INoD: Injected Noise Discriminator for Self-Supervised Representation Learning in Agricultural Fields

Julia Hindel[1], Nikhil Gosala[1], Kevin Bregler[2], and Abhinav Valada[1]

***Abstract*—Perception datasets for agriculture are limited both in quantity and diversity which hinders effective training of supervised learning approaches. Self-supervised learning techniques alleviate this problem, however, existing methods are not optimized for dense prediction tasks in agricultural domains which results in degraded performance. In this work, we address this limitation with our proposed Injected Noise Discriminator (INoD) which exploits principles of feature replacement and dataset discrimination for self-supervised representation learning. INoD interleaves feature maps from two disjoint datasets during their convolutional encoding and predicts the dataset affiliation of the resultant feature map as a pretext task. Our approach enables the network to learn unequivocal representations of objects seen in one dataset while observing them in conjunction with similar features from the disjoint dataset. This allows the network to reason about higher-level semantics of the entailed objects, thus improving its performance on various downstream tasks. Additionally, we introduce the novel Fraunhofer Potato 2022 dataset consisting of over 16,800 images for object detection in potato fields. Extensive evaluations of our proposed INoD pretraining strategy for the tasks of object detection, semantic segmentation, and instance segmentation on the Sugar Beets 2016 and our potato dataset demonstrate that it achieves state-of-the-art performance.**

***Index Terms*—Robotics and Automation in Agriculture and Forestry, Deep Learning for Visual Perception, Computer Vision for Automation**

I. Introduction

In today's times, there is an ever-growing urgency to make agricultural practices ecologically sustainable while simultaneously improving farm throughput. Precision agriculture provides a solution to this challenge via the use of precise intervention techniques such as targeted spraying of chemicals, as well as using gripper arms, stomping feet, and lasers to destroy weeds while preserving crops. However, precision agriculture heavily relies on plant detection and segmentation which is often challenging due to their varied appearance throughout their growth cycle. This is further exacerbated by extensive overlaps with neighboring plants which results in their edges being indistinguishable from one another. Additionally, the lack of labeled datasets encompassing different crops hinders the development of fully-supervised plant detection and segmentation approaches. Self-supervised learning (SSL) techniques help combat the outlined label deficiency by pretraining the network on pseudo labels obtained from self-derived pretext tasks before finetuning on the downstream target task. The pretraining step enables the network to learn the underlying semantics of the image which helps it better adapt to a wide range of relevant tasks and datasets.

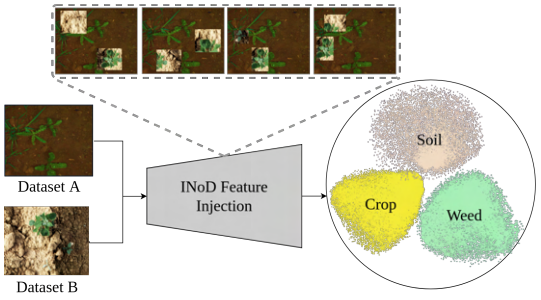


Fig. 1: INoD interleaves convolutional features of two disjoint agricultural datasets during their convolutional encoding to learn semantically meaningful representations of entailed soil, crop, and weed in a self-supervised manner.

Existing SSL approaches in the agricultural domain rely on techniques such as self-labeling [1], feature decorrelation [2], and contrastive learning [3] for training their models in a label-efficient manner. However, self-labeling approaches rely on the NDVI metric [1] which is often very susceptible to varying crop color and external lighting conditions. Further, existing agricultural models directly leverage feature decorrelation and contrastive learning techniques originally introduced for performing classification on the ImageNet dataset. Accordingly, these networks only focus on pretraining the network backbone which is often insufficient for dense prediction tasks such as segmentation and object detection as the task-specific heads are initialized randomly [4]. Moreover, little research has focused on adapting existing SSL techniques to the challenging domain of precision agriculture. Typical SSL approaches are largely tailored for the ImageNet dataset which significantly deviates from agricultural images, especially with regard to the scale, location, and frequency of objects.

To address the aforementioned limitations, we propose ***Injected Noise Discriminator (INoD)***, a novel self-supervised representation learning approach for precision agriculture. The principle behind INoD is to interleave features of two disjoint datasets during their convolutional encoding phase

Manuscript received: March 31, 2023; Revised June 15, 2023; Accepted July 16, 2023.

This paper was recommended for publication by Editor Hyungpil Moon upon evaluation of the Associate Editor and Reviewers' comments. This work was partly funded by the German Research Foundation (DFG) Emmy Noether Program grant number 468878300.

[1]First Author, Second Author and Fourth Author are with the Department of Computer Science, University of Freiburg, Germany hindel@cs.uni-freiburg.de;gosalan@cs.uni-freiburg.de.

[2]Third Author is with Fraunhofer IPA, Stuttgart, Germany.

Digital Object Identifier (DOI): see top of this page.

and then determine the dataset affiliation of the resultant feature map using task-specific detection or segmentation heads. We hypothesize that INoD enables the network to learn unequivocal representations of various types of vegetation while observing them in conjunction with other valid features. This interleaving of features is performed at multiple levels during the forward pass of the network which prevents the trivial distinction of datasets based on generic dataset-specific statistics. Consequently, the network is constrained to determine the origin of features by reasoning about higher-level semantics of the entailed objects. This higher level reasoning enables the network to learn rich representations which improve its overall performance on downstream tasks. We also introduce the novel Fraunhofer Potato 2022 (FP22) dataset which is one of the first publicly available potato datasets with crop and weed annotations. We perform extensive evaluations of INoD on the downstream tasks of object detection, semantic segmentation, and instance segmentation. Our experiments on the Sugar Beets 2016 [5] and the FP22 datasets demonstrate that INoD outperforms state-of-the-art SSL techniques by more than $1.3\,\mathrm{pp}$ in terms of the AP and mIoU metrics. Additional ablation studies also demonstrate that our approach consistently exceeds the performance of standard SSL baselines across a wide range of finetuning splits and pretraining epochs.

Our main contributions can thus be summarized as follows:

1) A novel SSL technique, INoD, for pretraining any dense prediction network without further adaptations.
2) A potato dataset comprising over $16,800$ images and $1,433$ object detection annotations.
3) Several competitive SSL baselines for dense prediction tasks on agricultural datasets.
4) Extensive evaluation and ablation of INoD on two agricultural datasets.
5) Publicly available code and pretrained models at http://inod.cs.uni-freiburg.de.

II. Related Work

In this section, we summarize existing works in SSL and present an overview of approaches that have been adapted to the domain of precision agriculture.

Self-Supervised Learning: SSL is a type of unsupervised learning where pseudo labels are automatically derived from an unlabeled dataset [6]. Since the scope of SSL is wide-reaching, we limit this section to self-supervised pretraining approaches that are commonly employed in conjunction with downstream tasks such as image classification, object detection, and semantic segmentation.

Early SSL approaches employed simple augmentation tasks such as predicting the transformation of images [7] or determining the spatial arrangement of image patches [8] for pretraining the network backbone. These approaches often rely on ad-hoc heuristics during pretraining which often fail to generalize over a wide variety of downstream tasks. Recently, discriminative pretext tasks including contrastive and self-training methods have gained increased attention in the field of SSL. The pioneering contrastive models SimCLR [9] and MoCo [10] have outperformed supervised pretraining on various downstream tasks. Another direction of research omits the need for negative samples in contrastive learning by enforcing the proximity of positive pairs in the latent space. This proximity of positive samples is often realized using different variants of clustering [11], self-distillation [12], [13], feature decorrelation [14] or Siamese networks [15]. In the context of object detection and instance segmentation, where the localization ability of the network is crucial [16], [17], a mismatch between the pretext and final task can easily occur when significantly different objectives are optimized in the different training phases. Nevertheless, existing classification-based pretraining approaches, that learn translation and scale invariance, have been used for object detection resulting in low overall performance [18], [19]. Other works encourage the network to reason about localization using a dense contrastive loss function that only compares region-specific features obtained from known region-based correspondences [19]–[21]. Further extensions contrast global features in addition to region-specific features to generate robust representations [22]–[24]. Parallel works use pre-processing steps such as selective search and unsupervised perceptual grouping to generate local object regions for contrastive learning [25], [26].

More recently, cut-and-paste pretraining strategies, wherein random crops of an image are pasted onto different backgrounds and then classified as foreground and background have been employed to learn rich representations for dense prediction tasks [18], [27]. These contrastive learning-based approaches often demand significant resources during pretraining due to their reliance on large batch sizes or queues of data samples. In this paper, we propose a novel SSL approach, INoD, where feature maps from two disjoint datasets are interleaved during their convolutional encoding, and a network is then tasked to determine the dataset affiliation of the composite feature map. Our approach does not exhibit intra-batch dependencies and can thus be easily trained on a single GPU. Moreover, these contrastive learning-based approaches need to be specifically adapted to pretrain the complete target network and often depend on finding the correct combination of loss terms. In contrast, our approach can be used with any network architecture without additional modifications and can directly be employed for various downstream dense prediction tasks.

Self-Supervised Learning in Agriculture: There are only a handful of SSL approaches in the agricultural domain. Zapata *et al.* [28] train a triple Siamese network to distinguish between plant seedlings and demonstrate its effectiveness using the image retrieval task. Besides, [29] employs randomized color channel recognition to pretrain a network for fruit anomaly detection, while [2] uses BarlowTwins [14] coupled with domain-adapted augmentations for leaf and plant segmentation. SwAV [11] is an SSL approach that was used to pretrain classification and segmentation networks on datasets of grassland, aerial farmland, and weed species [3]. However, the authors report no significant improvement when using SSL for plant segmentation, thus highlighting the need for further research on leveraging SSL for dense prediction tasks in the agricultural domain. Our proposed self-supervised INoD approach demonstrates exceptional performance for

a range of dense prediction tasks such as object detection, semantic segmentation, and instance segmentation.

III. Technical Approach

In this section, we first present an overview of the proposed self-supervised INoD pretraining approach which is tailored for dense prediction tasks in the agricultural domain. The goal of injected feature discrimination is to make the network learn unequivocal representations of objects from one dataset while observing them alongside objects from a disjoint dataset. Accordingly, the pretraining is based on the premise of injecting features from a disjoint *noise* dataset into different feature levels of the original *source* dataset during its convolutional encoding. The network is then trained to determine the dataset affiliation of the resultant feature map which enables learning rich discriminative feature representations for different objects in the image. Fig. 2 provides an overview of our approach.

A. Overview of INoD

Our approach provides a *drop-in* solution to pretrain any network using our novel injected feature discrimination strategy. INoD encourages the network to learn semantically representative features, thus improving the transfer learning ability for dense prediction tasks. This can be attributed to the fact that INoD forces the network to decipher feature origin based on a higher level semantic understanding of the feature maps instead of generic dataset-specific statistics. Our approach comprises four phases, namely, (i) multi-level noise feature generation, (ii) random noise mask generation, (iii) iterative noise injection and subsequent forward pass of the multi-level source image features, and (iv) dense prediction tasks to determine the origin of features in the composite feature map.

First, we use an off-the-shelf network backbone Θ to compute multi-scale feature maps for the noise image: $\Theta(\mathcal{I}_N) \rightarrow \{\mathcal{E}_1^N, \mathcal{E}_2^N, ..., \mathcal{E}_L^N\}$ (Fig. 2(a)). Second, we generate random layer-specific binary noise masks, N_l, for each scale l to determine the spatial locations at which the noise features are injected into the source features. Third, we extract region-specific noise features, $\mathcal{E}_l^R$, by multiplying each of the multi-scale noise features, $\mathcal{E}_l^N$, with their corresponding layer-specific noise mask N_l. Then, we iteratively inject $\mathcal{E}_l^R$ at equivalent locations in the source feature map $\mathcal{E}_l^S$ to generate a composite feature map $\mathcal{E}_l^C = f(\mathcal{E}_l^N, \mathcal{E}_l^S, N_l)$ (Fig. 2(b)). We then perform the subsequent source network traversal step of Θ to generate the next level feature map $\mathcal{E}_{l+1}^S$. Thus, noise injected in early feature maps is carried through higher layers of the network backbone as shown in Fig. 2(c). We intertwine features only along the height and width dimensions of the feature map to ensure the integrity of convolutional feature representations in the channel dimension. Finally, we pass the composite feature maps through a set of convolutional layers to further entangle features and generate coherent multi-scale representations (Fig. 2(d)). We then provide these feature maps as input to downstream task heads such as object detection, semantic segmentation, or instance segmentation to infer the dataset affiliation of the composite feature maps.

B. Noise Mask Generation

Noise mask generation forms one of the core components of the noise injection protocol and defines the regions where source features are replaced by noise features. A noise mask N is primarily defined by its granularity which determines the smallest possible volume that can be replaced in a feature map. In other words, a noise mask with high granularity allows for the substitution of small feature regions and vice versa. In this paper, we select one feature scale from the network backbone as our reference size and correspondingly specify the minimal feature replacement dimensions according to its effective receptive field. Once the granularity is defined, we can trivially generate the noise mask by randomly sampling noise injection positions on a blank canvas. However, these masks often exhibit disjoint and similarly-shaped noise injection positions which reduces the training diversity and prevents the network from being able to predict diverse shapes and sizes. We mitigate this problem by first generating random binary sample masks of size 3×3 with probability $P(1) = \frac{2}{3}$, rescaling them to $[\frac{2}{3}, \frac{1}{6}\times$ reference scale dimensions], and then randomly placing them within the noise mask bounds. This strategy allows us to create highly complex noise patterns that facilitate learning semantically meaningful representations for all three dense prediction tasks.

C. Noise Injection

Our noise injection protocol comprises two stages, namely, (i) layer-specific noise mask extraction, and (ii) layer-specific noise injection. First, we generate layer-specific noise masks $N_{1...L}$ by randomly sub-dividing the noise mask N, defined in Sec. III-B, such that non-zero regions in N are sampled only once in $N_{1...L}$ to ensure that $N = \Sigma_l N_l$ as visualized in Fig. S.1 . We then estimate the region-specific noise features for layer l, $\mathcal{E}_l^R$, by multiplying N_l with $\mathcal{E}_l^N$. Finally, we inject noise into a source feature level by replacing regions of $\mathcal{E}_l^S$ with $\mathcal{E}_l^N$ at equivalent locations. Mathematically,

$$\mathcal{E}_l^C = \mathcal{E}_l^N \cdot N_l + \mathcal{E}_l^S \cdot \neg N_l. \tag{1}$$

The next source feature level is then computed as

$$\mathcal{E}_{l+1}^S = \text{conv}_{l+1}(\mathcal{E}_l^C). \tag{2}$$

D. Task-Specific Ground Truth Label Generation

We supervise the dataset affiliation of the composite feature map by generating task-specific pseudo labels as shown in Fig. S.2 . The following sections outline our pseudo label generation process for three dense prediction tasks.

Object Detection: We supervise the object detection pretext task by creating bounding box pseudo labels from the noise mask N generated in Sec. III-B. First, we compute contours around non-zero elements in N and then draw tight axis-aligned bounding boxes around them. For the specific case when noise mask elements only share a common corner, we avoid computing a common contour and instead compute element-specific contours to prevent a large performance drop caused by sparse bounding boxes. Drawing boxes around contours instead of around every independent noise element allows us to

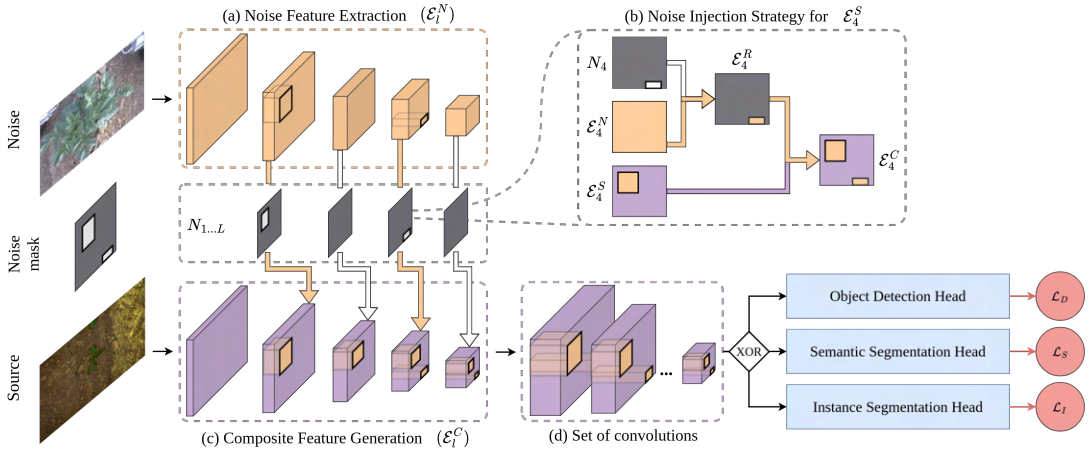


Fig. 2: INoD injects features from a disjoint noise dataset into various feature levels of a source dataset during their convolutional encoding. The noise injection is defined in binary layer-specific noise masks which originate from a randomly generated noise mask. The network is then trained end-to-end to determine the dataset affiliation with an object detection, semantic segmentation or instance segmentation head. In this figure, the "set of convolutions" refers to an FPN for the instance segmentation and object detection tasks, and a DPC module for semantic segmentation.

generate diverse bounding box pseudo labels in terms of both box size and scale which improves the model performance.

Semantic Segmentation: For the semantic segmentation task, we create the corresponding pseudo label by resizing the noise mask N to the pre-determined output resolution using the nearest neighbor interpolation algorithm.

Instance Segmentation: We compute instance segmentation pseudo labels by first generating bounding box pseudo labels and then post-processing them to extract instance IDs of different elements in N. The post-processing step entails extracting non-zero pixels within a bounding box and assigning them a common instance label. We follow the aforementioned two-step process to ensure consistency between the bounding box and instance segmentation pseudo labels which are needed for pre-training the Mask R-CNN architecture.

E. Loss functions

We pretrain networks for target prediction tasks using only the corresponding task-specific losses. In this paper, we pretrain a semantic segmentation network by minimizing the binary focal loss ($\mathcal{L}_S$) [30], optimize an object detection network using the standard detection-specific loss components of the Faster R-CNN architecture ($\mathcal{L}_D$), and train an instance head using Mask R-CNN losses such as proposal, class, bounding box, and mask loss ($\mathcal{L}_I$) [31]. We further describe the network architectures employed in our experiments in Sec. IV-B.

IV. Experimental Evaluation

In this section, we quantitatively and qualitatively evaluate the performance of INoD on three dense prediction tasks and also provide a comprehensive ablation study to demonstrate the importance of our contribution. We first present an overview of the used agricultural datasets and then describe the experimental settings for the pretraining and finetuning pipelines.

A. Datasets

Sugar Beets 2016 (SB16): This dataset comprises recordings of the two months farming cycle of a sugar beet field near Bonn, Germany [5]. It consists of $123,062$ samples while ground truth semantic labels are provided for a subset of $12,196$ images. We generate instance ground truth labels for these samples by drawing contours around the semantic labels using morphological closure and vegetation heuristics. For pretraining, we randomly divide the dataset into splits of $110,756$ training and $12,306$ validation samples. For the finetuning step, we divide the labeled samples into train, validation, and test splits based on their timestamp. Specifically, we finetune all models using the first $9,137$ labeled samples, validate them on the next 612 images, and present the results on the remaining $2,447$ samples. We split the dataset using their timestamps to prevent the networks from remembering the characteristics of vegetation on different farming days, and instead allow them to demonstrate their true generalization capabilities.

Fraunhofer Potato 2022 (FP22): This is a recent dataset by Fraunhofer IPA which was recorded using an agricultural robot at a potato cultivation facility in the outskirts of Stuttgart, Germany. The robot depicted in Fig. 3 comprises a Jai Fusion FS 3200D 10G camera mounted at the bottom of the robot chassis at a height of $0.8\,\mathrm{m}$ above the ground. The dataset contains $16,891$ images obtained from two different stages in the farming cycle of which a subset of $1,433$ images have been annotated with bounding box labels following a peer-reviewed process. For the pretraining step, we train the model using $15,202$ training samples and validate it on the remaining $1,689$ samples. Similar to SB16, we divide the labeled samples into three splits based on their timestamp yielding 867 train, 169 validation, and 397 test samples. We make this dataset publicly available with this work at http://inod.cs.uni-freiburg.de.

B. Experimental Setup

We evaluate our self-supervised INoD pretraining strategy on the dense prediction tasks of instance segmentation, semantic

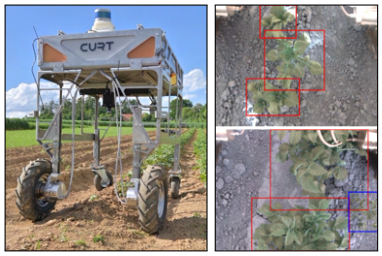


Fig. 3: Agricultural robot by Fraunhofer IPA (left) used to collect our FP22 dataset and samples from our FP22 potato dataset (right).

segmentation, and object detection. Accordingly, we employ a Mask R-CNN network [31] for instance segmentation, a ResNet-50 backbone with a DPC-based head [32] for semantic segmentation, and a Faster R-CNN (R50+FPN) model [33] for object detection. The following sections detail the various parameters used in the pretraining and finetuning stages.

1) Self-supervised Pretraining: We perform self-supervised pretraining for dense prediction tasks using image crops of size 224×224 and perturb these images using random combinations of horizontal flips, gaussian blur, random grayscale, and color jitter as proposed in MoCo-v2 [34]. We randomly initialize the model weights and update them using the SGD optimizer with batches of size 256, momentum of 0.9, and weight decay of 0.0001. We adopt a learning rate of 0.02 which we decay by a factor of 0.1 after 60% and 80% of the training steps. We train the models for 100 and 200 epochs on the SB16 and FP22 datasets respectively and ablate over different pre-training lengths for FP22 in Sec. IV-D2. We inject noise features into the outputs of all residual blocks of the ResNet-50 backbone. For instance segmentation, we inject noise with a granularity of $1/32$, while we employ noise masks with a granularity $1/4$ for semantic segmentation and object detection. Further, we limit the quantity of noise to 20% of the final composite feature map for object detection and instance segmentation, while we apply 40% of noise features for semantic segmentation. We ablate different granularities and quantities of noise injection in Sec. IV-D3 and Sec. IV-D4, respectively. We utilize the FP22 dataset as noise for the SB16 dataset and vice versa. Lastly, we normalize both the source and noise images using the dataset statistics computed using the source dataset.

2) Supervised Finetuning: We first prepare the task-specific pretrained networks for finetuning by recomputing the batch normalization parameters for 400 iterations following the approach outlined in [35]. This pre-processing step negates any discrepancy between pretraining and finetuning dataset distributions and creates a fair starting point for model finetuning. We then finetune these networks on images of size 800×800 for 20 epochs on the SB16 dataset and 50 epochs on the FP22 dataset. We optimize our model using SGD with a batch size of 16, base learning rate of 0.02, momentum of 0.9, and weight decay of 0.0001. Similar to pretraining, we decay the learning rate by a factor of 0.1 after 60% and 80% of training epochs. We augment the train split using random horizontal flips for instance segmentation and object detection, while we also incorporate random gaussian blur and color jitter for the semantic segmentation task.

TABLE I: Evaluation of instance segmentation on Sugar Beets 2016. All metrics are reported in [%] and averaged over three runs.

Pretraining	AP_{crop}	AP_{weed}	mAP	AP_{75}	AP_{50}
Supervised (IN)	38.85	12.58	25.72	18.84	51.28
MoCo-v2 [34]	42.68	15.12	28.90	22.76	55.33
BYOL [12]	48.15	14.04	31.10	31.81	55.49
InsLoc [18]	42.58	12.76	27.67	23.30	51.62
DenseCL [22]	44.94	15.34	30.14	27.65	55.45
AgriBT [2]	42.87	13.83	28.35	24.95	50.66
INoD (Ours)	**54.96**	**15.48**	**35.22**	**41.35**	**58.30**

TABLE II: Evaluation of semantic segmentation on Sugar Beets 2016. All metrics are reported in [%] and averaged over three runs.

Pretraining	IoU_{crop}	IoU_{weed}	IoU_{soil}	mIoU
Supervised (IN)	31.51	9.57	98.97	46.35
MoCo-v2 [34]	34.92	8.06	97.44	46.81
BYOL [12]	23.01	7.71	97.46	42.73
InsLoc [18]	18.40	9.09	98.95	41.81
DenseCL [22]	30.21	7.98	**99.16**	45.24
AgriBT [2]	25.84	**10.23**	96.97	44.35
INoD (Ours)	**36.66**	9.98	97.74	**48.13**

C. Quantitative Results

We benchmark INoD against six popular pretraining baselines, namely, ImageNet pretraining, MoCo-v2 [34], BYOL [22], InsLoc [18], DenseCL [22] and domain-specific BarlowTwins (AgriBT) [2]. We ensure fair comparison between all the baselines by following the same experimental settings outlined in Sec. IV-B and using their published codebases wherever possible. Tab. I and Tab. II present the results for finetuning the network on the SB16 dataset for instance and semantic segmentation, while Tab. III presents the finetuning results for object detection on the FP22 dataset. We evaluate the models using the COCO evaluation metrics, and thereby compute mAP, AP_{crop} and AP_{weed} over IoU=50:95.

We observe from Tab. I that our approach outperforms all the pretraining baselines by more than 4.12 pp on the instance segmentation task. This substantial improvement in performance is a consequence of a distinct increase in both the crop and weed classes and can be attributed to the rich semantic understanding brought about by our pretraining framework. Further, we observe that INoD significantly outperforms the baselines by 9.54 pp on the AP_{75} metric while surpassing them by only 2.97 pp on AP_{50}. This disparity between the AP_{50} and AP_{75} metrics highlights the uncertainty of segmentation predictions of the baselines in challenging regions such as leaf structures and leaf boundaries. We also visually verify this observation using qualitative results in Sec. IV-E.

Tab. II presents the performance of INoD for semantic segmentation in terms of mIoU. This metric is computed as the average over the crop, weed, and background (soil) classes. We observe that INoD exceeds the best-performing baseline by 1.32 pp which can be attributed to an improved segmentation ability on the crop class. While AgriBT shows superior results on the weed class, it performs poorly on crop detection. Further, we note that supervised ImageNet pretraining demonstrates superior results compared to three self-supervised baselines namely BYOL, InsLoc, and DenseCL for this task. This large disparity underlines the degraded performance of existing SSL

TABLE III: Evaluation of object detection on Fraunhofer Potato 2022 dataset. All metrics are reported in [%] and averaged over three runs.

Pretraining	AP_{crop}	AP_{weed}	mAP
Supervised (IN)	56.77	**35.07**	45.92
MoCo-v2 [34]	58.05	32.39	45.22
BYOL [12]	56.46	34.04	45.25
InsLoc [18]	58.31	30.40	44.36
DenseCL [22]	57.63	32.54	45.08
AgriBT [2]	57.70	33.64	45.67
INoD (Ours)	**60.85**	33.24	**47.05**

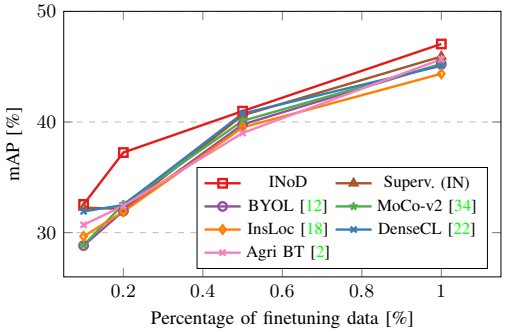


Fig. 4: Ablation study on the impact of finetuning with an increasing amount of data. The results are averaged over three runs.

algorithms for semantic segmentation in agriculture fields.

Tab. III shows the results of INoD for object detection on our FP22 dataset. We observe that our approach outperforms the best SSL benchmark by 1.38 pp and the supervised ImageNet pretraining by 1.13 pp. This substantial improvement in the mAP score can be attributed to the better detection performance on the crop class. Further, we note that supervised ImageNet pretraining achieves the best AP score on the weed class, exceeding the closest self-supervised benchmark by 1.03 pp. We highlight that ImageNet pretraining has an unfair advantage due to two main reasons, namely, (i) a significantly longer training schedule and (ii) a diverse pretraining dataset. ImageNet pretraining is trained for nearly three magnitudes longer as compared to SSL baselines (1.2 billion iterations for ImageNet vs 3 million iterations for SSL) and is also trained on a very diverse dataset which allows it to extract rich information from a wide variety of scenarios. Further, the relatively high performance of the ImageNet pretrained weights on the weed class can be reasoned with the fact that ImageNet comprises several instances of monocots (e.g. grass) which constitute the majority of observed weeds in our FP22 dataset. Nevertheless, our SSL framework is still able to extract useful information pertaining to the crop and weed classes allowing it to achieve competitive detection performance for both classes.

D. Ablation Study

In this section, we study the impact of various architectural components and hyperparameters on the performance of our approach. We execute all ablation experiments on the object detection task using our FP22 dataset.

1) Impact of Pretraining: In this study, we analyze the impact of pretraining on the overall performance of the model by varying the amount of labeled data during the finetuning step. Therefore, we reduce the amount of labeled data during finetuning to 10%, 20%, and 50% of the original finetuning dataset size but maintain a constant number of iterations during finetuning. Fig. 4 shows the mAP scores for the different percentage splits. We observe that our model consistently outperforms all the baselines even when using a small number of labeled samples during finetuning, thus highlighting the benefit of our INoD strategy. Our model also shows superior performance when using only 20% of finetuning samples, outperforming all the baselines by more than 4.67 pp.

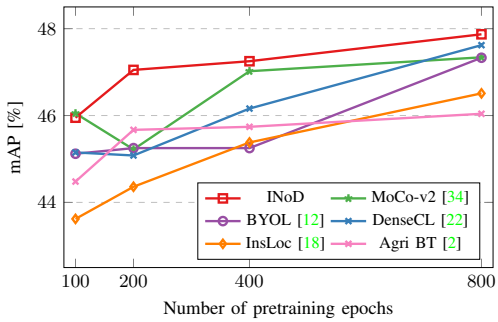


Fig. 5: Ablation study on the impact of pretraining for an increasing number of epochs. The results are averaged over three runs.

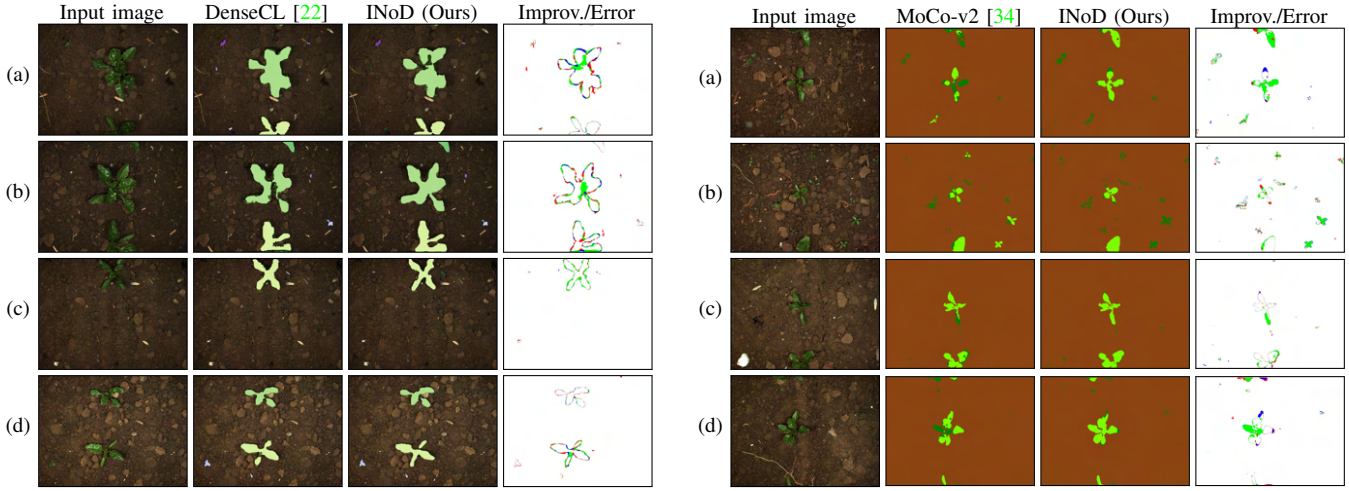
TABLE IV: Ablation study on the granularity of noise injection. All metrics are reported in [%] and averaged over three runs.

Granularity	1/4	1/8	1/16	1/32
mAP	**47.05**	46.71	45.00	46.58

2) Length of Pretraining: In this section, we study the impact of pretraining schedule length on the overall performance of the model. We pretrain all the SSL baselines for 100, 200, 400, and 800 epochs and report the corresponding finetuning results in Fig. 5. We observe that INoD consistently outperforms the SSL baselines when pretrained with longer training schedules and also approaches its final convergence score faster than the other baselines. For instance, our model achieves an mAP of 47.05% at 200 epochs which only marginally improves to 47.87% when pretrained for 800 epochs. In contrast, the baselines achieve an improvement of 1.85 pp on average when the training schedule is increased four-fold from 200 epochs to 800 epochs. We emphasize that quick and early convergence is a crucial attribute for SSL-based approaches as it reduces the use of computational resources and improves practicality for many real-world applications.

3) Granularity of Noise Injection: We analyze the impact of the granularity of noise injection during pretraining in Tab. IV. As defined in Sec. III-B, granularity defines the smallest possible volume that can be replaced in the source feature map. Using the smallest granularity of $\frac{1}{4}$ (smallest replaceable unit of 4 pixels) results in the highest performance. Consequently, we argue that injecting noise with a high resolution (low granularity) allows the network to learn precise features pertaining to the edges of plant structures, thus improving the object detection performance of the model.

4) Quantity of Noise Injection: In this section, we study the impact of varying amounts of noise injection during the pretraining step of our INoD approach. Accordingly, we replace



(i) Instance segmentation results. Crop and weed instances are colored using varying shades of green and purple respectively.

(ii) Semantic segmentation results. Crops are shown in light green, weeds in dark green, and soil in brown.

Fig. 6: Qualitative results of INoD in comparison to the best performing baselines for instance and semantic segmentation on the SB16 dataset. The improvement/error maps show pixels misclassified by the baseline and correctly predicted by INoD in green and vice-versa in blue. Incorrect predictions of both models are colored in red.

Fig. 7: Qualitative results of our self-supervised INoD framework on our Fraunhofer Potato 2022 dataset.

TABLE V: Ablation study on varying the amount of noise injection. All metrics are reported in [%] and averaged over three runs

Amount of Injection	AP_{crop}	AP_{weed}	mAP
10%	59.33	32.55	45.94
20%	**60.85**	33.24	**47.05**
30%	58.58	33.02	45.80
40%	59.76	**33.46**	46.61

10%, 20%, 30%, and 40% of source features with noise features and present the results on the target task in Tab. V. We observe that the model performance slightly deteriorates when we replace either 10%, 30% or 40% of source features with noise features compared to the model where we replace 20% of source features with noise. A small amount of noise injection makes the pretraining task very easy and hinders the network from learning rich and meaningful features. Consequently, the finetuning step is also hindered which results in slightly worse results for both classes. In contrast, a large amount of noise injection results in fewer observed crops belonging to the source dataset, which in turn dampens the learning of the model for the crop class. However, weeds exhibit similar characteristics across the source and noise datasets which minimizes the impact of high noise injection for the weed class. We highlight that the amount of noise injection is a key parameter which should be carefully tuned according to the task and dataset.

E. Qualitative Results

We further evaluate the impact of INoD by qualitatively comparing its predictions with the best performing SSL baseline for the instance and semantic segmentation tasks in Fig. 6i and Fig. 6ii, respectively. For instance segmentation, we observe that both DenseCL and INoD enable the network to segment all multi-leaf crops in the image while effectively neglecting dead vegetation in the image extremities. In Fig. 6i(c), we note that INoD achieves a finer segmentation of the potato crop compared to DenseCL which is evident from the improvement/error map shown in the last column. Further, we observe in Fig. 6i(a, b, d) that INoD is able to segment intricate crop structures and also the connections between individual leaves, while DenseCL fails to do so. Moreover, we also highlight that INoD is able to better segment crops at the boundaries of images as shown in Fig. 6i(b). We argue that this characteristic stems from our pretraining approach that discourages the mixing of convolutional features in the backbone which results in the precise localization of features, especially at image borders where padding is applied. Additionally, we note that both pretraining approaches struggle to identify small weeds which are shown as red dots in the error maps of Fig. 6i(a, b, d).

In the semantic segmentation results shown in Fig. 6ii, we observe that both models precisely segment vegetation from soil but often misclassify crop and weed plants. While INoD routinely predicts the edges of crop leaves as weed, MoCo-v2 misclassifies entire leaves in Fig. 6ii(a, c, d). This characteristic can be attributed to our data split during finetuning wherein the training set comprises images from early farming days having smaller plant structures while the test set contains mature plants. Consequently, MoCo-v2 fails to adapt to this change in plant sizes while INoD succeeds, thus

highlighting its generalizability. Lastly, we note that MoCo-v2 commonly identifies small weeds as crops in Fig. 6ii(a, b) while INoD better generalizes on such occurrences.

We also qualitatively evaluate the object detection performance on the FP22 dataset in Fig. 7. Here, the crop and weed predictions are colored in green and yellow, while the ground truth labels are colored in red and blue, respectively. We observe that INoD predicts tight bounding boxes around complete plant structures and correctly classifies the detected objects. However, we observe that our model occasionally fails to accurately determine the exact edges of overlapping plants as shown in Fig. 7(c, d). Similar to the instance segmentation results, our model also fails to detect small weeds as is evident from the blue bounding box having no corresponding yellow box in Fig. 7(d). Nevertheless, as demonstrated in Fig. 7 our model detects most crops and weeds reliably thus enabling its use in a multitude of precision agriculture applications.

V. Conclusion

In this paper, we present INoD, a novel self-supervised pretraining strategy for dense prediction tasks in agricultural domains. Our approach is based on incorporating principles of feature replacement and dataset discrimination during the convolutional encoding of two disjoint datasets. We observe that our pretraining strategy outperforms the existing state-of-the-art SSL baselines as well as ImageNet pretraining for object detection, semantic segmentation, and instance segmentation on the SB16 dataset as well as our new FP22 dataset. We also publicly release our novel FP22 dataset comprising $16,800$ images for object detection in potato fields. Our approach is one of the early works in SSL to provide a targeted pretraining solution for multiple dense prediction tasks, without architecture-specific customization. Consequently, this research motivates the development of novel SSL approaches outside of the highly explored but limited contrastive learning objective.

References

[1] A. Wendel and J. Underwood, "Self-supervised weed detection in vegetable crops using ground based hyperspectral imaging," in *Int. Conf. on Robotics & Automation*, 2016, pp. 5128–5135.

[2] G. Roggiolani, F. Magistri, T. Guadagnino, *et al.*, "On domain-specific pre-training for effective semantic perception in agricultural robotics," in *Int. Conf. on Robotics & Automation*, 2023.

[3] R. Güldenring and L. Nalpantidis, "Self-supervised contrastive learning on agricultural images," *Comput. Electron. Agric.*, vol. 191, no. C, 2021.

[4] J. V. Hurtado and A. Valada, "Semantic scene segmentation for robotics," in *Deep Learning for Robot Perception and Cognition*, 2022.

[5] N. Chebrolu, P. Lottes, A. Schaefer, W. Winterhalter, W. Burgard, and C. Stachniss, "Agricultural robot dataset for plant classification, localization and mapping on sugar beet fields," *The Int. Journal of Robotics Research*, vol. 36, pp. 1045 – 1052, 2017.

[6] N. Gosala, K. Petek, P. L. J. Drews-Jr, W. Burgard, and A. Valada, "Skyeye: Self-supervised bird's-eye-view semantic mapping using monocular frontal view images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 14 901–14 910.

[7] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Int. Conf. on Learning Representations*, 2018.

[8] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *European Conf. on Computer Vision*, 2016.

[9] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Int. Conf. on Machine Learning*, vol. 119, 2020, pp. 1597–1607.

[10] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2020.

[11] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," in *Advances in Neural Information Processing Systems*, 2020.

[12] J.-B. Grill, F. Strub, F. Altché, *et al.*, "Bootstrap your own latent - a new approach to self-supervised learning," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 21 271–21 284.

[13] M. Caron, H. Touvron, I. Misra, H. J'egou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9630–9640, 2021.

[14] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: Self-supervised learning via redundancy reduction," in *Int. Conf. on Machine Learning*, 2021.

[15] X. Chen and K. He, "Exploring simple siamese representation learning," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2021.

[16] C. Lang, A. Braun, and A. Valada, "Robust object detection using knowledge graph embeddings," in *Proc. 44th DAGM German Conference on Pattern Recognition*, 2022, pp. 445–461.

[17] R. Mohan and A. Valada, "Amodal panoptic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 21 023–21 032.

[18] C. Yang, Z. Wu, B. Zhou, and S. Lin, "Instance localization for self-supervised detection pretraining," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 3986–3995, 2021.

[19] T. Xiao, C. Reed, X. Wang, K. Keutzer, and T. Darrell, "Region similarity representation learning," *Int. Conf. on Computer Vision*, 2021.

[20] P. O. Pinheiro, A. Almahairi, R. Y. Benmalek, F. Golemo, and A. C. Courville, "Unsupervised learning of dense visual representations," in *Advances in Neural Information Processing Systems*, 2020.

[21] J. Ding, E. Xie, H. Xu, C. Jiang, Z. Li, P. Luo, and G. Xia, "Deeply unsupervised patch re-identification for pre-training object detectors." *IEEE trans. on pattern analysis and machine intelligence*, 2022.

[22] X. Wang, R. Zhang, C. Shen, T. Kong, and L. Li, "Dense contrastive learning for self-supervised visual pre-training," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 3023–3032, 2021.

[23] E. Xie, J. Ding, W. Wang, X. Zhan, H. Xu, Z. Li, and P. Luo, "Detco: Unsupervised contrastive learning for object detection," *Int. Conf. on Computer Vision*, pp. 8372–8381, 2021.

[24] C. Lang, A. Braun, L. Schillingmann, K. Haug, and A. Valada, "Self-supervised representation learning from temporal ordering of automated driving sequences," *arXiv preprint arXiv:2302.09043*, 2023.

[25] O. J. H'enaff, S. Koppula, J.-B. Alayrac, A. van den Oord, O. Vinyals, and J. Carreira, "Efficient visual pretraining with contrastive detection," *Int. Conf. on Computer Vision*, pp. 10 066–10 076, 2021.

[26] F. Wei, Y. Gao, Z. Wu, H. Hu, and S. Lin, "Aligning pretraining for detection via object-level contrastive learning," in *Advances in Neural Information Processing Systems*, 2021.

[27] F. Wang, H. Wang, C. Wei, A. Yuille, and W. Shen, "CP2: Copy-Paste Contrastive Pretraining for Semantic Segmentation," *European Conf. on Computer Vision*, pp. 499–515, 2022.

[28] P. A. M. Zapata, S. Roth, D. Schmutzler, T. Wolf, E. Manesso, and D.-A. Clevert, "Self-supervised feature extraction from image time series in plant phenotyping using triplet networks," *Bioinformatics*, 2021.

[29] T. Choi, O. Would, A. S. Gomez, and G. Cielniak, "Self-supervised representation learning for reliable robotic monitoring of fruit anomalies," *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2266–2272, 2021.

[30] T.-Y. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *Int. Conf. on Computer Vision*, 2017.

[31] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Int. Conf. on Computer Vision*, 2017, pp. 2980–2988.

[32] L.-C. Chen, M. Collins, Y. Zhu, G. Papandreou, B. Zoph, F. Schroff, H. Adam, and J. Shlens, "Searching for efficient multi-scale architectures for dense image prediction," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018.

[33] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[34] X. Chen, H. Fan, R. B. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv preprint arXiv:2003.04297*, 2020.

[35] Y. Wu and J. Johnson, "Rethinking "batch" in batchnorm," *arXiv preprint arXiv:2105.07576*, 2021.