applied sciences

MDPI

Article

On the Importance of Diversity When Training Deep Learning Segmentation Models with Error-Prone Pseudo-Labels

Nana Yang [1,*], Charles Rongione [2], Anne-Laure Jacquemart [2], Xavier Draye [2] and Christophe De Vleeschouwer [1,*]

[1] ICTEAM Institute, UCLouvain, 1348 Louvain-la-Neuve, Belgium
[2] ELI Institute, UCLouvain, 1348 Louvain-la-Neuve, Belgium; charles.rongione@uclouvain.be (C.R.); anne-laure.jacquemart@uclouvain.be (A.-L.J.); xavier.draye@uclouvain.be (X.D.)
* Correspondence: nana.yang@uclouvain.be (N.Y.); christophe.devleeschouwer@uclouvain.be (C.D.V.)

Abstract: The key to training deep learning (DL) segmentation models lies in the collection of annotated data. The annotation process is, however, generally expensive in human resources. Our paper leverages deep or traditional machine learning methods trained on a small set of manually labeled data to automatically generate pseudo-labels on large datasets, which are then used to train so-called data-reinforced deep learning models. The relevance of the approach is demonstrated in two applicative scenarios that are distinct both in terms of task and pseudo-label generation procedures, enlarging the scope of the outcomes of our study. Our experiments reveal that (i) data reinforcement helps, even with error-prone pseudo-labels, (ii) convolutional neural networks have the capability to regularize their training with respect to labeling errors, and (iii) there is an advantage to increasing diversity when generating the pseudo-labels, either by enriching the manual annotation through accurate annotation of singular samples, or by considering soft pseudo-labels per sample when prior information is available about their certainty.

Keywords: traditional machine learning; segmentation; pseudo-label

Citation: Yang, N.; Rongione, C.; Jacquemart, A.-L.; Draye, X.; Vleeschouwer, C.D. On the Importance of Diversity When Training Deep Learning Segmentation Models with Error-Prone Pseudo-Labels. *Appl. Sci.* **2024**, *14*, 5156. https://doi.org/10.3390/app14125156

Academic Editor: Andrea Prati

Received: 25 March 2024
Revised: 31 May 2024
Accepted: 9 June 2024
Published: 13 June 2024

1. Introduction

Semantic segmentation, which consists in a pixel-wise classification, is a fundamental and essential task in the field of computer vision. With the recent development of deep learning, automatic image processing has become accurate in many fields, at the cost of the time-consuming annotation of large amounts of data [1–3]. To limit the annotation time, semi-supervised approaches have received extensive attention [4–7]. The main idea of semi-supervised methods is to learn a model using a small amount of labeled data and a large amount of unlabeled data. The key to the success of this strategy lies in how to leverage limited labeled data to generate pseudo-labels. Pseudo-labels are labels that are automatically generated, based on the knowledge extracted from the few labeled data. In contrast to ground-truth labels, pseudo-labels are subject to errors. Resorting to the automatic generation of pseudo-labels is motivated by the fact that manual annotation of the entire dataset is unrealistic in many practical applications, where the amount of data is large and the annotation is pixel-wise.

In practice, semi-supervised training is implemented in two steps [8–12]. In the first step, labeled data are used to train a relatively weak machine learning model. In the second step, the weak model is used to assign (error-prone) labels to unlabeled data, which are used to supervise the training of a hopefully more robust model as is shown in Figure 1.

This paper focuses on the cases where the manual annotation load is kept low. In this specific case, traditional machine learning methods are known to achieve a good trade-off in terms of accuracy and generalization capabilities [13–16]. Hence, our paper explores how traditional machine learning models can be integrated in a semi-supervised training approach either to facilitate the annotation of the few initial labeled data (first step) or to predict the pseudo-labels of unlabeled data (second step).

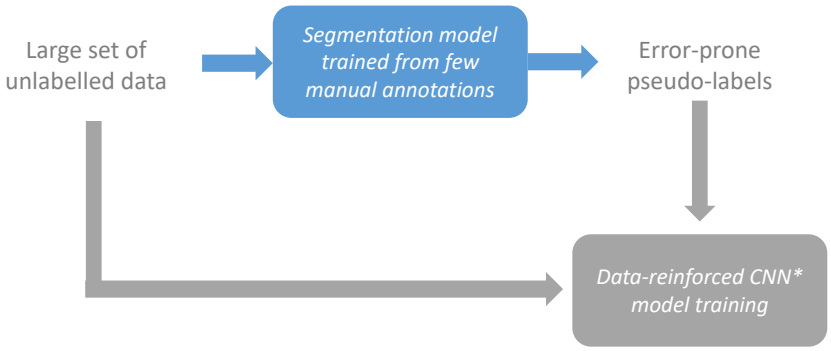


Figure 1. Overview of the envisioned semi-supervised framework. A small set of (semi-)manually annotated samples are considered to train a relatively weak segmentation model. This weak model is then used to annotate a potentially large set of unlabeled data, resulting in error-prone pseudo-labels, which are used to supervise the training of a convolutional neural network. This neural network, denoted CNN*, is named the *data-reinforced* model because it leverages the distribution of patterns in the unlabeled dataset to distill the knowledge embedded in the strictly manually supervised model.

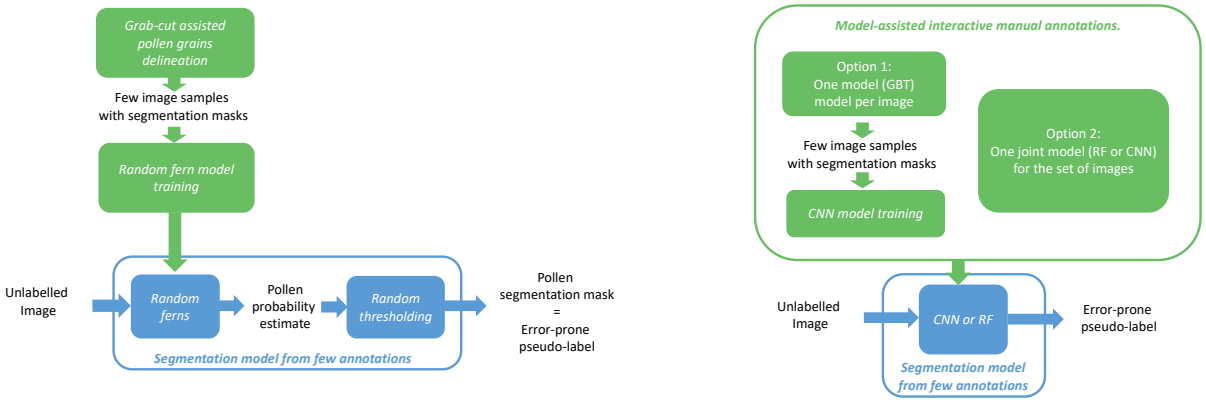
The envisioned semi-supervised framework is depicted in Figure 1. A small set of (semi-)manually annotated samples are considered to train an initial and relatively weak segmentation model. This initial model is then used to annotate a potentially large set of unlabeled data, resulting in error-prone labels, which are used to supervise the training of a convolutional neural network. This neural network is denoted CNN* in the following, and is named the *data-reinforced* model because it leverages the distribution of patterns in the unlabeled dataset to improve the strictly manually supervised model. In a sense, the unlabeled data and associated pseudo-labels are used to distill the knowledge captured by the supervised model.

Two flavors of the pseudo-label generation framework are presented in Figure 2a,b. They differ in the way they exploit traditional machine learning approaches to collect manual annotations and turn them into pseudo-labels. They have been designed to address the specificities of the segmentation problem they are dealing with.

Figure 2a considers a scenario where the objects to segment are easy to delineate manually. Hence, manual labels are assumed to be available for a few images and are used to supervise the training of a conventional semi-naive Bayesian estimation model. This model provides class probability estimates on unlabeled data, which are then used to define pseudo-labels and supervise the training of the data-reinforced CNN* model. Our experiments in Section 4 demonstrate that adopting a stochastic approach to randomly turn the soft probabilities into pseudo-labels at each training iteration is advantageous since it limits the potential bias induced by systematic errors of the Bayesian estimator.

In contrast, Figure 2b considers a case where objects have complex shapes, thereby requiring some semi-automatic approach to be delineated. In our study, various machine learning models (random forests, gradient-boosted detection trees, and even CNNs) have been considered to assign labels a few initial images. In practice, those machine learning models are trained interactively bymanually assigning labels to image parts (strokes or blocks). This is performed either on individual images (one model per image) or jointly on the whole set of initial images (one joint model for all images). When one model is trained per image, a CNN is then trained from the whole set of labeled images to aggregate the entire annotation knowledge in a single joint model. The gradient boosted trees (GBT) model is quite attractive for handling an interactive annotation procedure since it is convenient to update based on novel annotations. However, a single GBT has appears to be unable to generalize to a set of pictures. Hence, we propose to interactively learn one GBT per image, and then to use the resulting annotations to train a CNN as a pseudo-label generator. In all cases, the joint model is used to generate the pseudo-labels on unlabeled data. As a valuable and original outcome, our experiments reveal that the initial

annotation methods that randomly select the manually annotated image part samples, or tend to regularize the manual annotation inputs (by capturing their main average trends, e.g., consequently to the use of Bayesian models), result in weaker data-reinforced models, compared to methods that favor and are able to preserve the manual annotation diversity.



(**a**) (**b**)

Figure 2. Overview of the two scenarios envisioned in our study to collect the few manual annotations needed to train the pseudo-label generator. (**a**) The object is easy to segment manually, and a few manual labels are used to supervise the training of a conventional Bayesian estimation model providing class-probability estimates on unlabeled data. Those class probabilities are then used to define pseudo-labels and supervise the training of a CNN model. (**b**) The object has complex shapes requiring some semi-automatic approach to be delineated. Two options are envisioned. In Option 1, gradient-boosted detection trees are considered to interactively assign labels to a few initial images. The trees, which are Bayesian estimators in essence, appear to poorly generalize to new data. Hence, a CNN is considered to be trained from the few labeled data and is used to generate the pseudo-labels on unlabeled data. In Option 2, one model (either CNN or random forest) is progressively trained from the manual annotations that are interactively collected on a small set of images to correct the current model predictions. This final model is the used as the pseudo-label generator.

In our experiments, the two scenarios correspond to pollen and crop images, respectively. Through quantitative and qualitative analysis, we find that, in both cases, the integration of conventional machine learning algorithms in the annotation process can achieve good results with a moderate manual annotation effort. The main outcomes of our study can be summarized as follows:

(1) Data-reinforced training based on large set of unlabeled data helps, even with error-prone pseudo-labels. This is attested by the higher segmentation accuracy achieved by the data-reinforced models, compared to their corresponding (error-prone) pseudo-label generator.

(2) CNNs training is robust to labeling errors, especially when those errors do not present systematic patterns.

(3) Adopting stochastic pseudo-labels or increasing the diversity of manual annotations is shown to increase the benefit derived from data-reinforced training. This is attested by our experiments, which reveal that best performance are obtained with CNN pseudo-label generator, trained with samples that are interactively selected to correct its prediction errors. Using a CNN avoids the dilution (of annotation information) inherent to probability estimates manipulated by Bayesian models, while the interactive selection of samples results in larger diversity than a random selection. Despite the fact that they are consistent across all the methods and models tested on our two datasets, our experimental results remain too preliminary to draw a definitive and general conclusion about the need for diverse annotations and an expressive pseudo-label generator. Our study is, however, sufficiently convincing to trigger further theoretical and experimental investigations.

2. Related Work

Semantic segmentation currently plays a very important role in many fields, such as medical image analysis, scene understanding, and robotic perception [17]. Although deep learning models have achieved very good performance in semantic segmentation [18], they require labeled data. The acquisition of annotations generally requires a lot of time and manpower, and many fields require experts to do the work. Therefore, considering the use of unannotated data in semantic segmentation has been investigated.

Many methods based on unlabeled data have been studied to improve the performance of learning algorithms. Refs. [19–23] achieved good results in image classification using a semi-supervised approach, consisting in assigning pseudo-labels to unlabeled data, using a model trained from a few data. Ref. [24] has been a precursor in proposing to initially train one classifier on a labeled dataset in order to make predictions on an unlabeled dataset, which uses an initial small labeled dataset to train deep or shallow neural networks and other non-interpretable but high-precision models on the self-training framework, thereby obtaining an enlarged dataset, which is used to train interpretable models such as random trees. In the first stage of this method, a certain amount of labeled data is needed to obtain high accuracy based on CNN. Secondly, the simple model such as random trees has poor generalization ability. Semi-supervised segmentation approaches usually augment manually labeled training data by generating pseudo-labels for the unlabeled data and using these to generate segmentation models [25]. Consistency regularization and entropy minimization represent two prevalent strategies in using unlabeled data [26]. Consistency-based approaches [27,28] operate on the premise that the model's output should remain stable under input perturbations. Conversely, entropy minimization [29–31] suggests that unlabeled data can be exploited in achieving clear class distinctions. Alternatively, refs. [9,12,32,33] introduced methods where pseudo-labels are generated from a universal teacher model to train a problem-specific student model. Those works assume that enough annotated data are available to teach a high-quality teacher. Errors made in the early iterations can be propagated and amplified in subsequent iterations, leading to a student model that may become increasingly confident in incorrect predictions. Moreover, the model's predictions on unlabeled data may lack accuracy when the initial dataset labeled to train the teacher is not sufficiently representative of the domain targeted by the student.

Although the above methods achieve good results based on labeled and unlabeled data, previous works have largely overlooked the relation between the strategy (including the selection of manually labeled samples, and the pseudo-label generator model trained from those samples) adopted to generate pseudo-labels and the benefit obtained when using those pseudo-labels (at constant average quality) to train a data-reinforced model. We propose integrating traditional machine learning methods with deep learning to address the challenge of scarce labeled data. Semi-automatic interactive approaches are considered to collect accurate labels on complex segmentation problems. Two distinct scenarios are envisioned. In the scenario depicted in Figure 2a, the objects to segment are relatively easy to discriminate from their background, and a simple Bayesian random fern model is sufficient to capture the knowledge contained in manually annotated samples. When using the random ferns to infer pseudo-labels on new data, prior information is available about the class-label certainty/probability. As an original contribution, we propose to exploit this prior to introduce diversity among the pseudo-labels. Specifically, each time a sample is considered in a training iteration, the segmentation map of each class is defined based on a random threshold, applied to the certainty level. In the second scenario, considered in Figure 2b, the segmentation task is more complex, and a CNN is considered as an alternative to the Bayesian pseudo-label generator, to capture the knowledge associated to the (semi-)manual annotations, without 'averaging' it in a Bayesian inference framework. Moreover, in this second scenario, the samples to manually label are either selected randomly or to correct the errors of the model trained from previously selected and annotated data. As a main original outcome, our experiments reveal that, at constant pseudo-label average accuracy, the models trained from pseudo-labeled data are more accurate when

those pseudo-labels reflect the pseudo-label generator uncertainty (first scenario) and when the pseudo-label generator preserves the diversity of annotations inherent to an interactive selection of samples to manually annotate (second scenario, with the interactive selection of samples and CNN aggregation of knowledge, without Bayesian dilution).

3. Methods

As explained in Section 1, our paper investigates how to leverage machine learning models when resorting to pseudo-labels to train a deep learning model on a large dataset.

In this case, a small amount of manually annotated data is used to train a model, and this manually supervised model is exploited for the automatic labeling of additional data. Research questions of interest are related to (i) the selection of the data to manually annotate, (ii) the choice of tools to manually annotate them, and (iii) the design of appropriate methods to turn the (semi-)manually collected knowledge into pseudo-labels on novel data samples.

Two distinct scenarios are considered in our experimental section. For each scenario, the above questions are addressed differently, but common lessons can be drawn for both of them, resulting in useful practical guidelines.

The rest of the section presents the multiple options envisioned by each step of our framework, depicted in Figure 1. Section 3.1 considers the (semi-)manual annotation of a data subset. Section 3.2 focuses on how to embed the knowledge captured through those manual annotations into a machine learning model, in charge of transferring this knowledge to novel data. Section 3.3 then introduces two strategies to pseudo-label additional training data for the purpose of training more accurate models when considering a large set of (error-prone) pseudo-labeled data than the accurately labeled but smaller initial set of samples.

3.1. (Semi-)Manual Annotation

Multiple approaches are considered to manually assign a background/foreground segmentation mask to a given training sample. In practice, a specific off-the-shelf approach will be selected based on the case at hand, i.e., on the shape of the foreground region to discriminate from its background.

The investigated methods encompass the following:

- The manual delineation of the foreground object contours. This solution is suited to smooth and regular object shapes, and is adopted in the following to delineate the pollen grains in our first experimental scenario.
- Semi-manual methods, relying on interactive user annotations to refine and increment the training set that is used to train a machine learning model. Those learning-based methods are suited in cases where the foreground and background are visually dissimilar but are separated by an irregular border, requiring time for fully manual delineation.

Among the interactive training methods, we differentiate methods that train a single model to segment the whole set of images to manually annotate, from the ones that learn a specific model to label each individual image. In the first category, the Ilastik 1.3.3 [34] updates a random forest from user-defined foreground strokes until it achieves a sufficiently accurate segmentation of the training data. Another approach investigated in our experiments considers a set of small patches extracted randomly within the whole set of images, to train a fully convolutional neural network (CNN). In contrast, the RootPainter approach [35] selects the patches interactively. Specifically, it trains, jointly for all images, a CNN based on the user interactive selection and labeling of patches that are wrongly predicted by the CNN segmentation model at some point in the training.

In the second category, our LeafPainter method adopts gradient-boosted decision trees (GBTs) [36], and the manual annotation of small patches that are interactively selected by the user in response to the GBT prediction errors. One GBT model is learned for each image (GBT has been implemented using the LightGBM library version 3.2.0 from Microsoft (Ke et al., 2017) so as to support a fluid interactive annotation process. LeafPainter was created with the Python 3.10 programming language, using the Flask library (Grinberg, 2018).

The interface was implemented in HTML5, CSS, and Javascript. The code is available at https://github.com/charlesro/LeafpainterQT (13 April 2022), and a visual demonstration is available at https://www.youtube.com/watch?v=lMpzxDt40Lc (28 April 2022). GBTs have the advantage of being computationally simple to update when new annotations are provided by the user. They, however, suffer from a lack of generalization capabilities, preventing the use of a single GBT for all images.

In the rest of the paper, those methods are denoted as sRF, CNN(sRP), sCNN, and iGBT respectively. The s or i prefixes refer to the fact that a model is trained for the entire set of images or for each individual image, respectively. Our experiments reveal that increasing annotation diversity by learning a distinct model for each image or by interactively training a CNN (i.e., a model offering a large expressivity) improves the benefit drawn from unlabeled data (see Section 3.3).

3.2. Training a Pseudo-Label Generator

Once a number of images (or image patches) have been properly labeled, they are used to train a machine learning model, using conventional supervised training. This model is the one in charge of predicting pseudo-labels.

Two types of models are considered at this stage.

The first one follows a Bayesian approach. Several variants are available, including random forests [37] and random ferns [38,39]. They can achieve good results with only a small amount of labeled data. Moreover, since those methods naturally provide a class posterior for each pixel, multiple pixel-wise segmentation masks can be generated by thresholding the estimated probability map with more or less conservative thresholds. Our experiments reveal that this diversity increases the benefit drawn from unlabeled data when using them as training samples (Section 3.3). In practice, our experiments rely on random ferns to segment pollen grain microscopy images. Random forests (RFs) have also been trained from a set of canopy images, to be used directly as a pseudo-label generator (see Table 1).

Table 1. Names of models on outdoor RGB canopy image dataset. X indicates that the test model has been trained on manually labeled data only, without pseudo-labeled data exploitation.

(Semi_)Manual Annotation	Pseudo-Label Generator	Test Model
sRF [34]	X	sRF
	sRF	CNN*_sRF
sRP	CNN(sRP)	CNN*_CNN(sRP)
	X	CNN(sRP)
iGBT	CNN(iGBT)	CNN*_CNN(iGBT)
	X	CNN(iGBT)
sCNN [35]	sCNN	CNN*_sCNN
	X	sCNN

The second one adopts a recent convolutional deep learning model. These kinds of models are more expressive and offer better generalization capabilities than conventional machine learning models when dealing with 'in the wild' observations [40–42], namely, with observation conditions that are weakly constrained, as is the case for the canopy images considered in our second experimental scenario. In this scenario, plant images are (semi-)manually segmented, either using one GBT model per image or a single convolutional network for the whole set of images (as proposed in [35] and presented in Section 3.1). When one GBT is trained per image, which offers fluent interactions with the user given the computational efficiency associated to GBT updates, the resulting segmentation masks are used to supervise a convolutional neural network (denoted as CNN(iGBT) in Table 1).

3.3. Training a CNN from Pseudo-Labels

Given a segmentation model trained with manually labeled data, our study considers the exploitation of this model to generate/predict pseudo-labels on unlabeled data so as to

train a so-called *data-reinforced* segmentation CNN model, denoted CNN* in the following, with the * symbol indicating the use of pseudo-labels on a large dataset to train the model.

When the manually supervised model follows a Bayesian paradigm, a posterior background/foreground class probability map is predicted for each unlabeled image. Instead of using a fixed threshold to turn this probability map into a foreground/background binary mask, which is likely to induce systematic errors in the supervision, we envision the use of distinct thresholds each time a sample is considered during training. Specifically, each time unlabeled data are considered during training, a threshold is randomly selected using a uniform distribution on a pre-defined range interval (a,b), to produce a corresponding mask of pseudo-labels. In this way, the reference segmentation mask of the same picture is different at distinct training iterations, thus increasing the diversity of labels. Our experiments demonstrate that this supervision, named soft-supervision in the following, results in improved generalization performance of the model trained on unlabeled samples (see Tables 2 and 3).

Table 2. Foreground IoU of different methods on pollen test dataset as a function of the number of manually labeled images. In total, 250 pseudo-label samples, and 50 test samples are considered.

	Foreground IoU (%)					
Method	**1 Images**	**5 Images**	**10 Images**	**15 Images**	**25 Images**	**50 Images**
CNN	37.4	58.9	69.9	72.4	80.4	81.6
Random Ferns	74.7	74.0	77.3	80.0	79.6	79.2
CNN*_0.45	72.2	76.0	76.0	73.7	74.7	74.3
CNN*_0.50	80.0	81.3	82.2	82.4	82.6	82.7
CNN*_0.55	75.0	73.3	76.2	80.3	79.7	78.2
CNN*_(0.45,0.55)	**84.5**	**83.3**	**84.3**	**84.8**	**85.0**	**85.1**
CNN*_CNN	45.7	65.9	73.8	75.7	82.8	82.9

Table 3. Pixel accuracy of different methods on pollen test dataset under different labeled images.

	Pixel Accuracy (%)					
Method	**1 Images**	**5 Images**	**10 Images**	**15 Images**	**25 Images**	**50 Images**
CNN	75.2	86.2	94.6	95.1	97.3	97.4
Random Ferns	97.0	96.9	97.3	97.6	97.5	97.5
CNN*_0.45	90.1	94.4	96.0	96.2	95.5	95.7
CNN*_0.50	97.0	97.9	98.0	97.4	97.7	97.8
CNN*_0.55	97.6	97.4	97.6	98.1	98.0	97.9
CNN*_(0.45,0.55)	**98.4**	**98.1**	**98.2**	**98.2**	**98.3**	**98.2**
CNN*_CNN	73.5	83.3	91.9	92.7	98.2	98.1

With a CNN trained from manual annotation, the access to a probability map, whilst possible [43–45], is less obvious since it requires multiple predictions to estimate the prediction certainty. Therefore, in our experiments, we consider a single mask per image when exploiting unlabeled data based on pseudo-labels generated with a CNN model.

4. Experimental Validation

This section considers semantic image segmentation in two use cases that significantly differ in their image acquisition set-up, which is known to affect generalization, and in the regularity of the shapes to be segmented, which directly affects the annotation strategy. For both use cases, the section considers experiments to demonstrate that assigning pseudo-labels to unlabeled data based on models trained on a few (semi-)manually annotated samples improves the quality of the resulting data-reinforced CNN* segmentation model, compared to a CNN model trained only on the manually labeled data.

4.1. Datasets

4.1.1. Pollen Grain Microscopy Images

The proposed method is evaluated on a pollen dataset. The pollen files are scanned by biologists in pure pollen files and uploaded to Cytomine [46]. The resolution of each file is up to 30,000 by 30,000, which limits the training of the model, so we crop 800 by 800 images in those large images. Fifty of these 800 × 800 images are extracted to be manually labeled. An additional set of 300 images is extracted from other high-resolution original images. Of these, 250 remain unlabeled, while the other 50 are manually labeled to define a test set.

4.1.2. RGB Canopy Images

The dataset consists of 2386 zenithal RGB images of experimental plots, each containing either a single vegetable species or a mixture of two different vegetables. From the dataset, 280 images were randomly extracted to be manually annotated and serve as a test set. There are four vegetable species in total: fennel, dwarf bean, cabbage, and kale. The dataset includes images of each species in pure culture as well as all possible combinations of two vegetables. We are interested in differentiating plant and non-plant pixels.

Each plot was photographed four times over the course of the four-month (July, August, September, and October 2021) experiment, resulting in images of the plants at different stages of growth and under varying sunlight conditions. The photos were taken in Belgium at either the Lauzelle farm (50.680, 4.618) or one of 20 volunteer market gardeners' plots. The resulting images exhibit a wide range of contexts, including variations in soil types, crop arrangements, and non-plant objects present on the plots, as well as differences in cultivation techniques among the market gardeners. The cameras used consist of 2 OEM OV2640 (Omnivision) camera modules (OmniVision, Santa Clara, CA, USA), RGB, 2Mpixel, embedded in an ESP32-Cam module (Espressif Systems, Shanghai, China),installed parallel to each other.

4.2. Methods and Terminology

Several variants of the methods described in Section 3 have been considered for each use case. They are summarized in Tables 1 and 4, and are presented below.

Table 4. Models considered on pollen grain microscopy image dataset. See Section 4.2 for a description of the models associated to the terminology. X indicates that the test model has been trained on manually labeled data only, without pseudo-labeled data exploitation.

(Semi_)Manual Annotation	Pseudo-Label Generator	Test Model
Manual Delineation	X	CNN
	X	Random Ferns
	Random Ferns τ	CNN*_τ
	Random Ferns (τ_1,τ_2)	CNN*_(τ_1,τ_2)
	CNN	CNN*_CNN

4.2.1. Pollen Grain Microscopy

The manual annotation of microscopy pollen grain images is implemented using Grabcut [47], which is both efficient and effective given the simplicity of the image content. Those manually labeled images are then used to train a random ferns model [48], which in turn is used to assign pseudo-labels to the unlabeled images. In practice, we utilize 100 ferns and each fern consists of 10 tests. The point neighborhood is typically defined using a small square window. This square window has a size of (2r + 1) by (2r + 1) pixels and is centered around the pixel of interest. Within this window, binary tests are performed. Five of them compare the absolute difference between two pixels with t, where t is derived from the statistical distribution of the absolute differences between any two pixels in the 800 by 800 image, and t is randomly selected from the interval (20, 40). This choice is motivated by the observation that the majority of difference values fall within the range of 0 to 20.

Consequently, when two pixels exhibit difference values within this range, they are either both part of the foreground or both part of the background. Therefore, these five tests are capable of distinguishing edges from non-edges. The other five tests involve comparing pixel value with the difference between the peak and the variance of the histogram of the 800 by 800 image. Given that most of the image area represents the background, the peak of the histogram is indicative of the background location. As the pixel values of the background exhibit slight variations, we utilize both the peak and the variance of the histogram to characterize the background. Consequently, these five tests effectively distinguish between the background and foreground.

The last column in Table 4 presents the test models, i.e., the models whose performance are measured on the test set and reported in our experimental section. The CNN and random ferns models are trained on manually labeled data only, while the CNN* models refer to data-reinforced models, meaning that they are trained on the entire training set based on pseudo-labels derived either from the random ferns or the manually trained CNN. When pseudo-labels are predicted by the CNN, the data-reinforced model is denoted CNN*_CNN. When deriving pseudo-labels from the random ferns, the pseudo-labels are generated either by thresholding the probability map predicted by random ferns at a fixed threshold τ, leading to a trained convolutional model that is denoted CNN*_τ, or by considering a uniform distribution of thresholds in an interval (τ_1, τ_2) to randomly select the threshold each time an image is considered along the SGD training iterations (see Section 3.2). This last definition of pseudo-labels results in a trained data-reinforced model denoted as CNN*_(τ_1, τ_2)

4.2.2. Outdoor RGB Canopy Images

The (semi-)manual segmentation of leaves requires more elaborated tools than a Grabcut method, due to the variation in illumination conditions, the presence of shadows, and the frequent occlusions by background objects or plants. Therefore, machine learning models are considered to label a subset of the training set through the iterative annotation of image parts.

The subset of samples to label consists of 800 samples, corresponding to 800×600 images, extracted from full resolution images. Those samples are jointly considered by the Ilastik software [34], which trains a random forest (RF) model (100 trees, with a maximal depth of 50) based on foreground strokes that are interactively drawn by the user. This approach is denoted sRF, with the prefix 's' referring to the fact that the whole set of images to manually label is jointly processed. As an alternative, the sRP method randomly samples patches in the same whole set and trains a CNN based on their annotation (see below for CNN parameters). Two more methods are considered to manually annotate the 800 image samples. They are interactive and proceed iteratively by adopting the corrective annotation protocol recommended in [35]. A small subset of samples is first annotated (based on the delineation of small patches corresponding to a single class), without referring to a model prediction. Those annotations are used to train an initial version of the model. Subsequent annotations correct the model predictions and are used to update the model. Two types of image segmentation models are envisioned, resulting in two distinct methods. The first one corresponds to the gradient-boosted decision trees [36] model, with default parameters. In that case, due to the experienced lack of generalization capabilities of GBT, one model is trained for each of the 800 images, and the method is denoted $iGBT$, with the 'i' prefix referring to the fact that one GBT is trained for each individual image. The second one is a fully convolutional neural network (CNN). Its expressivity is sufficient to capture the knowledge associated to the entire set of 800 images. Therefore, a single model is considered for the whole set of manually labeled images. It is denoted $sCNN$. The notation CNN*_X is adopted to refer to the data-reinforced convolutional network trained with pseudo-labels generated by model X.

4.3. Metrics and Implementation Details

4.3.1. Evaluation Metrics

In semantic segmentation, the intersection over union is a standard metric to assess segmentation models. It calculates the intersection over union ratio of ground truth and predicted binary masks for the foreground. Formally, it is defined as:

$$\text{IoU} = \frac{|G \cap P|}{|G \cup P|} \tag{1}$$

with G denoting the ground truth, and P the model prediction associated to the foreground.

Another metric used is pixel accuracy, which evaluates the model by calculating the proportion of correctly classified pixels to the total number of pixels in the image as

$$\text{PA} = \frac{\sum_{i=0}^{1} p_{ii}}{\sum_{i=0}^{1} \sum_{j=0}^{1} p_{ij}} \tag{2}$$

where p_{ii} is the number of pixels whose prediction and ground truth are both i. p_{ij} is the number of pixels whose prediction is i and label is j.

4.3.2. CNN Network

The CNN network used in our work consists of an encoder and decoder, following a U-Net shape [49,50]. Compared to the U-net introduced in [51], we use a seven-layer network. Each layer in the network consists of a convolution operation, followed by batch normalization and the Rectified Linear Unit (ReLU) activation function. The encoder is composed of four layers, with each layer being followed by a downsampling operation. Following the bottleneck layer, we have the decoder. In order to facilitate training with larger batch sizes on the pollen dataset, we aim to minimize model parameters. To achieve this, we simplify the decoder section by reducing the number of layers to two, each preceded by an upsampling operation, compared to the classic U-Net architecture. And a ResNet50 (pretrained on ImageNet) is used as an encoder for the canopy images.

During the training, random rotate90, flip, transpose, and brightness contrast adjustment are used as data augmentation methods.

4.3.3. Loss

The loss function adopted to train the CNN is defined in Equation (5) as the Log-Cosh Dice Loss [52]:

$$coshx = \frac{e^x + e^{-x}}{2} \tag{3}$$

$$DiceLoss = 1 - \frac{2 \sum_{i=1}^{N} \sum_{c=1}^{C} y_i^c \cdot p_i^c}{\sum_{i=1}^{N} \sum_{c=1}^{C} y_i^{c2} + \sum_{i=1}^{N} \sum_{c=1}^{C} p_i^{c2}} \tag{4}$$

$$Loss = \log(cosh(DiceLoss)) \tag{5}$$

In Equation (4), y_i^c is a binary indicator (0 or 1) indicating whether pixel i lies in class c or not, and p_i^c is the softmax output of the network for pixel i and class c.

4.3.4. Training

Stochastic gradient descent algorithm with 0.9 momentum and 0.0001 weight decay is used with a batch size of 4. In total, 100 training epochs are considered. We use 0.01 as the initial learning rate, and an exponential learning rate scheduler is implemented: $lr = lr_{init} \cdot 0.97^{epoch}$. We save the last 10 models from a training session and use the mean prediction as the result. All models are trained with Pytorch [53] on Nvidia GeForce GTX 1080 Ti 11Gb GPUs (Nvidia, Santa Clara, CA, USA).

4.4. Results and Discussion

The methods in Tables 1 and 4 and were experimentally tested on the pollen grain microscopy image dataset and the outdoor RGB canopy image dataset.

4.4.1. Pollen Grain Microscopy

In Tables 2 and 3, we present the results of our experiments on the pollen grain microscopy dataset. The results obtained with the models trained only from manually labeled data are provided as a baseline but also because they reflect the level of error affecting the pseudo-labels. They indicate that the random ferns method performs significantly better than the CNN method when only one annotated image is available. However, as the annotated data increase, the CNN method overcomes the random ferns. Specifically, when using 50 annotated images, the CNN achieves a foreground IoU result of 81.6%, surpassing random ferns, which achieves 79.2%. The pixel accuracy is also comparable, with values of 97.4% for CNN and 97.5% for random ferns. When using random ferns and CNN as pseudo-label generators, even with only one annotated image, the CNN*_0.50 yields an IoU result of 80.0%, outperforming random ferns, which achieves 74.7%. The experimental results clearly indicate that leveraging unlabeled data through pseudo-labels enhances CNN* model quality compared to a model (either random ferns or CNN) trained solely on manually labeled samples. Interestingly, Tables 2 and 3 also demonstrate that CNN*_(0.45,0.55) consistently outperforms CNN*_0.45, CNN*_0.50, and CNN*_0.55, indicating that increasing the diversity of pseudo-labels is beneficial. Additionally, Figure 3 illustrates the predictions of various methods on the pollen test set. The models depicted in the figure are trained using only a single labeled image. Through the comparison between the predictions and the ground truth, it is observed that employing a fixed threshold to generate pseudo-labels causes a bias in training [10] , resulting in inferior predictions compared to using a varying threshold.

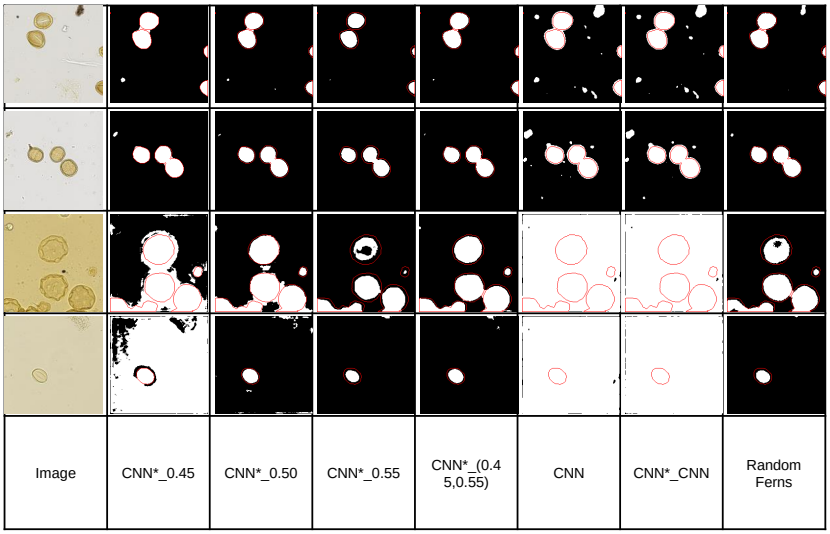


Figure 3. Qualitative results on pollen test dataset. See Table 4 for model names definition. All models are trained using 1 labeled image. Red contour is ground truth.

4.4.2. Outdoor RGB Canopy Images

Table 5 shows the foreground IoU for various methods on the outdoor RGB canopy image dataset under different manual annotation times. Here, the manual annotation time denotes the time needed to annotate the training set used to learn the pseudo-label generator. It includes both the time to manually correct the errors affecting the current model prediction, and the (computational) time to update the model. Table 6 presents the corresponding pixel accuracy. The results indicate that the data-reinforced CNN* consistently outperforms the methods that are solely trained on manually labeled data. This demonstrates that leveraging

unlabeled data through pseudo-labels is beneficial. Note that the quality of pseudo-labels is measured by the test performance obtained by models trained from manually labeled data. Without surprise, for a given pseudo-label generator model, the higher the pseudo-label quality, the higher the CNN* quality.

A similar observation can be made in Figure 4a, where each CNN* model achieves test performance that is generally superior to that obtained by the model from which the CNN* training pseudo-labels have been predicted. Hence, we conclude that resorting to pseudo-labels helps in reducing the human annotation workload. This is confirmed by Figure 4b, which compares the pixel accuracy on the test set between the CNN* model and its corresponding pseudo-label generator. The points in the figure are consistently located above the diagonal, emphasizing that leveraging unlabeled data through pseudo-labels improves the model quality compared to a model trained only on manually labeled samples.

Figure 4a also clearly reveals that the boost obtained by the CNN* compared to the model used to predict the pseudo-labels is much larger for CNN(iGBT) and sCNN than for CNN(sRP) and sRF. For example, when the model trained on manually labeled samples reaches about 83%, CNN*_CNN(iGBT) reaches 94.5%, while CNN*_sRF only reaches 88.6%. This observation was not expected, and is thus quite interesting.

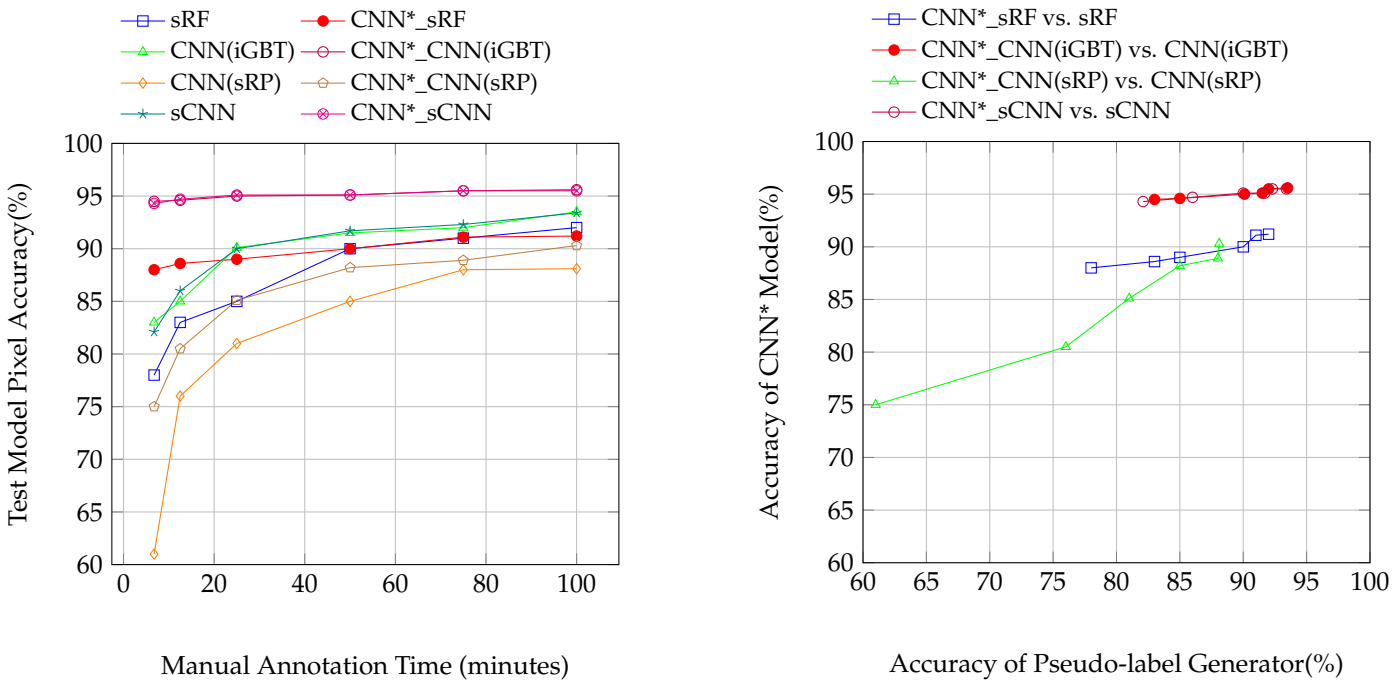


Figure 4. (**a**) Test accuracy vs. manual annotation time on RGB canopy images, for the models defined in Table 1. (**b**) CNN* test accuracy vs. test accuracy obtained by its corresponding pseudo-label generator on RGB canopy images, for the models defined in Table 1.

This unexpected discrepancy is attributed to the combination of two factors. First, compared to CNN(sRP), the sRF, CNN(iGBT), and sCNN models benefit from more diverse manually labeled samples since those samples are not randomly selected but instead chosen interactively as image parts where the current state of the model fails. Second, compared to sRF, CNN(iGBT) and sCNN build on a CNN model, which is known to have sufficient expressivity to capture a large annotation diversity without regularizing it [54], i.e., without aggregating the annotation cues through average probabilities, as done by Bayesian models like random ferns or random forests. While the enriched representation captured by a CNN model does not necessarily translate into improvements on the test images (e.g., 83% test accuracy both for the CNN(iGBT) and sRF models), it proves to be beneficial when

confronted with the distribution of unlabeled data, i.e., when leveraging their predicted labels to train a data-reinforced CNN*.

Figure 5 displays the visual results of different models on the test set. From the results, it is evident that CNN*_CNN(iGBT) exhibits superior segmentation performance.

Table 5. IoU results on RGB canopy images under different annotation times (minutes). In total, 280 test samples are considered.

	Foreground IoU (%)					
Annotation Time	**6.75**	**12.5**	**25**	**50**	**75**	**100**
sRF	63.0	69.5	72.3	81.7	83.3	84.2
CNN*_sRF	79.6	80.2	81.8	82.8	83.5	83.7
CNN(iGBT)	69.5	72.4	81.8	83.5	84.3	86.9
CNN*_CNN(iGBT)	88.7	89.2	89.6	89.9	90.7	90.8
CNN(sRP)	42.9	67.2	74.2	78.7	79.9	82.2
CNN*_CNN(sRP)	61.3	68.1	78.6	80.2	81.8	83.5
sCNN	69.3	72.6	81.9	83.3	84.3	86.9
CNN*_sCNN	**89.0**	**89.4**	**89.6**	**89.9**	**90.8**	**91.0**

Table 6. Pixel accuracy results on RGB canopy images under different annotation times (minutes).

	Pixel Accuracy (%)					
Annotation Time	**6.75**	**12.5**	**25**	**50**	**75**	**100**
sRF	78.3	83.2	85.1	89.7	91.0	91.8
CNN*_sRF	88.1	88.6	89.1	90.0	91.1	91.2
CNN(iGBT)	83.1	85.2	90.1	91.5	92.0	93.5
CNN*_CNN(iGBT)	94.5	94.6	95.0	**95.1**	**95.5**	**95.6**
CNN(sRP)	61.1	75.9	81.3	85.2	87.8	88.1
CNN*_CNN(sRP)	75.2	80.5	85.1	88.2	88.9	90.3
sCNN	82.1	85.9	90.0	91.7	92.3	93.4
CNN*_sCNN	**94.3**	**94.7**	**95.1**	**95.1**	**95.5**	95.5

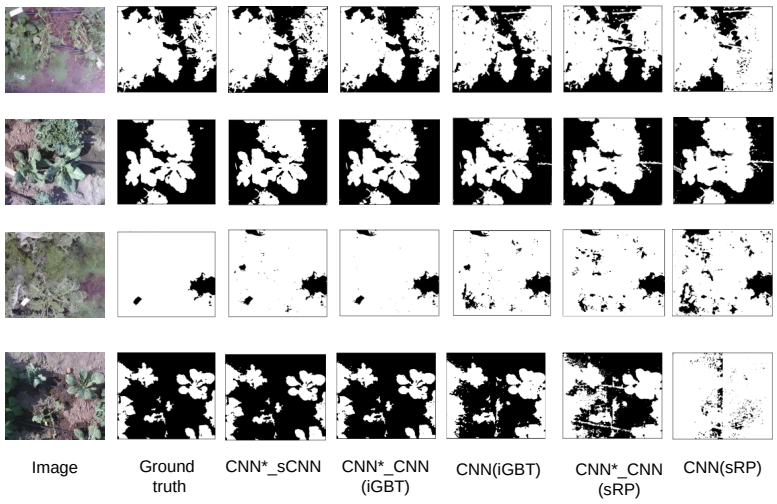


Figure 5. Qualitative results on RGB canopy images as obtained with a 6.75 min annotation time. See Table 1 for the model names definition.

5. Conclusions

In our quest to address the challenge of training CNN models with limited manual annotation resources, we delve into the fusion of traditional machine learning techniques with deep learning, focusing on predicting pseudo-labels. We investigate two scenarios

that are representative of typical simple (no occlusion, reasonably stable appearance) and complex (shape variations and occlusions) object image segmentation problems. Through experiments across two datasets, we discover that integrating unlabeled data via pseudo-labels emerges as a potent strategy, markedly enhancing model efficacy compared to models trained solely on manually labeled samples. We extensively investigate the relation between (i) the strategy adopted to collect manual annotations and turn them into a pseudo-label generator, and (ii) the benefit drawn from pseudo-labels. Our study underscores the significance of pseudo-label diversity. Enriching the pseudo-label generator can be achieved explicitly by considering multiple pseudo-labels per image or implicitly through interactive annotation methods to select and annotate diverse samples to train a pseudo-label generator that offers high expressivity (thereby avoiding regularizing/averaging the diversity inherent to selected samples). These insights collectively enrich our understanding of the advantages and tactics associated with harnessing pseudo-labels, offering valuable guidance for improving model performance when human resources are limited for annotation.

Author Contributions: N.Y. and C.R.: Methodology (Annotation and CNN design), Software Development (Python libraries and codes, and annotation interfaces), Formal and experimental analysis, Visualization of failure cases, Writing—original draft. C.D.V., X.D. and A.-L.J.: Conceptualization (Problem formulation, and data collection), Methodology (Imaging and Machine Learning), Supervision of experimental analysis, Writing part of original draft + review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: Nana Yang is funded by China Scholarship Council. C. De Vleeschouwer is partly funded by the Belgian F.N.R.S.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used in the paper are available at: https://www.kaggle.com/datasets/yangnana/outdoor-rgb-canopy-and-pollen-grain-microscopy (13 March 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Apley, D.W.; Zhu, J. Visualizing the effects of predictor variables in black box supervised learning models. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2020**, *82*, 1059–1086. [CrossRef]
2. Castelli, M.; Vanneschi, L.; Largo, Á.R. Supervised learning: Classification. *Encycl. Bioinform. Comput. Biol.* **2018**, *1*, 342–349.
3. Sarmadi, H.; Entezami, A. Application of supervised learning to validation of damage detection. *Arch. Appl. Mech.* **2021**, *91*, 393–410. [CrossRef]
4. Zhou, Z.H. Semi-supervised learning. In *Machine Learning*; Springer: Singapore, 2021; pp. 315–341.
5. Ouali, Y.; Hudelot, C.; Tami, M. An overview of deep semi-supervised learning. *arXiv* **2020**, arXiv:2006.05278.
6. Zheng, M.; You, S.; Huang, L.; Wang, F.; Qian, C.; Xu, C. SimMatch: Semi-supervised Learning with Similarity Matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 14471–14481.
7. Sayez, N.; De Vleeschouwer, C. Accelerating the creation of instance segmentation training sets through bounding box annotation. In Proceedings of the 2022 26th International Conference on Pattern Recognition (ICPR), Montreal, QC, Canada, 21–25 August 2022; pp. 252–258.
8. Mendel, R.; De Souza, L.A.; Rauber, D.; Papa, J.P.; Palm, C. Semi-supervised segmentation based on error-correcting supervision. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part XXIX 16; pp. 141–157.
9. Xie, Q.; Luong, M.T.; Hovy, E.; Le, Q.V. Self-training with noisy student improves imagenet classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10687–10698.
10. Arazo, E.; Ortego, D.; Albert, P.; O'Connor, N.E.; McGuinness, K. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8.
11. Yang, L.; Zhuo, W.; Qi, L.; Shi, Y.; Gao, Y. St++: Make self-training work better for semi-supervised semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 4268–4277.

12. Zhu, Y.; Zhang, Z.; Wu, C.; Zhang, Z.; He, T.; Zhang, H.; Manmatha, R.; Li, M.; Smola, A.J. Improving semantic segmentation via efficient self-training. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *46*, 1589–1602. [CrossRef] [PubMed]
13. Smith, A. Image segmentation scale parameter optimization and land cover classification using the Random Forest algorithm. *J. Spat. Sci.* **2010**, *55*, 69–79. [CrossRef]
14. Chen, H.; Wu, L.; Chen, J.; Lu, W.; Ding, J. A comparative study of automated legal text classification using random forests and deep learning. *Inf. Process. Manag.* **2022**, *59*, 102798. [CrossRef]
15. Fröhlich, B.; Rodner, E.; Denzler, J. Semantic segmentation with millions of features: Integrating multiple cues in a combined random forest approach. In Proceedings of the Asian Conference on Computer Vision, Daejeon, Republic of Korea, 5–9 November 2012; pp. 218–231.
16. Wei, P.; Hänsch, R. Random Ferns for Semantic Segmentation of PolSAR Images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5218212. [CrossRef]
17. Mo, Y.; Wu, Y.; Yang, X.; Liu, F.; Liao, Y. Review the state-of-the-art technologies of semantic segmentation based on deep learning. *Neurocomputing* **2022**, *493*, 626–646. [CrossRef]
18. Xiao, Z.; Liu, B.; Geng, L.; Zhang, F.; Liu, Y. Segmentation of lung nodules using improved 3D-UNet neural network. *Symmetry* **2020**, *12*, 1787. [CrossRef]
19. Rizve, M.N.; Duarte, K.; Rawat, Y.S.; Shah, M. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *arXiv* **2021**, arXiv:2101.06329.
20. Zhang, B.; Wang, Y.; Hou, W.; Wu, H.; Wang, J.; Okumura, M.; Shinozaki, T. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 18408–18419.
21. Wei, C.; Shen, K.; Chen, Y.; Ma, T. Theoretical analysis of self-training with deep networks on unlabeled data. *arXiv* **2020**, arXiv:2010.03622.
22. Liu, J.; Yao, J.; Bagheri, M.; Sandfort, V.; Summers, R.M. A semi-supervised CNN learning method with pseudo-class labels for atherosclerotic vascular calcification detection. In Proceedings of the 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), Venice, Italy, 8–11 April 2019; pp. 780–783.
23. Liu, B.; Yu, X.; Zhang, P.; Tan, X.; Yu, A.; Xue, Z. A semi-supervised convolutional neural network for hyperspectral image classification. *Remote Sens. Lett.* **2017**, *8*, 839–848. [CrossRef]
24. Pintelas, E.; Livieris, I.E.; Pintelas, P. A grey-box ensemble model exploiting black-box accuracy and white-box intrinsic interpretability. *Algorithms* **2020**, *13*, 17. [CrossRef]
25. Chakravarthy, A.D.; Abeyrathna, D.; Subramaniam, M.; Chundi, P.; Gadhamshetty, V. Semantic image segmentation using scant pixel annotations. *Mach. Learn. Knowl. Extr.* **2022**, *4*, 621–640. [CrossRef]
26. Zou, Y.; Zhang, Z.; Zhang, H.; Li, C.L.; Bian, X.; Huang, J.B.; Pfister, T. Pseudoseg: Designing pseudo labels for semantic segmentation. *arXiv* **2020**, arXiv:2010.09713.
27. Xu, H.; Liu, L.; Bian, Q.; Yang, Z. Semi-supervised semantic segmentation with prototype-based consistency regularization. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 26007–26020.
28. Zhang, B.; Zhang, Y.; Li, Y.; Wan, Y.; Guo, H.; Zheng, Z.; Yang, K. Semi-supervised deep learning via transformation consistency regularization for remote sensing image semantic segmentation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.* **2022**, *16*, 5782–5796. [CrossRef]
29. Vu, T.H.; Jain, H.; Bucher, M.; Cord, M.; Pérez, P. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2517–2526.
30. Vesal, S.; Gu, M.; Kosti, R.; Maier, A.; Ravikumar, N. Adapt everywhere: Unsupervised adaptation of point-clouds and entropy minimization for multi-modal cardiac image segmentation. *IEEE Trans. Med Imaging* **2021**, *40*, 1838–1851. [CrossRef]
31. Zeng, G.; Peng, H.; Li, A.; Liu, Z.; Liu, C.; Yu, P.S.; He, L. Unsupervised Skin Lesion Segmentation via Structural Entropy Minimization on Multi-Scale Superpixel Graphs. *arXiv* **2023**, arXiv:2309.01899.
32. Cioppa, A.; Deliege, A.; Istasse, M.; De Vleeschouwer, C.; Van Droogenbroeck, M. ARTHuS: Adaptive real-time human segmentation in sports through online distillation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 15–20 June 2019.
33. Baldeon Calisto, M. Teacher-student semi-supervised approach for medical image segmentation. In *MICCAI Challenge on Fast and Low-Resource Semi-supervised Abdominal Organ Segmentation*; Springer: Cham, Switzerland, 2022; pp. 152–162.
34. Berg, S.; Kutra, D.; Kroeger, T.; Straehle, C.N.; Kausler, B.X.; Haubold, C.; Schiegg, M.; Ales, J.; Beier, T.; Rudy, M.; et al. Ilastik: Interactive machine learning for (bio) image analysis. *Nat. Methods* **2019**, *16*, 1226–1232. [CrossRef]
35. Smith, A.G.; Han, E.; Petersen, J.; Olsen, N.A.F.; Giese, C.; Athmann, M.; Dresbøll, D.B.; Thorup-Kristensen, K. RootPainter: Deep learning segmentation of biological images with corrective annotation. *New Phytol.* **2022**, *236*, 774–791. [CrossRef] [PubMed]
36. Si, S.; Zhang, H.; Keerthi, S.S.; Mahajan, D.; Dhillon, I.S.; Hsieh, C.J. Gradient boosted decision trees for high dimensional sparse output. In Proceedings of the 34th International Conference on Machine Learning, PMLR, Sydney, NSW, Australia, 6–11 August 2017; pp. 3182–3190.
37. Schonlau, M.; Zou, R.Y. The random forest algorithm for statistical learning. *Stata J.* **2020**, *20*, 3–29. [CrossRef]
38. Browet, A.; De Vleeschouwer, C.; Jacques, L.; Mathiah, N.; Saykali, B.; Migeotte, I. Cell segmentation with random ferns and graph-cuts. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 4145–4149.

39. Parisot, P.; De Vleeschouwer, C. Scene-specific classifier for effective and efficient team sport players detection from a single calibrated camera. *Comput. Vis. Image Underst.* **2017**, *159*, 74–88. [CrossRef]
40. Bay, Y.Y.; Yearick, K.A. Machine Learning vs. Deep Learning: The Generalization Problem. *arXiv* **2024**,arXiv:2403.01621.
41. Neyshabur, B.; Bhojanapalli, S.; McAllester, D.; Srebro, N. Exploring generalization in deep learning. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 2017, 4–9 December 2017.
42. Kawaguchi, K.; Kaelbling, L.P.; Bengio, Y. Generalization in Deep Learning. In *Mathematical Aspects of Deep Learning*; Cambridge University Press: Cambridge, UK, 2022. [CrossRef]
43. Léger, J.; Leyssens, L.; Kerckhofs, G.; De Vleeschouwer, C. Ensemble learning and test-time augmentation for the segmentation of mineralized cartilage versus bone in high-resolution microCT images. *Comput. Biol. Med.* **2022**, *148*, 105932. [CrossRef] [PubMed]
44. Gal, Y. Uncertainty in Deep Learning. 2016. Available online: http://106.54.215.74/2019/20190729-liuzy.pdf (accessed on 8 June 2024).
45. Smith, L.; Gal, Y. Understanding measures of uncertainty for adversarial example detection. *arXiv* **2018**, arXiv:1803.08533.
46. Rubens, U.; Hoyoux, R.; Vanosmael, L.; Ouras, M.; Tasset, M.; Hamilton, C.; Longuespée, R.; Marée, R. Cytomine: Toward an open and collaborative software platform for digital pathology bridged to molecular investigations. *PROTEOMICS- Appl.* **2019**, *13*, 1800057. [CrossRef]
47. Li, Y.; Zhang, J.; Gao, P.; Jiang, L.; Chen, M. Grab cut image segmentation based on image region. In Proceedings of the 2018 IEEE 3rd international conference on image, vision and computing (ICIVC), Chongqing, China, 27–29 June 2018; pp. 311–315.
48. Ozuysal, M.; Calonder, M.; Lepetit, V.; Fua, P. Fast keypoint recognition using random ferns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *32*, 448–461. [CrossRef]
49. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015; Proceedings, Part III 18; Springer: Cham, Switzerland, 2015; pp. 234–241.
50. Ibtehaz, N.; Rahman, M.S. MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation. *Neural Netw.* **2020**, *121*, 74–87. [CrossRef] [PubMed]
51. Yang, N.; Joos, V.; Jacquemart, A.L.; Buyens, C.; De Vleeschouwer, C. Using Pure Pollen Species When Training a CNN To Segment Pollen Mixtures. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, New Orleans, LA, USA, 18–24 June 2022; pp. 1695–1704.
52. Jadon, S. A survey of loss functions for semantic segmentation. In Proceedings of the 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), Via del Mar, Chile, 27–29 October 2020; pp. 1–7.
53. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. In Proceedings of the Advances in Neural Information Processing Systems 32 (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019.
54. Zhang, C.; Bengio, S.; Hardt, M.; Recht, B.; Vinyals, O. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM* **2021**, *64*, 107–115. [CrossRef]