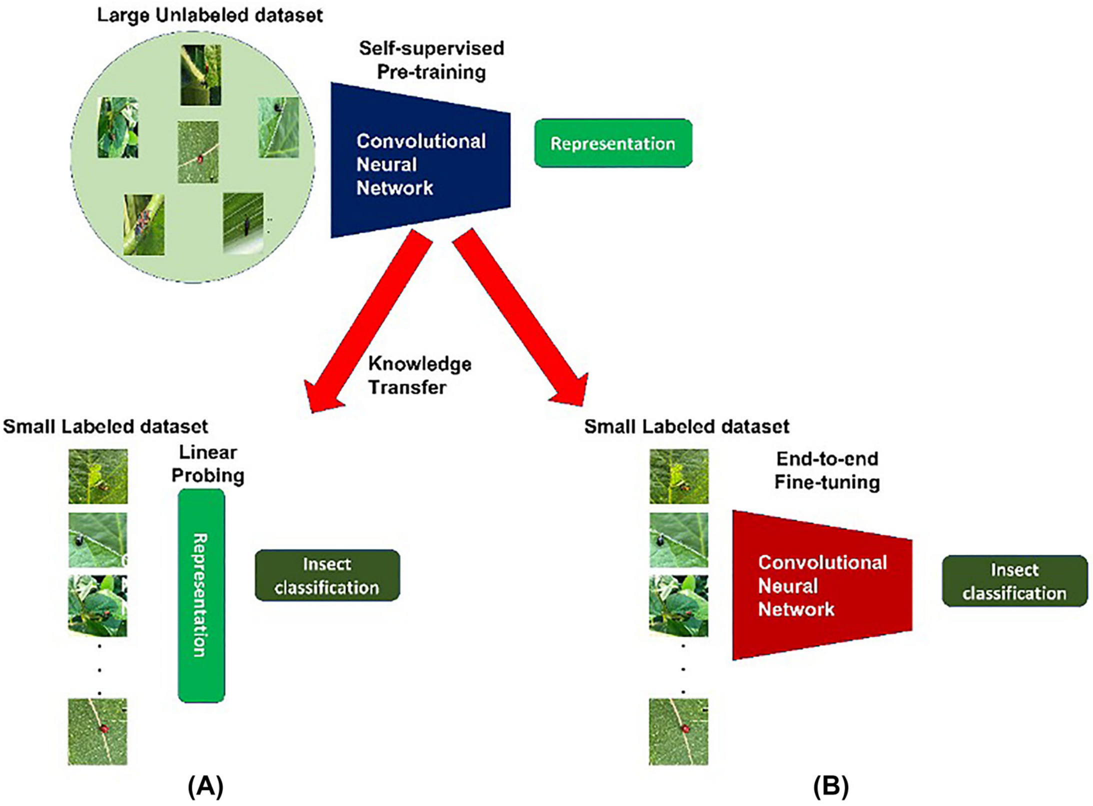


FIGURE 6 Illustration of (A) linear classification and (B) end-to-end fine-tuning methods, which were used to compare the accuracy of self-supervised learning (SSL) methods. In (A), only weights of the last fully connected layer are fine-tuned, and in (B), all model weights are fine-tuned in the end-to-end evaluation.

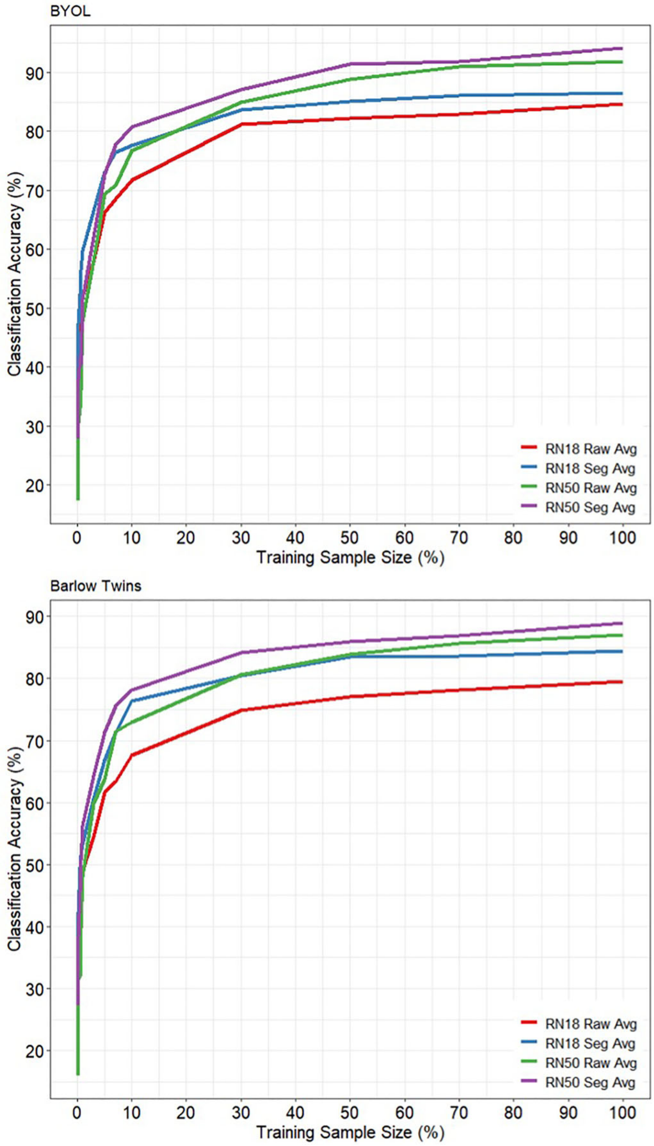


FIGURE 7 The mean (across all four sampling strategies and three repetitions) self-supervised learning (SSL) performance with both ResNet-18 and 50 (RN18/RN50) backbones is plotted for raw and segmented datasets.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = 2\,\frac{precision\ \cdot recall}{precision + recall} \tag{4}$$

3 | RESULTS AND DISCUSSION

3.1 | Linear probing

The overall pre-training results (Figure 7), obtained by taking the mean across all the sampling strategies and three repetitions of each, show that with 100% training data, BYOL achieves the highest classification accuracy (94.16%). The training and validation curves also showed that it rapidly reaches the plateau, and further improvement in performance drastically slows down beyond 200 epochs. This is followed by NNCLR with 93.05%, and then Barlow Twins with 88.98%. NNCLR was the most annotation-efficient method as it reached an accuracy of 90% with just 30% of the training data.

The segmented images helped enhance the pre-training performance compared to the raw images, as expected. With less than 1% labeling of the training data, ∼10%–11% accuracy improvement was observed in the case of segmented images. The highest improvement was noticed in NNCLR. With just 3% sample size and ResNet-18 backbone, segmented images reached an accuracy of 70.87% compared to 58.59% of raw images, with a remarkable increment of 12.27%. As the amount of sampling data increased this difference was reduced, still leading to an average of 3% increment with 100% training samples. This shows that entropy-based image segmentation combined with the NNCLR-SSL method could be a highly annotation-efficient solution with greater than 70% accuracy, even with very low sample size of 3% (i.e., 3% of 10,725 = 322 images in this case).

Currently, there are varied SSL implementations for solving fine-grained image classification problem, for example, semantic learning from the discriminative feature-representations of image parts (Yang et al., 2022; Yu et al., 2022), part-level contrastive learning (Wang et al., 2022), attentively identifying fine-grained images by interaction (Zhuang et al., 2020). However, this study shows the ability of local entropy-mask segmentation in enhancing SSL performance to classify insect pests from complex images, as segmentation helps retain mostly the foreground portions that accentuate the learning of more meaningful representations during the pretext task, compared to the raw images. In the latter case, some of the latent representations could belong to the image background, which is intuitively not very helpful in generalizing the downstream task. Utilizing image segmentation for aiding supervised classification performance has been found to be beneficial in previous studies (Liu, HaoChen, et al., 2021; Mahbod et al., 2020). Additionally, it may be noted that such improvement in model effectiveness was achieved from "local" entropy-mask-based segmentation that may still be influenced by external factors like illumination and occlusion. Hence, as a future research domain, the "locally adaptive" entropy-based thresholding (Zhang et al., 2022), which is rather a computationally expensive approach can be tested to determine the change in performance.

Regarding the backbone architecture, ResNet-50-based experiments yielded a 5%–9% increase in accuracy than the ResNet-18-based experiments, when sample size was 100%. However, when the training size was just 1% or less, ResNet-18-based experiments seemed to achieve an average of ∼3% higher accuracy than the ResNet-50 ones. Such an effect was prominent in the BYOL and Barlow Twins methods. However, in the case of NNCLR, ResNet-50 proved beneficial across all the sample sizes with a 4% increase in accuracy on an average, both on raw and segmented datasets. This states that when the training size is extremely low, simpler architectures are better for information maximization or distillation-based SSL methods. However, based on the overall result from the comparison between the backbone architectures, the sampling strategies were examined for the ResNet-50-based models (Figure 8).

There was no improvement noticed with the augmented dataset containing balanced classes, on any of the three SSL methods. It was observed that classification accuracy rather dropped with diverse-augmented samples, particularly if the proportion of labeled samples in the training set was less than 10%. This could have potentially resulted from over-sampling that led to overfitting for specific classes, where the model tried to learn all the data points including noise and inaccurate values present in the dataset, thereby reducing model accuracy (Santos et al., 2018). There were very minor to no differences noticed in the performance between the random and diverse sampling strategies. In addition, in the case of random sampling, results from both the imbalanced (raw) and balanced (raw-augmented) datasets were almost similar with no noticeable difference. Thus, these findings confirmed that SSL methods are robust to class imbalance, also suggested in Liu, Zhang et al. (2021), and these methods can achieve better performance with segmented images. Therefore, the subsequent results demonstrate the performance difference between linear probing and fine-tuning, based only on the randomly sampled segmented images, and do not include the diverse and augmented cases.

3.2 | Fine-tuning evaluation

Figure 9 shows the performance of end-to-end fine-tuning results of ResNet-18 and ResNet-50 models. All the fine-tuning experiments were repeated three times, and the mean classification accuracy across the three repetitions is shown in Figure 9a,b. NNCLR was the best performing SSL method. For 5% of the labeled samples, the NNCLR method obtained a mean classification accuracy of ∼79% for the ResNet-50 model and an accuracy of ∼74% for the ResNet-18 model. All the SSL pre-training methods outperformed supervised baseline for end-to-end fine-tuning evaluation. The SSL pre-training methods were more annotation efficient than ImageNet initialization for training fractions less than 5%. The performance of ImageNet initialization was on-par with SSL methods for training fractions greater than 5%. These results were as expected because evidence suggests that the benefit of SSL models increases with the availability of larger amounts of unlabeled data for pre-training. Among the SSL methods, Barlow Twins had the lowest performance. For 10% training data, the ResNet-50 model obtained a mean classification accuracy of 86% and was ∼4% better than the ResNet-18 model.

Figures 10 and 11 show the confusion matrices of the ResNet-50 model with ImageNet and NNCLR initializations, respectively. The model was trained with 7% of labeled data, and the input images were pre-processed with entropy-based segmentation. For confounding classes, like bean leaf beetle and ladybird beetle, the NNCLR model performed better than ImageNet initialization. The NNCLR initialization obtained an accuracy of 96% for bean leaf beetle, whereas the ImageNet model obtained an accuracy of 78%. Similarly, for the confounding classes like FAW and corn earworm larvae, the NNCLR model obtained accuracies of 97% and 90%, respectively, whereas the ImageNet model obtained accuracies of 89% and 92%, respectively.

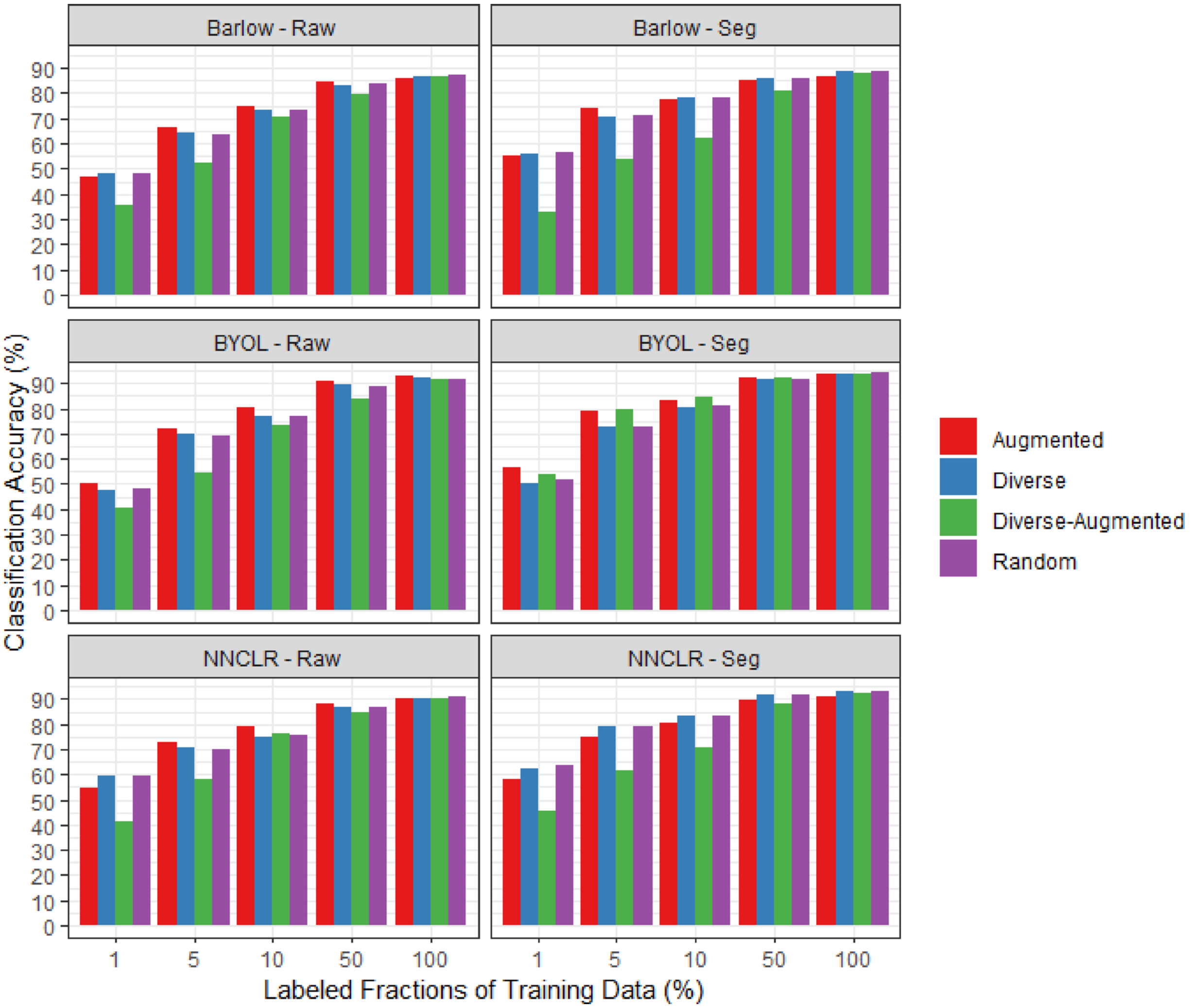


FIGURE 8 Comparison of the impact of different sampling strategies on each of the self-supervised learning (SSL) methods. For brevity, the results are plotted for sample sizes of 1%, 5%, 10%, 50%, and 100%, which potentially capture the overall pattern of improvement in classification accuracy as the sample size increases.

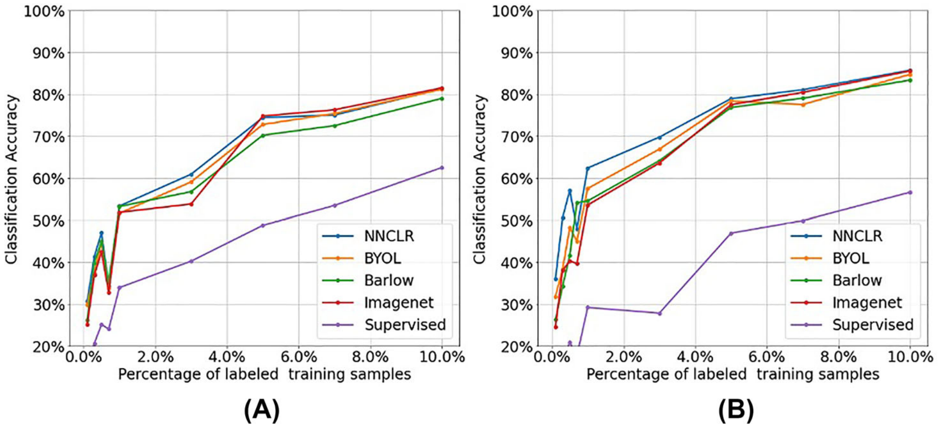


FIGURE 9 End-to-end fine-tuning evaluation of (A) ResNet-18 and (B) ResNet-50 models using segmented images. The "Supervised" curve corresponds to training from random initialization. The models were fine-tuned for different label percentage fractions (0.1%, 0.3%, 0.5%, 0.7%, 1%, 3%, 5%, 7%, and 10%).

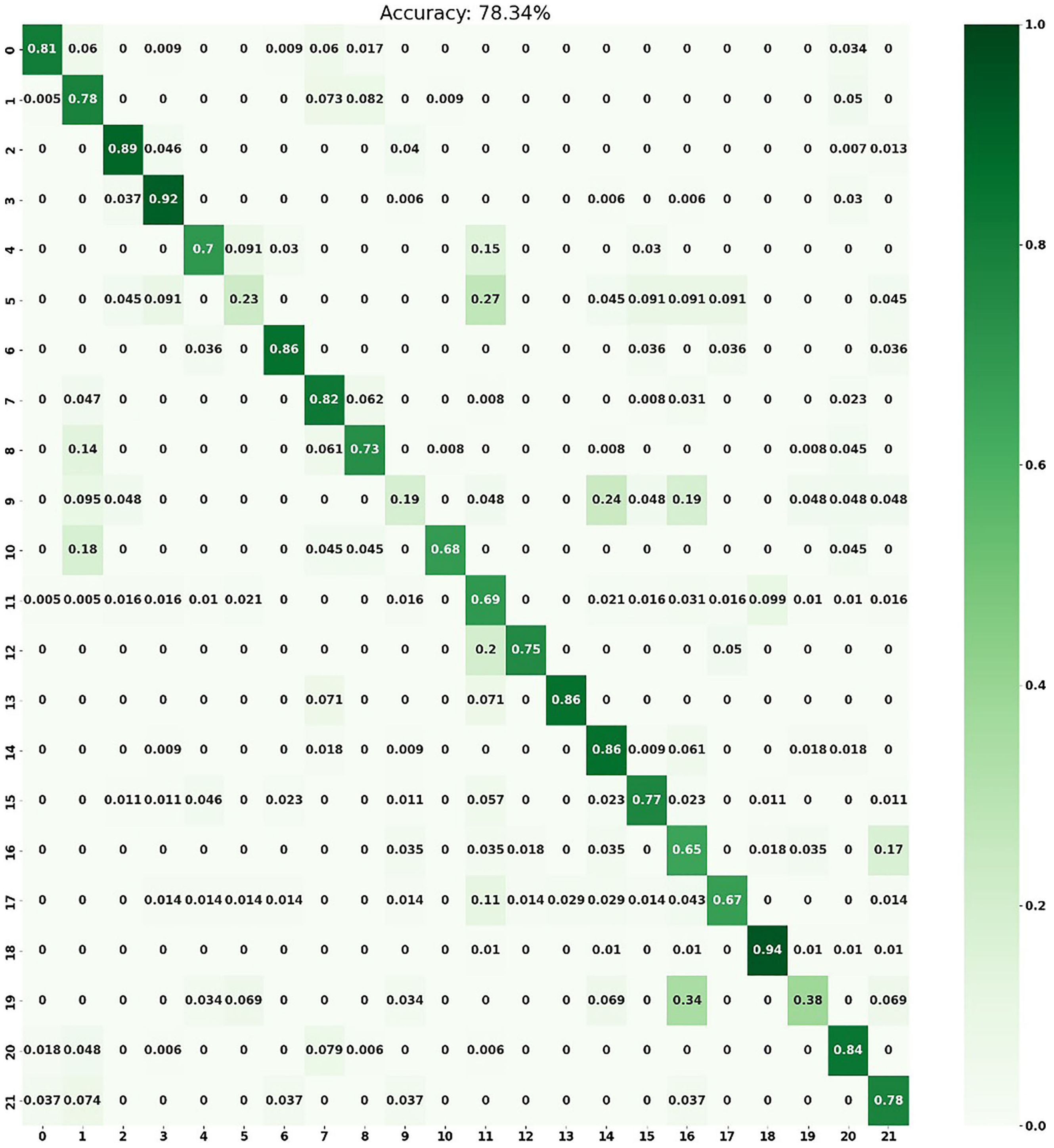
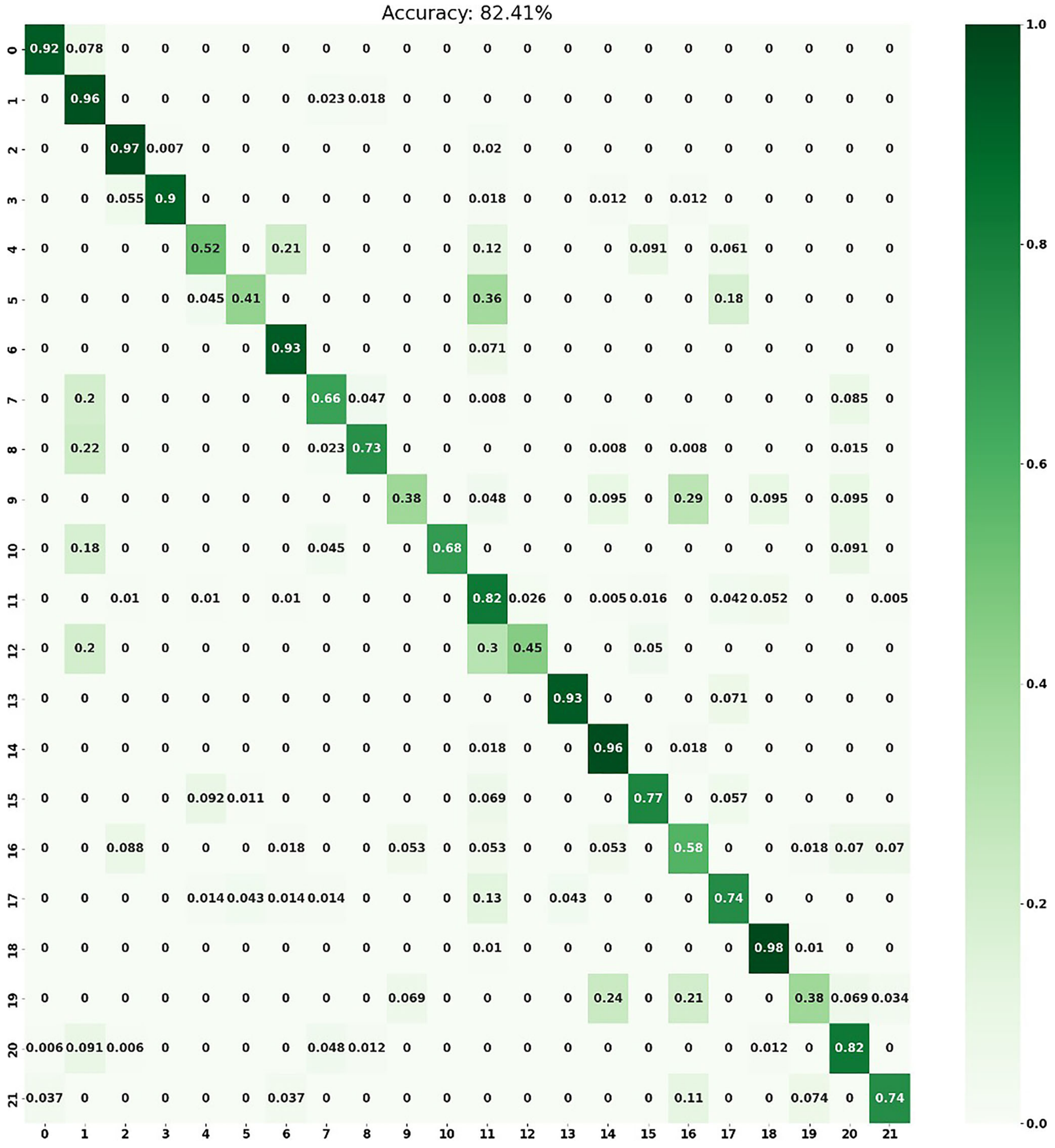


FIGURE 10 Confusion matrix for ImageNet initialized ResNet-50. The model was trained with 7% of labeled images. The input images were pre-processed with entropy-based segmentation for removing the background. The 22 classes are “Aphids”: 0, “Bean leaf beetle”: 1, “Corn earworm larvae”: 2, “Fall armyworm”: 3, “Flea beetle”: 4, “Green lacewing”: 5, “Green leaf hopper”: 6, “Japanese beetle”: 7, “Ladybird beetle”: 8, “Maize calligrapher”: 9, “Milkweed bug”: 10, “Northern corn rootworm beetle”: 11, “Sap beetle”: 12, “Silver spotted caterpillars”: 13, “Soldier beetle”: 14, “Southern corn rootworm beetle”: 15, “Soybean nodule fly”: 16, “Stink bug”: 17, “Striped cucumber beetle”: 18, “Tarnished plant bug”: 19, “Western corn rootworm beetle”: 20, “White fly”: 21.



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
0	0.92	0.078	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0.96	0	0	0	0	0	0.023	0.018	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0.97	0.007	0	0	0	0	0	0	0	0.02	0	0	0	0	0	0	0	0	0	0
3	0	0	0.055	0.9	0	0	0	0	0	0	0	0.018	0	0	0.012	0	0.012	0	0	0	0	0
4	0	0	0	0	0.52	0	0.21	0	0	0	0	0.12	0	0	0	0.091	0	0.061	0	0	0	0
5	0	0	0	0	0.045	0.41	0	0	0	0	0	0.36	0	0	0	0	0	0.18	0	0	0	0
6	0	0	0	0	0	0	0.93	0	0	0	0	0.071	0	0	0	0	0	0	0	0	0	0
7	0	0.2	0	0	0	0	0	0.66	0.047	0	0	0.008	0	0	0	0	0	0	0	0	0.085	0
8	0	0.22	0	0	0	0	0	0.023	0.73	0	0	0	0	0	0.008	0	0.008	0	0	0	0.015	0
9	0	0	0	0	0	0	0	0	0	0.38	0	0.048	0	0	0.095	0	0.29	0	0.095	0	0.095	0
10	0	0.18	0	0	0	0	0	0.045	0	0	0.68	0	0	0	0	0	0	0	0	0	0.091	0
11	0	0	0.01	0	0.01	0	0.01	0	0	0	0	0.82	0.026	0	0.005	0.016	0	0.042	0.052	0	0	0.005
12	0	0.2	0	0	0	0	0	0	0	0	0	0.3	0.45	0	0	0.05	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0.93	0	0	0	0.071	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0.018	0	0	0.96	0	0.018	0	0	0	0	0
15	0	0	0	0	0.092	0.011	0	0	0	0	0	0.069	0	0	0	0.77	0	0.057	0	0	0	0
16	0	0	0.088	0	0	0	0.018	0	0	0.053	0	0.053	0	0	0.053	0	0.58	0	0	0.018	0.07	0.07
17	0	0	0	0	0.014	0.043	0.014	0.014	0	0	0	0.13	0	0.043	0	0	0	0.74	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0.01	0	0	0	0	0	0	0.98	0.01	0	0
19	0	0	0	0	0	0	0	0	0	0.069	0	0	0	0	0.24	0	0.21	0	0	0.38	0.069	0.034
20	0.006	0.091	0.006	0	0	0	0	0.048	0.012	0	0	0	0	0	0	0	0	0	0.012	0	0.82	0
21	0.037	0	0	0	0	0	0.037	0	0	0	0	0	0	0	0	0	0.11	0	0	0.074	0	0.74

FIGURE 11 Confusion matrix for Nearest Neighbor Contrastive Learning of Visual Representations (NNCLR) initialized ResNet-50 model trained on segmented images. The model was trained with 7% of labeled images. The input images were pre-processed with entropy-based segmentation for removing the background. The 22 classes are "Aphids": 0, "Bean leaf beetle": 1, "Corn earworm larvae": 2, "Fall armyworm": 3, "Flea beetle": 4, "Green lace wing": 5, "Green leaf hopper": 6, "Japanese beetle": 7, "Ladybird beetle": 8, "Maize calligrapher": 9, "Milkweed bug": 10, "Northern corn rootworm beetle": 11, "Sap beetle": 12, "Silver spotted caterpillars": 13, "Soldier beetle": 14, "Southern corn rootworm beetle": 15, "Soybean nodule fly": 16, "Stink bug": 17, "Striped cucumber beetle": 18, "Tarnished plant bug": 19, "Western corn rootworm beetle": 20, "White fly": 21.

TABLE 2 Precision obtained for each of the 22 classes at 5%, 7%, and 10% proportions of training data, from the ImageNet and Nearest Neighbor Contrastive Learning of Visual Representations (NNCLR) models.

	Precision					
	5p		7p		10p	
	ImageNet	NNCLR	ImageNet	NNCLR	ImageNet	NNCLR
Aphids: 0	0.926	0.955	0.958	0.987	0.990	0.999
Bean leaf beetle: 1	0.546	0.503	0.696	0.653	0.728	0.685
Corn earworm larvae: 2	0.850	0.859	0.925	0.934	0.957	0.966
Fall armyworm: 3	0.820	0.992	0.825	0.997	0.857	0.999
Flea beetle: 4	0.833	0.764	0.888	0.819	0.920	0.871
Green lace wing: 5	0.541	0.884	0.556	0.899	0.676	0.991
Green leaf hopper: 6	0.884	0.763	0.909	0.863	0.941	0.955
Japanese beetle: 7	0.668	0.812	0.743	0.887	0.863	0.919
Ladybird beetle: 8	0.775	0.905	0.820	0.950	0.852	0.982
Maize calligrapher: 9	0.485	0.757	0.520	0.792	0.640	0.824
Milkweed bug: 10	0.976	1.000	0.976	1.000	0.978	1.000
Northern corn rootworm beetle: 11	0.417	0.401	0.467	0.451	0.787	0.483
Sap beetle: 12	0.959	0.945	0.974	0.960	0.976	0.992
Silver spotted caterpillars: 13	0.967	0.956	0.972	0.981	0.974	0.983
Soldier beetle: 14	0.639	0.699	0.789	0.849	0.841	0.881
Southern corn rootworm beetle: 15	0.753	0.831	0.828	0.906	0.860	0.938
Soybean nodule fly: 16	0.430	0.472	0.505	0.547	0.625	0.579
Stink bug: 17	0.776	0.643	0.871	0.738	0.903	0.770
Striped cucumber beetle: 18	0.880	0.860	0.935	0.915	0.967	0.947
Tarnished plant bug: 19	0.747	0.788	0.812	0.853	0.932	0.885
Western corn rootworm beetle: 20	0.724	0.659	0.789	0.724	0.821	0.756
White fly: 21	0.644	0.872	0.699	0.947	0.731	0.979

The precision, recall, and $F1$-scores are presented in Tables 2–4. Overall, the NNCLR model yielded 4.9%, 5.43%, and 2.56% better precision than the ImageNet model with 5%, 10%, and 10% labeling of the training samples, respectively. Similarly, the NNCLR model's recall was higher by 2.07%, 4.12%, and 2.0%, whereas the $F1$-score improved by 2.46%, 4.07%, and 0.52% for 5%, 7%, and 10% labeled fractions of the training set. Nevertheless, it was interesting to note that some classes, for example, bean leaf beetle, northern corn rootworm beetle, and stink bug, could be classified with better precision by the ImageNet, while the corresponding recall scores from the NNCLR model were higher. This implied that the NNCLR model produced fewer FN, that is, it was better at identifying both positive and negative samples of the classes with high intra-class variability like the bean leaf beetle, and the northern corn rootworm beetle that is tan to pale green in color and easily camouflages with the background in the field. Considering all the three sampling scenarios, the NNCLR-based recall for the northern corn rootworm was 12% higher than that of ImageNet. Contrarily, western corn rootworm beetle was the only class for which the ImageNet classifier performed better in all the three metrics, with a mean increase of ~6% (precision), 2% (recall), and 5% ($F1$-score) across the three scenarios with 5%, 7%, and 10% labeled data. However, for the minority classes like the green lacewing, and the maize calligrapher, NNCLR performed remarkably better. In the case of green lacewing, precision and recall were higher by 33.3% and 22.3%, whereas for maize calligrapher, the respective scores were up by 24.3% and 15%. Another notable example demonstrating the efficiency of the SSL-pre-trained model in correctly classifying a confounding class is that of the southern corn rootworm beetle (with ~8% higher precision, recall, and $F1$-score), which looks very similar to a bean leaf beetle (Figure 3b-ii).

These classification results show that the NNCLR model that was trained on smaller in-domain unlabeled data was able to obtain good accuracy for challenging classes with few labels compared to ImageNet model that was pre-trained on

TABLE 3 Recall obtained for each of the 22 classes at 5%, 7%, and 10% proportions of training data, from the ImageNet and Nearest Neighbor Contrastive Learning of Visual Representations (NNCLR) models.

	Recall					
	5p		**7p**		**10p**	
	ImageNet	**NNCLR**	**ImageNet**	**NNCLR**	**ImageNet**	**NNCLR**
Aphids: 0	0.811	0.922	0.836	0.947	0.891	0.972
Bean leaf beetle: 1	0.781	0.959	0.806	0.984	0.861	0.999
Corn earworm larvae: 2	0.894	0.973	0.919	0.988	0.944	0.993
Fall armyworm: 3	0.915	0.903	0.920	0.958	0.945	0.983
Flea beetle: 4	0.699	0.519	0.849	0.669	0.944	0.819
Green lace wing: 5	0.230	0.412	0.480	0.662	0.505	0.812
Green leaf hopper: 6	0.857	0.929	0.902	0.974	0.927	0.989
Japanese beetle: 7	0.821	0.660	0.846	0.910	0.941	0.965
Ladybird beetle: 8	0.730	0.742	0.805	0.817	0.955	0.842
Maize calligrapher: 9	0.189	0.379	0.264	0.454	0.414	0.479
Milkweed bug: 10	0.683	0.683	0.883	0.883	0.938	0.908
Northern corn rootworm beetle: 11	0.691	0.823	0.791	0.923	0.846	0.948
Sap beetle: 12	0.750	0.450	0.845	0.545	0.900	0.570
Silver spotted caterpillars: 13	0.858	0.929	0.923	0.974	0.948	0.999
Soldier beetle: 14	0.858	0.964	0.898	0.994	0.923	0.999
Southern corn rootworm beetle: 15	0.772	0.771	0.797	0.921	0.822	0.946
Soybean nodule fly: 16	0.653	0.578	0.718	0.633	0.868	0.658
Stink bug: 17	0.675	0.741	0.805	0.834	0.900	0.859
Striped cucumber beetle: 18	0.940	0.980	0.953	0.993	0.978	0.998
Tarnished plant bug: 19	0.382	0.379	0.455	0.452	0.805	0.547
Western corn rootworm beetle: 20	0.837	0.824	0.902	0.889	0.957	0.914
White fly: 21	0.778	0.741	0.793	0.891	0.888	0.946

large, labeled data from out-of-domain. This showed that SSL could solve fine-grained inter- and intra-class classification problems, because the bean leaf beetle class contained the high intra-class variability, whereas the confounding classes had fine-grained inter-class variability. As the proportion of labeled samples increased from 5% to 10%, the recall or the ability of the SSL method in correctly identifying the bean leaf beetle images increased from 95.9% to 99.9%, compared to a recall of 0.861 by the ImageNet model, when trained with just 10% labeled samples. Similar patterns in the results were also observed in the case of confounding classes like green lace wing and the green leaf hopper, also identified as one of the minority classes in the dataset. Aphids is another class with high fine-grained variability, which could be classified with 92% accuracy with 7% training using SSL, whereas the ImageNet method's accuracy was 11% lower.

Such robustness of SSL to dataset imbalance could be attributed to its ability to learn richer features that are transferable across layers to help classify the rare classes and downstream tasks (Liu, Zhang, et al., 2021; Yang & Xu, 2020). More specifically, SSL is not actuated by any labels, unlike the SL approach. Hence, SSL is not limited to learning only the label-relevant features that help predict the frequent classes, but rather a diverse set of generalizable representations, including both label-relevant and irrelevant features from unlabeled data. Learning during the pretext task also contributes to the representation-invariance property of an SSL model (Tendle & Hasan, 2021), such that it captures the ingrained characteristics of the input distribution, that are generalizable or transferable to downstream tasks. Therefore, SSL methods can generalize to rare classes better than SL approaches. SSL's robustness to class imbalance is thoroughly demonstrated by Liu, Zhang et al. (2021), and the generalizability of self-supervised representations is discussed by Tendle and Hasan (2021).

Overall, the SSL methods provide an exciting opportunity and application in the plant science domain. At the same time, there are several open questions that require future

TABLE 4 *F*1-Score obtained for each of the 22 classes at 5%, 7%, and 10% proportions of training data, from the ImagNet and Nearest Neighbor Contrastive Learning of Visual Representations (NNCLR) models.

	***F*1-Score**					
	5p		**7p**		**10p**	
	ImageNet	**NNCLR**	**ImageNet**	**NNCLR**	**ImageNet**	**NNCLR**
Aphids: 0	0.864	0.938	0.893	0.967	0.938	0.985
Bean leaf beetle: 1	0.643	0.660	0.747	0.785	0.789	0.813
Corn earworm larvae: 2	0.871	0.913	0.922	0.960	0.950	0.979
Fall armyworm: 3	0.865	0.945	0.870	0.977	0.899	0.991
Flea beetle: 4	0.760	0.618	0.868	0.736	0.932	0.844
Green lace wing: 5	0.323	0.562	0.515	0.762	0.578	0.892
Green leaf hopper: 6	0.870	0.838	0.905	0.915	0.934	0.972
Japanese beetle: 7	0.737	0.728	0.791	0.898	0.900	0.941
Ladybird beetle: 8	0.752	0.815	0.812	0.878	0.901	0.906
Maize calligrapher: 9	0.272	0.505	0.351	0.577	0.503	0.606
Milkweed bug: 10	0.804	0.811	0.927	0.938	0.958	0.952
Northern corn rootworm beetle: 11	0.520	0.539	0.587	0.606	0.816	0.640
Sap beetle: 12	0.842	0.610	0.905	0.695	0.936	0.724
Silver spotted caterpillars: 13	0.910	0.942	0.947	0.977	0.961	0.991
Soldier beetle: 14	0.733	0.810	0.840	0.916	0.880	0.936
Southern corn rootworm beetle: 15	0.763	0.800	0.813	0.913	0.841	0.942
Soybean nodule fly: 16	0.518	0.520	0.593	0.587	0.726	0.616
Stink bug: 17	0.722	0.689	0.837	0.783	0.902	0.812
Striped cucumber beetle: 18	0.909	0.916	0.944	0.953	0.973	0.972
Tarnished plant bug: 19	0.505	0.512	0.583	0.591	0.864	0.676
Western corn rootworm beetle: 20	0.777	0.732	0.842	0.798	0.884	0.827
White fly: 21	0.705	0.801	0.743	0.918	0.802	0.962

research. The SSL-based insect-pest identification should investigate (a) designing pretext classes specifically to insect-pest classification, (b) using class-specific loss functions, (c) pre-training with both out-of-domain and in-domain data, and (d) developing a mobile application for farmers and breeders.

4 | CONCLUSIONS

This paper presents an IA insect-pest dataset that generates exciting opportunities for researchers and practitioners to utilize the dataset in ML model development. This dataset includes (a) several classes with large intra-class variability in size, shape, color, patterns, and texture; (b) insects from different classes that look similar; (c) high class imbalance; (d) large background noise compared to the insect or the foreground; (e) varying illumination conditions and shadows; (f) overlapping objects in the image; (g) multiple insect-pest species in the same image frame. Using this insect-pest dataset, we thoroughly investigated different SSL methods, with and without prior image segmentation, to circumvent data annotation challenges that plague plant scientists as biological systems are inherently very complex. We found that SSL-pre-trained models were annotation efficient for insect-pest classification. For learning with few labels, the model initializations and latent representation from NNCLR was better than the ImageNet model. Pre-training with segmented input images provided better performance than the original images. All the SSL methods performed better than the supervised baseline for both linear probing and end-to-end evaluation. The SSL-pre-trained models were robust to class imbalances and were able to differentiate confounding insect classes. These results indicate the usefulness of SSL methods, especially with segmented images for data labeling/annotation challenges to save time, cost, physical resource, computation, and integrate phone-based imaging with ML pipeline that can work across geographies to help identify and eventually control insect pests in the field. SSL models from our paper will be efficient in solving a variety of plant phenomics problems, which includes the early detection of insect pests, species identification, damage assessment, yield loss due to insect infestation, and provide vital information to farming community to maintain a healthy crop cycle.

AUTHOR CONTRIBUTIONS

Soumyashree Kar: Conceptualization; data curation; formal analysis; investigation; methodology; validation; writing—original draft; writing—review and editing. **Koushik Nagasubramanian**: Data curation; formal analysis; investigation; methodology; validation; writing—original draft. **Dinakaran Elango**: Data curation; investigation; methodology; writing—original draft; writing—review and editing. **Matthew E. Carroll**: Writing—review and editing. **Craig A. Abel**: Funding acquisition; methodology; resources; writing—review and editing. **Ajay Nair**: Investigation; resources; writing—review and editing. **Daren S. Mueller**: Investigation; resources; writing—review and editing. **Matthew E. O'Neal**: Investigation; resources; writing—review and editing. **Asheesh K. Singh**: Conceptualization; investigation; methodology; resources; validation; writing—review and editing. **Soumik Sarkar**: Funding acquisition; methodology; resources; writing—review and editing. **Baskar Ganapathysubramanian**: Conceptualization; data curation; funding acquisition; investigation; methodology; project administration; resources; supervision; validation; visualization; writing—review and editing. **Arti Singh**: Conceptualization; data curation; formal analysis; funding acquisition; investigation; methodology; project administration; resources; supervision; validation; writing—original draft; writing—review and editing.

ACKNOWLEDGMENTS

We thank team members and collaborators of Arti Singh and Asheesh K Singh's group for their help in imaging and data collection.

Open access funding provided by the Iowa State University Library.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

DATA AVAILABILITY STATEMENT

Raw dataset and the Python code used for analysis can be accessed at https://github.com/SoylabSingh/Insect1.

ORCID

Soumyashree Kar https://orcid.org/0000-0003-2158-2540
Dinakaran Elango https://orcid.org/0000-0003-2226-486X
Asheesh K. Singh https://orcid.org/0000-0002-7522-037X
Arti Singh https://orcid.org/0000-0001-6191-9238

REFERENCES

Agastya, C., Ghebremusse, S., Anderson, I., Reed, C., Vahabi, H., & Todeschini, A. (2021). Self-supervised contrastive learning for irrigation detection in satellite imagery. *ArXiv Preprint*.

Ahmad, I., Yang, Y., Yue, Y., Ye, C., Hassan, M., Cheng, X., Wu, Y., & Zhang, Y. (2022). Deep learning based detector YOLOv5 for identifying insect pests. *Applied Sciences*, *12*(19), 10167. https://doi.org/10.3390/app121910167

Azimi, S., Kaur, T., & Gandhi, T. K. (2021). A deep learning approach to measure stress level in plants due to Nitrogen deficiency. *Measurement*, *173*, 108650. https://doi.org/10.1016/j.measurement.2020.108650

Bah, M., Hafiane, A., & Canals, R. (2018). Deep learning with unsupervised data labeling for weed detection in line crops in UAV images. *Remote Sensing*, *10*(11), 1690. https://doi.org/10.3390/rs10111690

Bahtiar, A. R., Pranowo, P., Santos, A. J., & Juhariah, J. (2020). Deep learning detected nutrient deficiency in chili plant. *2020 8th International Conference on Information and Communication Technology (ICoICT)* (pp. 1–4). IEEE. https://doi.org/10.1109/ICoICT49345.2020.9166224

Barbedo, J. G. A. (2019). Detection of nutrition deficiencies in plants using proximal images and machine learning: A review. *Computers and Electronics in Agriculture*, *162*, 482–492. https://doi.org/10.1016/j.compag.2019.04.035

Bereciartua-Pérez, A., Gómez, L., Picón, A., Navarra-Mestre, R., Klukas, C., & Eggers, T. (2022). Insect counting through deep learning-based density maps estimation. *Computers and Electronics in Agriculture*, *197*, 106933. https://doi.org/10.1016/j.compag.2022.106933

Bortolato, B., Smolkovič, A., Dillon, B. M., & Kamenik, J. F. (2022). Bump hunting in latent space. *Physical Review D*, *105*(11), 115009. https://doi.org/10.1103/PhysRevD.105.115009

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., & Joulin, A. (2021). Emerging properties in self-supervised vision transformers. *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 9650–9660). IEEE.

Chen, H., Chen, A., Xu, L., Xie, H., Qiao, H., Lin, Q., & Cai, K. (2020). A deep learning CNN architecture applied in smart near-infrared analysis of water pollution for agricultural irrigation resources. *Agricultural Water Management*, *240*, 106303. https://doi.org/10.1016/j.agwat.2020.106303

da Costa, V. G. T., Fini, E., Nabi, M., Sebe, N., & Ricci, E. (2022). Solo-learn: A library of self-supervised methods for visual representation learning. *Journal of Machine Learning Research*, *23*, 1–6.

dos Santos Ferreira, A., Matte Freitas, D., Gonçalves da Silva, G., Pistori, H., & Theophilo Folhes, M. (2017). Weed detection in soybean crops using ConvNets. *Computers and Electronics in Agriculture*, *143*, 314–324. https://doi.org/10.1016/j.compag.2017.10.027

Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., & Zisserman, A. (2021). With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 9588–9597). IEEE.

Gazdic, M., & Groom, Q. (2019). iNaturalist is an unexploited source of plant-insect interaction data. *Biodiversity Information Science and Standards*, *3*, e37303. https://doi.org/10.3897/biss.3.37303

Ghosal, S., Blystone, D., Singh, A. K., Ganapathysubramanian, B., Singh, A., & Sarkar, S. (2018). An explainable deep machine vision framework for plant stress phenotyping. *Proceedings of the National Academy of Sciences of the United States of America*, *115*(18), 4613–4618. https://doi.org/10.1073/pnas.1716999115

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., Piot, B., Kavukcuoglu, K., Munos, R., & Valko, M. (2020). Bootstrap your own latent: A new approach to self-supervised Learning. *Advances in Neural Information Processing Systems*, *33*, 21271–21284.

Gullino, M., Albajes, R., Al-Jboory, I., Angelotti, F., Chakraborty, S., Garrett, K., Hurley, B., Juroszek, P., Makkouk, K., Pan, X., & Stephenson, T. (2021). *Scientific review of the impact of climate change on plant pests*. FAO on behalf of the IPPC Secretariat. https://doi.org/10.4060/cb4769en

Hao, G.-F., Zhao, W., & Song, B.-A. (2020). Big data platform: An emerging opportunity for precision pesticides. *Journal of Agricultural and Food Chemistry*, *68*(41), 11317–11319. https://doi.org/10.1021/acs.jafc.0c05584

Hržić, F., Štajduhar, I., Tschauner, S., Sorantin, E., & Lerga, J. (2019). Local-entropy based approach for X-ray image segmentation and fracture detection. *Entropy*, *21*(4), 338. https://doi.org/10.3390/e21040338

Jubery, T. Z., Carley, C. N., Singh, A., Sarkar, S., Ganapathysubramanian, B., & Singh, A. K. (2021). Using machine learning to develop a fully automated Soybean Nodule Acquisition Pipeline (SNAP). *Plant Phenomics*, *2021*, 9834746. https://doi.org/10.34133/2021/9834746

Kahn, G., Abbeel, P., & Levine, S. (2021). BADGR: An autonomous self-supervised learning-based navigation system. *IEEE Robotics and Automation Letters*, *6*(2), 1312–1319. https://doi.org/10.1109/LRA.2021.3057023

Kolesnikov, A., Zhai, X., & Beyer, L. (2019). Revisiting self-supervised visual representation learning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 1920–1929). IEEE.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90. https://doi.org/10.1145/3065386

Kulkarni, O. (2018). Crop disease detection using deep learning. *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)* (pp. 1–4). IEEE. https://doi.org/10.1109/ICCUBEA.2018.8697390

Li, W., Chen, P., Wang, B., & Xie, C. (2019). Automatic localization and count of agricultural crop pests based on an improved deep learning pipeline. *Scientific Reports*, *9*(1), 7024. https://doi.org/10.1038/s41598-019-43171-0

Li, W., Zheng, T., Yang, Z., Li, M., Sun, C., & Yang, X. (2021). Classification and detection of insects from field images using deep learning for smart pest management: A systematic review. *Ecological Informatics*, *66*, 101460. https://doi.org/10.1016/j.ecoinf.2021.101460

Liebhold, A., & Bentz, B. (2011). *Insect disturbance and climate change*. USDA Forest Service, Climate Change Resource Center. www.fs.usda.gov/ccrc/topics/insectdisturbance/insect-disturbance

Liu, H., HaoChen, J. Z., Gaidon, A., & Ma, T. (2021). Self-supervised learning is more robust to dataset imbalance. *arXiv preprint*.

Liu, J., & Wang, X. (2021). Plant diseases and pests detection based on deep learning: A review. *Plant Methods*, *17*, 22. https://doi.org/10.1186/s13007-021-00722-9

Liu, Y., Zhang, Z., Liu, X., Wang, L., & Xia, X. (2021). Efficient image segmentation based on deep learning for mineral image classification. *Advanced Powder Technology*, *32*(10), 3885–3903.

Mahbod, A., Tschandl, P., Langs, G., Ecker, R., & Ellinger, I. (2020). The effects of skin lesion segmentation on the performance of dermatoscopic image classification. *Computer Methods and Programs in Biomedicine*, *197*, 105725.

Margapuri, V., & Neilsen, M. (2021). Classification of seeds using domain randomization on self-supervised learning frameworks. *2021 IEEE Symposium Series on Computational Intelligence (SSCI)* (pp. 01–08). IEEE. https://doi.org/10.1109/SSCI50451.2021.9659998

Masood, A., Al-Jumaily, A., & Anam, K. (2015). Self-supervised learning model for skin cancer diagnosis. *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)* (pp. 1012–1015). IEEE. https://doi.org/10.1109/NER.2015.7146798

Misra, I., & van der Maaten, L. (2020). Self-supervised learning of pretext-invariant representations. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 6707–6717). IEEE.

Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in Plant Science*, *7*, 1419. https://doi.org/10.3389/fpls.2016.01419

Nagasubramanian, K., Jubery, T., Fotouhi Ardakani, F., Mirnezami, S. V., Singh, A. K., Singh, A., Sarkar, S., & Ganapathysubramanian, B. (2021). How useful is active learning for image-based plant phenotyping? *The Plant Phenome Journal*, *4*(1), e20020. https://doi.org/10.1002/ppj2.20020

Nagasubramanian, K., Singh, A. K., Singh, A., Sarkar, S., & Ganapathysubramanian, B. (2022). Plant phenotyping with limited annotation: Doing more with less. *The Plant Phenome Journal*, *5*, e20051. https://doi.org/10.1002/ppj2.20051

Nanni, L., Manfè, A., Maguolo, G., Lumini, A., & Brahnam, S. (2022). High performing ensemble of convolutional neural networks for insect pest image detection. *Ecological Informatics*, *67*, 101515. https://doi.org/10.1016/j.ecoinf.2021.101515

Osorio, K., Puerto, A., Pedraza, C., Jamaica, D., & Rodríguez, L. (2020). A deep learning approach for weed detection in lettuce crops using multispectral images. *AgriEngineering*, *2*(3), 471–488. https://doi.org/10.3390/agriengineering2030032

Rairdin, A., Fotouhi, F., Zhang, J., Mueller, D. S., Ganapathysubramanian, B., Singh, A. K., Dutta, S., Sarkar, S., & Singh, A. (2022). Deep learning-based phenotyping for genome wide association studies of sudden death syndrome in soybean. *Frontiers in Plant Science*, *13*, 966244. https://doi.org/10.3389/fpls.2022.966244

Rangarajan, A. K., Purushothaman, R., & Ramesh, A. (2018). Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia Computer Science*, *133*, 1040–1047. https://doi.org/10.1016/j.procs.2018.07.070

Razfar, N., True, J., Bassiouny, R., Venkatesh, V., & Kashef, R. (2022). Weed detection in soybean crops using custom lightweight deep learning models. *Journal of Agriculture and Food Research*, *8*, 100308. https://doi.org/10.1016/j.jafr.2022.100308

Riera, L. G., Carroll, M. E., Zhang, Z., Shook, J. M., Ghosal, S., Gao, T., Singh, A., Bhattacharya, S., Ganapathysubramanian, B., Singh, A. K., & Sarkar, S. (2021). Deep multiview image fusion for soybean yield estimation in breeding applications. *Plant Phenomics*, *2021*, 9846470. https://doi.org/10.34133/2021/9846470

Santos, M. S., Soares, J. P., Abreu, P. H., Araujo, H., & Santos, J. (2018). Cross-validation for imbalanced datasets: Avoiding overoptimistic and overfitting approaches [research frontier]. *IEEE Computational Intelligence Magazine*, *13*(4), 59–76.

Shook, J., Gangopadhyay, T., Wu, L., Ganapathysubramanian, B., Sarkar, S., & Singh, A. K. (2021). Crop yield prediction integrating genotype and weather variables using deep learning. *PLoS One*, *16*(6), e0252402. https://doi.org/10.1371/journal.pone.0252402

Shurrab, S., & Duwairi, R. (2022). Self-supervised learning methods and applications in medical imaging analysis: A survey. *PeerJ Computer Science*, *8*, e1045. https://doi.org/10.7717/peerj-cs.1045

Singh, A., Ganapathysubramanian, B., Singh, A. K., & Sarkar, S. (2016). Machine learning for high-throughput stress phenotyping in plants. *Trends in Plant Science*, *21*(2), 110–124. https://doi.org/10.1016/j.tplants.2015.10.015

Singh, A., Jones, S., Ganapathysubramanian, B., Sarkar, S., Mueller, D., Sandhu, K., & Nagasubramanian, K. (2021a). Challenges and opportunities in machine-augmented plant stress phenotyping. *Trends in Plant Science*, *26*(1), 53–69. https://doi.org/10.1016/j.tplants.2020.07.010

Singh, A. K., Ganapathysubramanian, B., Sarkar, S., & Singh, A. (2018). Deep learning for plant stress phenotyping: Trends and future perspectives. *Trends in Plant Science*, *23*(10), 883–898. https://doi.org/10.1016/j.tplants.2018.07.004

Singh, A. K., Singh, A., Sarkar, S., Ganapathysubramanian, B., Schapaugh, W., Miguez, F. E., Carley, C. N., Carroll, M. E., Chiozza, M. V., Chiteri, K. O., Falk, K. G., Jones, S. E., Jubery, T. Z., Mirnezami, S. V., Nagasubramanian, K., Parmley, K. A., Rairdin, A. M., Shook, J. M., van der Laan, L., … Zhang, J. (2021b). High-throughput phenotyping in soybean. In J. Zhou & H. T. Nguyen (Eds.), *High-throughput crop phenotyping. Concepts and strategies in plant sciences* (1st ed., pp. 129–163). Springer. https://doi.org/10.1007/978-3-030-73734-4_7

Singh, D. P., Singh, A. K., & Singh, A. (2021c). *Plant breeding and cultivar development* (1st ed.). Elsevier. https://doi.org/10.1016/C2018-0-01730-2

Skendžić, S., Zovko, M., Živković, I. P., Lešić, V., & Lemić, D. (2021). The impact of climate change on agricultural insect pests. *Insects*, *12*(5), 440. https://doi.org/10.3390/insects12050440

Stutz, D. (2015). Superpixel segmentation: An evaluation. In J. Gall, P. Gehler, & B. Leibe (Eds.), Pattern *r*ecognition. DAGM 2015. Lecture *notes in computer sci*ence*: Vol. 9358* (pp. 555–562). Springer. https://doi.org/10.1007/978-3-319-24947-6_46

Tendle, A., & Hasan, M. R. (2021). A study of the generalizability of self-supervised representations. *Machine Learning with Applications*, *6*, 100124.

Tetila, E. C., Machado, B. B., Astolfi, G., de Souza Belete, N. A., Amorim, W. P., Roel, A. R., & Pistori, H. (2020). Detection and classification of soybean pests using deep learning with UAV images. *Computers and Electronics in Agriculture*, *179*, 105836. https://doi.org/10.1016/j.compag.2020.105836

Thenmozhi, K., & Srinivasulu Reddy, U. (2019). Crop pest classification based on deep convolutional neural network and transfer learning. *Computers and Electronics in Agriculture*, *164*, 104906. https://doi.org/10.1016/j.compag.2019.104906

Venugoban, K., & Ramanan, A. (2014). Image classification of paddy field insect pests using gradient-based features. *International Journal of Machine Learning and Computing*, *4*, 1–5. https://doi.org/10.7763/IJMLC.2014.V4.376

Waheed, H., Zafar, N., Akram, W., Manzoor, A., Gani, A., & Islam, S. U. (2022). Deep learning based disease, pest pattern and nutritional deficiency detection system for "*Zingiberaceae*" crop. *Agriculture*, *12*(6), 742. https://doi.org/10.3390/agriculture12060742

Wang, C., Fu, H., & Ma, H. (2022). PaCL: Part-level contrastive learning for fine-grained few-shot image classification. *Proceedings of the 30th ACM International Conference on Multimedia* (pp. 6416–6424). Association for Computing Machinery.

Wang, M., Xu, S., & Zhou, H. (2020). Self-supervised learning for low frequency extension of seismic data. *SEG Technical Program Expanded Abstracts 2020* (pp. 1501–1505). SEG. https://doi.org/10.1190/segam2020-3427086.1

Wu, X., Zhan, C., Lai, Y.-K., Cheng, M.-M., & Yang, J. (2019). IP102: A large-scale benchmark dataset for insect pest recognition. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 8779–8788). IEEE. https://doi.org/10.1109/CVPR.2019.00899

Xia, D., Chen, P., Wang, B., Zhang, J., & Xie, C. (2018). Insect detection and classification based on an improved convolutional neural network. *Sensors*, *18*, 4169. https://doi.org/10.3390/s18124169

Xie, C., Zhang, J., Li, R., Li, J., Hong, P., Xia, J., & Chen, P. (2015). Automatic classification for field crop insects via multiple-task sparse representation and multiple-kernel learning. *Computers and Electronics in Agriculture*, *119*, 123–132. https://doi.org/10.1016/j.compag.2015.10.015

Yang, X., Wang, Y., Chen, K., Xu, Y., & Tian, Y. (2022). Fine-grained object classification via self-supervised pose alignment. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 7399–7408). IEEE.

Yang, Y., & Xu, Z. (2020). Rethinking the value of labels for improving class-imbalanced learning. *Advances in Neural Information Processing Systems*, *33*, 19290–19301.

Yi, J., Krusenbaum, L., Unger, P., Hüging, H., Seidel, S. J., Schaaf, G., & Gall, J. (2020). Deep learning for non-invasive diagnosis of nutrient deficiencies in sugar beet using RGB images. *Sensors*, *20*(20), 5893. https://doi.org/10.3390/s20205893

Yu, X., Zhao, Y., & Gao, Y. (2022). SPARE: Self-supervised part erasing for ultra-fine-grained visual categorization. *Pattern Recognition*, *128*, 108691.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., & Deny, S. (2021). Barlow Twins: Self-supervised learning via redundancy reduction. *International Conference on Machine Learning* (pp. 12310–12320). PMLR.

Zhang, M., Cheng, S., Cao, X., Chen, H., & Xu, X. (2022). *Entropy-based locally adaptive thresholding for image segmentation*. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4010416

Zhong, Y., Gao, J., Lei, Q., & Zhou, Y. (2018). A Vision-based counting and recognition system for flying insects in intelligent agriculture. *Sensors*, *18*(5), 1489. https://doi.org/10.3390/s18051489

Zhuang, P., Wang, Y., & Qiao, Y. (2020). Learning attentive pairwise interaction for fine-grained classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(07), 13130–13137.

How to cite this article: Kar, S., Nagasubramanian, K., Elango, D., Carroll, M. E., Abel, C. A., Nair, A., Mueller, D. S., O'Neal, M. E., Singh, A. K., Sarkar, S., Ganapathysubramanian, B., & Singh, A. (2023). Self-supervised learning improves classification of agriculturally important insect pests in plants. *The Plant Phenome Journal*, *6*, e20079. https://doi.org/10.1002/ppj2.20079