information

Article

Enabling Multi-Part Plant Segmentation with Instance-Level Augmentation Using Weak Annotations

Semen Mukhamadiev, Sergey Nesteruk, Svetlana Illarionova and Andrey Somov *

Skolkovo Institute of Science and Technology, 121205 Moscow, Russia
* Correspondence: a.somov@skoltech.ru

Abstract: Plant segmentation is a challenging computer vision task due to plant images complexity. For many practical problems, we have to solve even more difficult tasks. We need to distinguish plant parts rather than the whole plant. The major complication of multi-part segmentation is the absence of well-annotated datasets. It is very time-consuming and expensive to annotate datasets manually on the object parts level. In this article, we propose to use weakly supervised learning for pseudo-annotation. The goal is to train a plant part segmentation model using only bounding boxes instead of fine-grained masks. We review the existing weakly supervised learning approaches and propose an efficient pipeline for agricultural domains. It is designed to resolve tight object overlappings. Our pipeline beats the baseline solution by 23% for the plant part case and by 40% for the whole plant case. Furthermore, we apply instance-level augmentation to boost model performance. The idea of this approach is to obtain a weak segmentation mask and use it for cropping objects from original images and pasting them to new backgrounds during model training. This method provides us a 55% increase in mAP compared with the baseline on object part and a 72% increase on the whole plant segmentation tasks.

Keywords: image instance segmentation; weakly supervised segmentation; multi-part segmentation

Citation: Mukhamadiev, S.; Nesteruk, S.; Illarionova, S.; Somov, A. Enabling Multi-Part Plant Segmentation with Instance-Level Augmentation Using Weak Annotations. *Information* **2023**, *14*, 380. https://doi.org/10.3390/info14070380

Academic Editor: Heming Jia

Received: 8 May 2023
Revised: 25 June 2023
Accepted: 29 June 2023
Published: 3 July 2023

1. Introduction

Computer vision tasks, such as object detection and segmentation, require large-scale datasets to train neural network models [1]. An object detection task involves identification of object boundaries in an image, while a segmentation task assumes pixel classification. These tasks pose several challenges for researchers. The first one is image collection; the second issue concerns the preparation of high-quality annotations for the dataset. Obtaining precise annotations is a time-consuming and costly process, especially for large-scale datasets [2].

Computer vision tasks in the agriculture domain are even more challenging [3,4]. Plants are very diverse and volatile. For many practical problems, we have to solve even more difficult tasks. We need to distinguish plant parts rather than the whole plant. These masks are used to identify different parts of the plant, such as leaves, stems, and fruits. This information can be used to quantify plant traits such as leaf area, stem diameter, and fruit size. It can also help in identifying specific plant diseases that affect certain parts of the plant. The use of computer vision systems in agriculture can help automate many tasks, such as crop monitoring, weed detection, and yield estimation. Accurate plant segmentation and part segmentation masks are essential for developing such automated systems. The plant part segmentation model can be used as a component in a larger pipeline for precision agriculture. For example, the model can be used to identify and segment different parts of a plant, such as leaves, stems, and fruits, from images captured by drones or other imaging devices. This information can then be used to analyze the health and growth of the plant, optimize irrigation and fertilization, and detect diseases or pests early on. The model can also be beneficial in reducing manual labor and increasing efficiency in agricultural

operations. By automating the process of plant part segmentation, farmers can save time and resources that would otherwise be spent on manual inspection and analysis.

For this task, we usually need fine-grained manual annotations. However, it is not feasible to collect and annotate plant parts datasets for every plant variety and condition. It is reasonable to utilize data augmentation techniques in such cases to enlarge the dataset. In some cases, synthetic data can save us thousands of hours for manual annotation [5]. Besides classical augmentations such as color and geometrical transformations, there are advanced techniques [6,7]. For instance, one can apply object-based augmentation (OBA) [8,9]. The key idea of the OBA is to crop foreground target objects from the image using their masks, apply some augmentations to these instances, and then past them onto a new background. OBA is more flexible than classical image-based augmentation, providing more ways to handle the target objects [10]. However, to extend the amount of data in the custom dataset using OBA, masks of the target objects are necessary [11]. Semantic segmentation annotation requires more time and resources than image-level annotation because it involves defining per-pixel boundaries of the target objects. One promising approach to make the pixel-wise annotation process easier is weakly supervised semantic segmentation (WSSS), which utilizes weak supervision such as image-level labels and bounding boxes. Implementing WSSS to obtain masks of target objects for OBA techniques significantly accelerates the creation of custom datasets and simplifies the process of data labeling.

Several approaches have been proposed recently to deal with the task of WSSS. Image-level labels ascribing is the most convenient and cost-effective type of image annotation. Most recent studies in WSSS that use image-level labels employ a class activation map (CAM) method to generate pseudo-masks. The CAM is obtained from the classification network with a global average pooling (GAP) layer [12]. The classification network activates specific features of the input image depending on the class label. The CAM approach highlights the most important parts of the image on which the class prediction is based. However, WSSS methods based on the CAM have drawbacks such as underactivation. It means that CAM produces high response only in the most discriminative regions, but ignores other regions that can be important for segmentation. Therefore, many research studies are devoted to enlarging region coverage provided by CAM. It is important to emphasize that OBA employment makes tough demands for CAM quality because too noisy or corrupted pseudo-masks can ruin OBA as well as the training process.

Typically, WSSS refers to methods for addressing the semantic segmentation task, which is an important computer vision task that is applied in critical systems such as aerial image analysis [13], unmanned aerial vehicles (UAVs) [14], autonomous vehicles (AVs) [15], robotics [16], and environmental analysis [17]. However, the high cost of pixel-wise annotations limits progress in these research fields. Combining OBA and WSSS can significantly improve progress and increase the size of annotated datasets, which in turn can have a positive impact on neural network training in general. This work aims to obtain a segmentation mask using only a limited amount of weak supervision labels, such as class labels and bounding boxes. All experiments in this work were performed on agricultural images, which present several challenges for computer vision tasks. Firstly, plants exhibit a wide range of morphological variations and can vary in appearance at different stages of growth, making accurate recognition difficult. Secondly, the appearance of plants can be significantly influenced by environmental factors and imaging properties, such as lighting conditions, background clutter, and occlusion, which can further complicate the recognition process. Thirdly, plants can share similar visual characteristics with each other, making it challenging to distinguish between different species or varieties of plants. Additionally, acquiring large amounts of high-quality labeled data for training models in plant segmentation can be difficult and costly, especially for rare or exotic plant species. Lastly, the computational complexity of plant segmentation tasks can be high, particularly when dealing with large-scale datasets requiring significant amounts of computational resources and specialized hardware.

The novelty of this paper is in the exploration of weakly-supervised approaches for object parts segmentation.

The main contributions of the work are the following:

- We collect and annotate a dataset of images in agricultural domain. The dataset covers multiple subdomains, and has segmentation masks for each plant part.
- We commit a detailed review of weakly supervised and unsupervised images segmentation methods.
- We present a new robust weakly supervised algorithm that allows training instance segmentation models having only bounding box annotations.
- We present a pipeline with instance-level augmentation based on weakly supervised segmentation and prove its efficiency.

The remainder of the paper is organized as follows: Section 2 describes the literature devoted to the recent WSSS methods; Section 3 describes experiment methodology and methods used; Sections 4 and 5 report on the results and include discussion.

2. Literature Review

This section introduces recent works on weakly supervised or unsupervised methods. These methods have general applications that include not only segmentation but also object detection and saliency detection tasks. One of the simplest forms of weak supervision in weakly supervised semantic segmentation (WSSS) is through the use of image-level labels.

2.1. Weakly Supervised Methods

The approaches below generate class activation maps (CAMs) by constructing graphs. In A2GNN [18], images are transformed into weighted graphs, where each node represents a super-pixel. To provide additional supervision from bounding box information, the authors introduce the multi-point (MP) loss specifically designed for the A2GNN method. For this work, image-level labels are used to generate the foreground using CAM inference, while bounding box labels are used to generate the background. In [19], a network is designed to produce CAMs and online accumulated class attention maps (OA-CAMs). In the OA-CAMs approach, the different parts of the target object from attention map are combined to improve poor CAM quality. However, in this solution, most of the attention is focused on enlarging salient regions around the target object while objects outside the salient region do not gather enough attention. To activate objects outside of the salient region, a graph-based global reasoning unit is integrated into the classification branch of the network. Furthermore, to enhance the quality of pseudo-labels, a potential object mining module (POM) and a nonsalient region masking module (NSRM) are employed. These modules combine semantic information of the target object and can generate pseudo-labels for the complex scenes in images.

Self-supervised equivariant attention mechanism (SEAM) [20], embedded discriminative attention mechanism (EDAM) [21], and image segmentation with iterative masking (ISIM) [22] are methods used for self-improvement in computer vision. SEAM uses a Siamese network that takes both original and augmented images as input to produce a CAM (class activation map) at the output. Each Siamese branch includes a pixel correlation module (PCM) that refines the CAM. The PCM module proposed by the authors is used to include low-level features in the CAM. The CAM and PCM module activation maps from the Siamese branches are regularized to ensure consistency. EDAM includes a discriminative activation layer (DA) after the backbone, as well as a collaborative multi-attention module (CMA). The DA layer predicts a class-specific mask for each category. Each mask is then multiplied with a feature map. The CMA module, which is located after the DA layer, applies a self-attention mechanism to explore activation maps of each category and extract common category-specific information from the images in the batch. These modules work together to improve the network's ability to discriminate between classes and attend to important features in the input. In the ISIM model, an input image and its corresponding image-level label are passed through an encoder network to extract a

CAM. Then, pseudo-segmentation labels are generated using the dense conditional random field (dCRF) algorithm, which is used to refine CAM quality. The model is retrained using these pseudo-segmentation labels as ground truth. A pixel-level loss function is used to activate less discriminative areas in the CAM inference. An iterative process is performed with a pixel-level loss, and a CAM threshold is set to optimize the final CAM result.

Another approach is to divide images into patches. In [23], the authors propose a complementary patch network (CPN). A CPN is formed by a triplet network with three branches. In the CPN, the original image is split into pairs of images with hidden parts, and the CAM is defined as the sum of the pair. To refine CAM results, the proposed pixel-region correlation module (PRCM) is used. This module finds semantic relations between regions or pixels and uses information with the help of the PCM module proposed in the ISIM work [22]. In the PPL [24] method, the image is split into patches. Each patch is fed to subsequent convolutional layers separately. In this case, the neural network has access only to the local features. It pushes the neural network to focus more attention on local features. The patch learning processing performs from low-level layers of the network to high-level layers. It allows focusing on low-level as well as high-level discriminative regions.

Several approaches to weakly supervised semantic segmentation (WSSS) utilize bounding boxes as annotations. In [25], foreground and background regions are extracted from the bounding boxes, and segmentation labels are obtained using CAM from the classification network, using background-aware pooling (BAP). CAM is applied for each bounding box. Finally, CNN is trained for semantic segmentation using noise-aware loss (NAL) to reduce the influence of noisy labels. In [26], foreground and background objects are considered as positive and negative instances, respectively. The multiple instances learning (MIL) loss is applied to the bounding boxes. Since bounding boxes usually include multiple foreground objects, it leads to classification problems. Therefore, the labeling-balance loss is used to overcome this drawback. Recent and most promising work describes the Segment Anything Model (SAM) developed [27] by Facebook. Images on the SAM input are fed to the image encoder that is based on the pretrained vision transformer and produces image embedding. Then, different kind of the prompts are used to map image embeddings into a mask. There are a few types of the prompts: points, boxes, and text. In the cases when prompt is quite ambiguous, SAM produces multiple masks with different confidence scores.

Other papers consist of different approaches to solving the WSSS task. The ACFN [28] model is based on atrous (dilated) convolution and includes two modules: the cascade module and the pyramid module. The cascade module is composed of three atrous convolutional layers inserted in the middle of the backbone network. The pyramid module is composed of four parallel atrous convolutional layers with different atrous rates, allowing it to learn different scales of context information. After the pyramid module, the image information of different scales is fused. The SLAM [29] framework contains two training stages. In the first stage, the semantic encoder is trained to learn the features of each category. In the second stage, the segmentation neural network is trained using the learned features of the semantic encoder. The AuxSegNet [30] is based on the cross-talk module that consists of three task-specific branches after the backbone. Since each branch is responsible for the specific type of learning (classification, saliency detection, semantic segmentation), the cross-talk affinity learning module learns task-specific affinities and features, which are used to enlarge the feature map produced by CAM for the saliency detection and semantic segmentation tasks. Then, these two task-specific affinity maps are used to produce a global cross-task affinity map. This affinity map is used to refine both saliency and segmentation predictions. In the CODNet [31] model, a pair of images is used as inputs, and common semantic features are extracted. For each location in the target images, features from a similar region in the reference images are extracted and concatenated. In [32], authors propose to erase misclassified regions of the CAM and then enlarge them properly. The contextual information captured by the semantic segmentation network is used as a guide to accurately erase the misclassified area in the CAM. Then, hierarchical deep seeded region growing (H-DSRG) is performed, accurately growing the semantic

regions by taking into account the spatial distance between regions. The HSPP [33] model consists of parallel branches of global average pooling and max pooling with different scales. Inferences from each branch are averaged. In addition, a visual word encoder (VWE) module is used to encode local visual words and improve CAM inference. TransCAM authors decided to use the Conformer network as a backbone. Conformer consists of two branches: transform and CNN. The CNN branch generates CAM, and the transform branch generates the attention map. Combining the attention map and CAM inference allows significant improvement of the quality of the CAM result. The solution demonstrated in the paper [34] is based on the antiadversarial method called AdvCAM. It manipulates an attention map of an image to improve the classification task inference. In the classic adversarial attack method, pixel-level perturbations are used to change the network output. AdvCAM allows for the involvement of more regions in an attention map and improves the CAM result.

2.2. Unsupervised Methods

In the LOST model [35], features are obtained from the visual transformer. The image is divided into patches and fed into the DINO model [36], which uses the visual transformer mechanism. Similarities among patches are computed, and by selecting a patch with the fewest similarities (seed), object parts are localized. Then, seed expansion is performed, which involves adding correlated patches to the initial seed. However, authors of the [37] paper claim that the attention map provided by LOST is noisy and have proposed a method called TokenCut to eliminate this issue. TokenCut is based on a graph where edges represent similarities between graph nodes. Segmentation of the foreground and background objects is performed by the normalized cut (Ncut) approach, which performs eigendecomposition. To select the foreground object, an assumption is utilized that the eigenvector of the foreground object is less than the background eigenvector. Another graph-based approach was proposed in [38] and utilizes eigenvalues. First, a weighted graph over image patches is constructed, where the graph edge weights show the affinity of the pair patches. This is the process of constructing a semantic affinity matrix for the image. The Laplacian eigenvectors of this matrix are calculated, and these eigenvectors can be used to produce a segmentation mask or bounding box. In the [39] paper, as well as in the LOST and TokenCut works, they introduced a network for the object detection task. The network consists of foreground and background models. In the foreground model, the feature map generator produces a feature map and scalar attention map. These maps are used to predict object scales and positions. The background model is an autoencoder that tries to learn the image background. In the CCAM paper [40], a model is proposed to produce cues that can be used by other models to improve results. In the CCAM model, images are fed to the autoencoder, and features are extracted to produce a class-agnostic activation map. Then, contrast learning is applied to distinguish foreground and background. CCAM only predicts one activation map to indicate foreground and background regions in an image. In the case where the background or foreground has complex colors or texture, the rank weighting is designed to reduce the influence of dissimilarities. CCAM can be used to improve CAM or object localization.

2.3. Few-Shots Methods

Segmentation tasks are not able to tackle the new and unseen during training classes. In order to eliminate this issue, the few-shot learning was introduced. It can be used to construct class-agnostic segmentation models that adjust to the new classes. In few-shot learning, support datasets are used to assist the model in learning and generalizing to new tasks or data. The support dataset contains an extremely small number of labeled examples for each specific task or class of interest. Besides support images, there is query image term that refers to the image for which the model needs to generate segmentation masks.

One of the promising frameworks [41] utilizes singular value decomposition (SVD) matrices. Since the amount of the support data is too small, the model can experience

overfitting. However, an unfreezed backbone with fine-tuning of a small amount of the backbone parameters helps to avoid the overfitting issue. To define these tunable parameters, all of the pretrained parameter are decomposed by the SVD. When only the singular matrix is fine-tuned, the other matrix values remain frozen. This approach is called singular value fine-tuning (SVF).

Another idea is demonstrated in the Multi-Similarity and Attention Network (MSANet) [42]. The pretrained backbone is used to extract features both query and support images. Then, these features are fed to the attention and multisimilarity modules where attention maps are produced and visual affinities are found in both of the images that are used in the process of obtaining final mask prediction.

In [43], foreground as well as background information in the support image is fully exploited. For this purpose, they proposed a dense pixel-wise cross-query-and-support attention-weighted mask aggregation (DCAMA) approach. Similarities and dissimilarities between query and support images are given different weight. Semantically similar pixels are given more weight than unlike pixels.

In [44], authors were faced with the issue that novel classes obtain lower activation than known ones. They proposed a hierarchically decoupled matching network (HDMNet). In this model, they used an extended transformer architecture. In this architecture, embedded correlation mechanism and correlation map distillation are used to extract more semantic information and eliminate the overfitting problem.

The most recent and high-performance approach [45] is based on the generative pretrained transformer (GPT) language model. The proposed segmentation GPT (SegGPT) framework can be applied to the various spectrum of the computer vision tasks such as video object segmentation, semantic segmentation, panoptic segmentation, and few-shot segmentation. The key feature of this model is that it does not require additional fine-tuning and still can show superior performance on the listed range of tasks.

The discussed papers are the most recent and provide some of the best results. They are focused on the semantic segmentation, object detection, and saliency detection tasks. Since in WSSS bounding boxes can be used as a labels, such techniques as object detection and saliency detection can be applied to the WSSS task. Table 1 summarizes the results provided by the literature overview of the methods where validation was performed on the PASCAL VOC 2012 dataset, and the key metric is mIoU. All of the papers in Table 1 are devoted to the semantic segmentation task.

Table 1. Results provided by weakly supervised studies on the validation PASCAL VOC 2012 dataset in terms of the mIoU.

Name	Type of the Annotations	mIoU, %
Weakly Supervised Semantic Segmentation via Progressive Patch Learning [24]	Image-level labels	67.8
SLAM: Semantic Learning based Activation Map for Weakly Supervised Semantic Segmentation [29]	Image-level labels	70.8
Co-attention dictionary network for weakly-supervised semantic segmentation [31]	Image-level labels	64.5
Erase then grow: Generating correct class activation maps for weakly-supervised semantic segmentation [32]	Image-level labels	66.8
Atrous convolutional feature network for weakly supervised semantic segmentation [28]	Image-level labels	66.0
Leveraging Auxiliary Tasks with Affinity Learning for Weakly Supervised Semantic Segmentation [30]	Image-level labels	69.0

Table 1. *Cont.*

Name	Type of the Annotations	mIoU, %
Embedded Discriminative Attention Mechanism for Weakly Supervised Semantic Segmentation [21]	Image-level labels	70.6
Learning Visual Words for Weakly-Supervised Semantic Segmentation [33]	Image-level labels	67.2
Background-Aware Pooling and Noise-Aware Loss for Weakly-Supervised Semantic Segmentation [25]	Bounding box labels	78.7
Affinity Attention Graph Neural Network for Weakly Supervised Semantic Segmentation [18]	Bounding box and Image-level labels	76.6
Delving Deeper into Pixel Prior for Box-Supervised Semantic Segmentation [26]	Bounding box labels	75.8
TransCAM: Transformer Attention-based CAM Refinement for Weakly Supervised Semantic Segmentation [46]	Image-level labels	69.3
Complementary Patch for Weakly Supervised Semantic Segmentation [23]	Image-level labels	67.8
Non-Salient Region Object Mining for Weakly Supervised Semantic Segmentation [19]	Image-level labels	70.4
Self-supervised Equivariant Attention Mechanism for Weakly Supervised Semantic Segmentation [20]	Image-level labels	64.5
Anti-Adversarially Manipulated Attributions for Weakly and Semi-Supervised Semantic Segmentation [34]	Image-level labels	68.0
ISIM: Iterative Self-Improved Model for Weakly Supervised Segmentation [22]	Image-level labels	70.38

3. Materials and Methods

3.1. Dataset Overview

For this research we have collected a dataset. This dataset contains the following category types of the plants:

- Cassava leaf disease (8 test and 17 train images);
- Corn leaves pathology (23 test and 52 train images);
- Fruit plants (12 test and 24 train images);
- Herbarium (24 test and 56 train images);
- Plant pathology (24 test and 56 train images);
- Tomato plants (24 test and 56 train images);
- Wild edible plants (25 test and 59 train images);
- Flowers (16 test and 39 train images).

In addition to these categories, each plant in the dataset is divided into individual parts (subcategories):

- Stem;
- Leaf;
- Fruit;
- Flower;
- Root.

Figure 1 shows an example of the instance masks in the dataset images, where each plant object has a mask, a bounding box, and an ID. These properties enable obtaining a mask and a bounding box of a whole plant from several parts. Moreover, each category in the dataset is already partitioned into train and test folders. This structured organization facilitates the utilization of the dataset for various tasks ranging from classification to part co-segmentation, i.e., the segmentation of individual parts of an object.

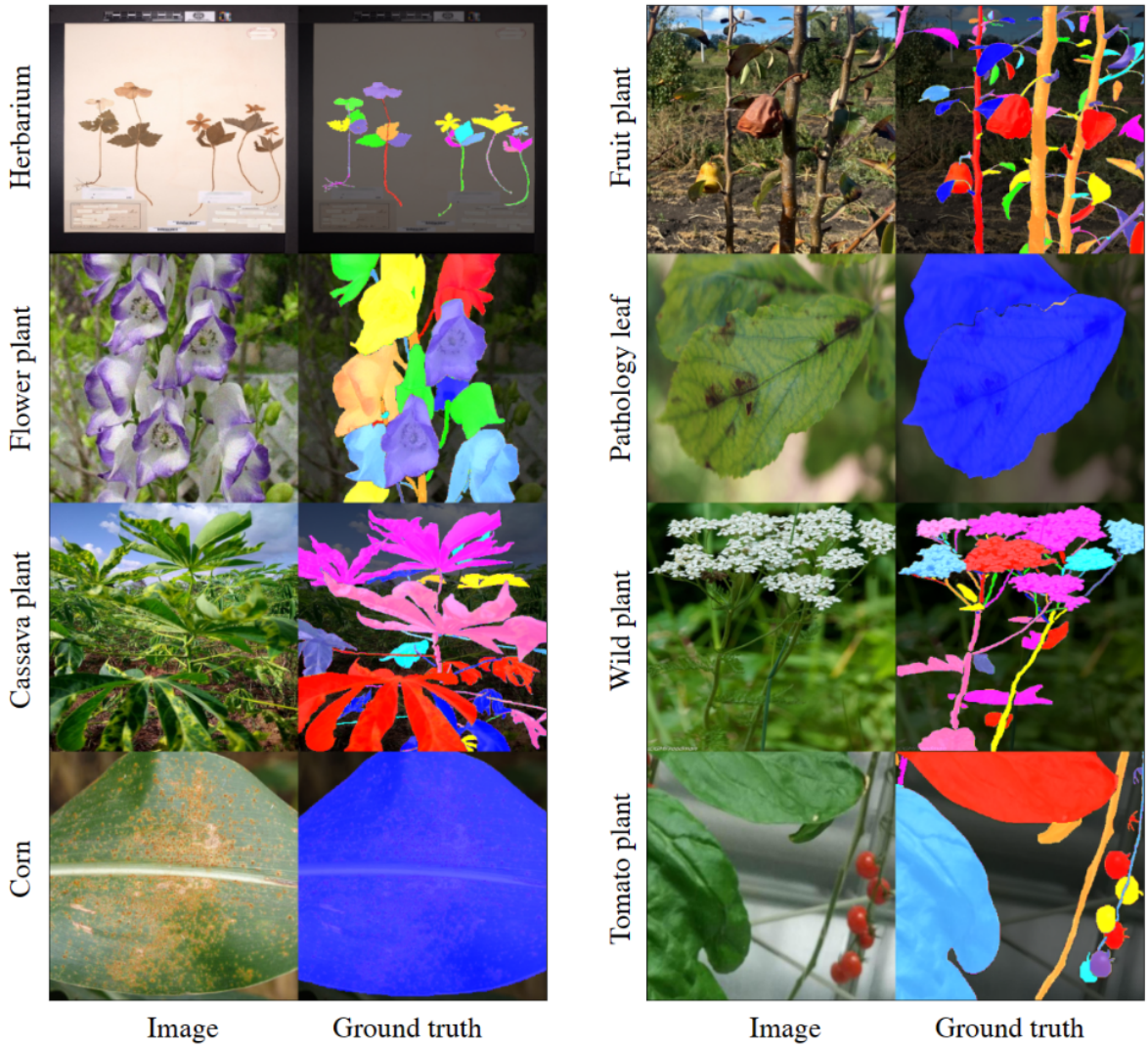

Figure 1. Example of the dataset annotations.

The primary challenge in working with the provided images is the large number of intersecting objects such as stems and leaves. This is a common scenario in real-world data, particularly for plants. As a result, one object may consist of multiple masks, which complicates the training process of neural networks and adds to the difficulty of the WSSS task, in addition to the reasons mentioned in the Introduction.

3.2. Methods

In order to obtain pseudo-instance masks, two approaches were used. The first approach involved a combination of two methods: TransCAM [46] and MiDaS [47]. TransCAM is a vision Transformer method used to obtain pseudo-semantic masks from image-level labels by thresholding the class attention map (CAM) using Otsu. The TransCAM network is a deep learning architecture for image classification. It combines the strengths of CNNs and vision transformer models. The architecture of TransCAM involves integrating CNN-based feature extraction with transformer-based attention mechanisms. It utilizes a CNN conformer backbone to extract image features. These features are then fed into a transformer encoder–decoder architecture. The transformer encoder–decoder enables capturing global context and long-range dependencies in the image. TransCAM introduces the concept of CAM into transformers to generate attention maps specific to each class. This allows the network to focus on discriminative regions during classification. The attention

maps are used to weight the CNN features, enabling the network to attend to relevant image regions.

MiDaS (monocular depth estimation) is a depth estimation method used to generate depth maps from 2D images. It is based on a deep neural network architecture that can estimate depth information by analyzing the visual cues in a single image. MiDaS uses a multiscale feature pyramid network to capture information at different levels of detail. It leverages a combination of low-level image features and high-level semantic information to estimate depth. MiDaS takes advantage of both monocular and stereo depth cues to improve the accuracy of depth estimation. It can handle challenging scenarios such as occlusions and textureless regions by incorporating contextual information. The output of MiDaS is a dense depth map, where each pixel represents the estimated depth value. The depth map in this work is used for depth-aware pseudo-semantic masks editing.

Utilizing image depth to obtain pseudo-masks is not a novel approach. In [48], image depth was used for the hand part segmentation purpose. Individual hand parts were distinguished based on its depth level. In [49], an image depth also was used in combination with CAM. However, in this paper, an image depth was incorporated into the segmentation loss function, rather than being utilized for converting pseudo-semantic masks into pseudo-instance masks.

Figure 2 presents a visualization of the CAM for different classes provided by TransCAM.

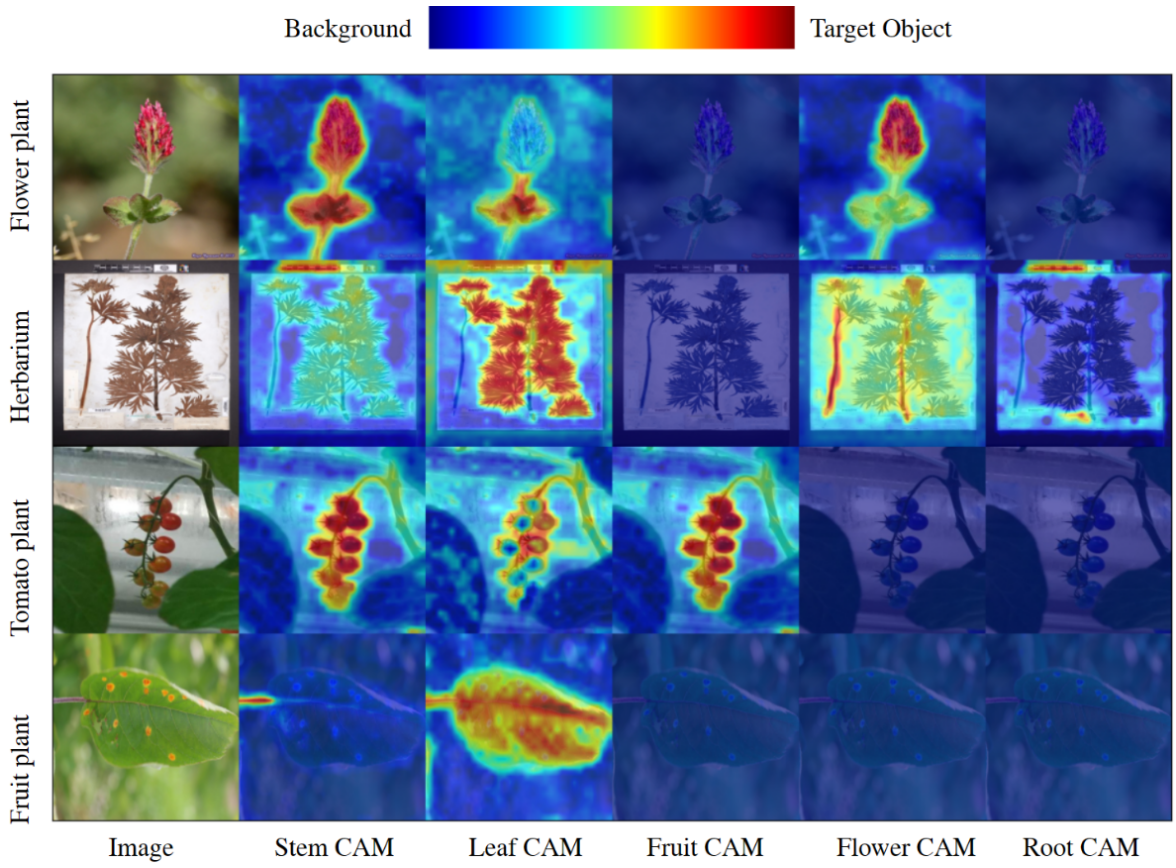

Figure 2. Examples of attention maps for different classes.

Since MiDaS computes relative depth in an image, it can be used to estimate depth in each bounding box. Based on the calculated depth, the TransCAM pseudo-semantic mask is transformed into a pseudo-instance mask. Figure 3 demonstrates the process of obtaining pseudo-instance masks.

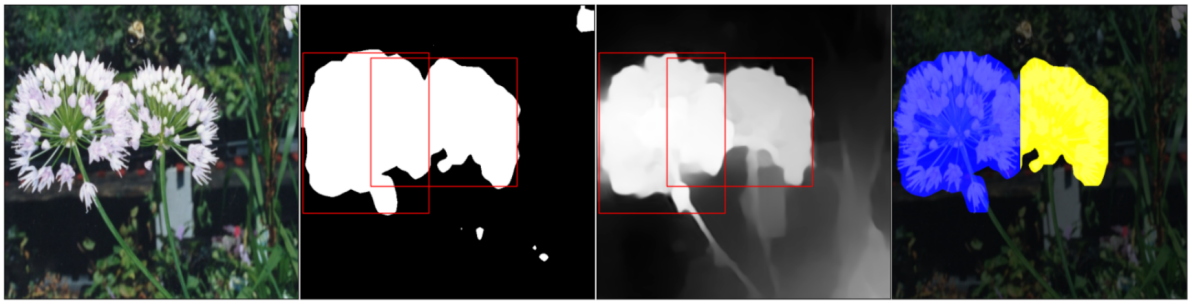

Figure 3. Visualization of the proposed approach.

To generate the pseudo-semantic mask, the original image (a) is used with TransCAM to generate the mask (b). The bounding boxes are used to discard wrong activations that are placed outside the box boundaries and divide the semantic mask into individual instances. However, the issue of box intersections arises. To address this issue, the decision algorithm was used. MiDaS is applied to the image (c) to obtain an image depth map. The mean depth value in the intersected area is compared to the mean depth value in both boxes. In this example, the depth in the left box is closer to the depth in the intersected area, indicating that the intersection area belongs to the left bounding box (d). To better understand the procedure of obtaining pseudo-instance masks, Figure 4 shows the algorithm procedure step by step.

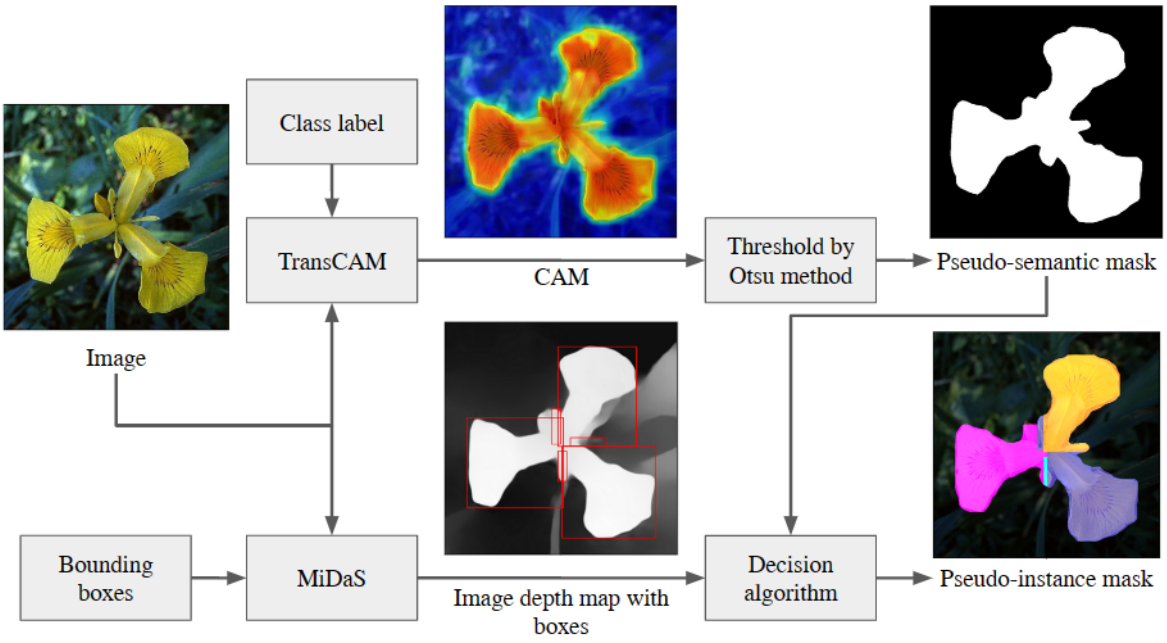

Figure 4. Algorithm for transforming pseudo-semantic mask to the pseudo-instance mask.

The second approach in this study utilizes the zero-shot method Segment-Anything (SAM) [27] from Meta, which can take a batch of bounding boxes and predict masks. For each image, the bounding boxes corresponding to the presented categories are extracted from annotations, and a batch of boxes for each category is used to obtain segmentation masks. This process is illustrated in Figure 5. In this work, the SAM method was used in the zero-shot mode.

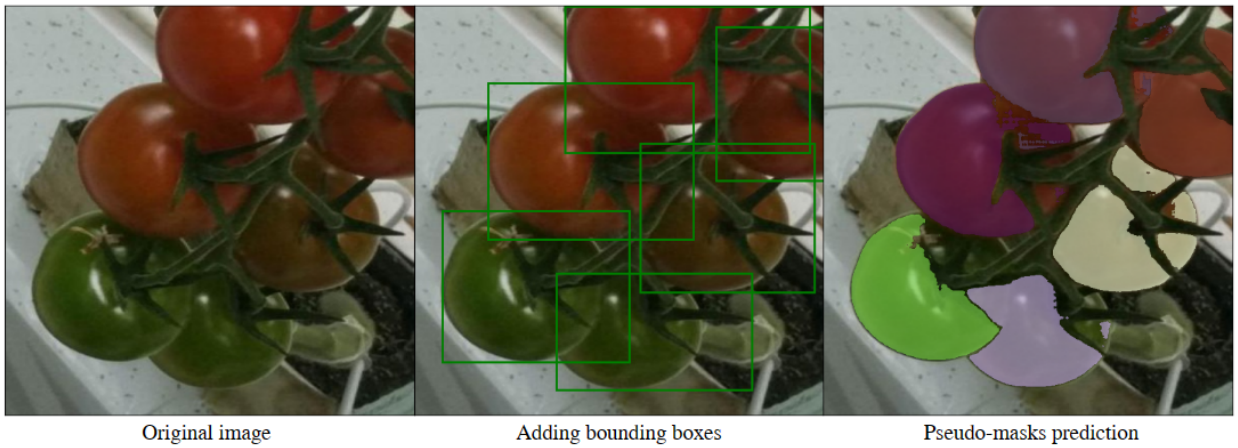

Figure 5. Illustration of the process for obtaining pseudo-masks using SAM.

The segmentation is applied to both subcategories (individual plant parts) and categories (full plants). In both cases, segmentation is performed using bounding box, pseudo-instance, and ground truth masks. The use of bounding boxes serves as a baseline [50] for comparison to other solutions. In this approach, the whole area inside each bounding box is considered as a mask of the corresponding object. The aim of this study is to determine the feasibility of using weak supervision. The pseudo-instance masks are considered qualitatively better if they yield higher metrics compared to the bounding boxes baseline. The closer the metrics obtained from pseudo-labels are to the ground truth metrics, the better the final result.

In addition to these results, the pseudo-instance masks combined with the OBA are used. This approach aims to prove that the weak supervision techniques can be improved and provide sufficient result with cost reduction of the dataset annotation.

3.3. Evaluation Metrics

To estimate performance of weak supervision, the mean average precision (mAP) will be calculated on the test part of the dataset as the evaluation metric (1):

$$mAP = \frac{1}{C}\sum_{c}^{C} AP_c \tag{1}$$

where

C—number of the classes;

AP_c —averaged precision for the c-th class.

The AP for each object class was computed using the formula shown in Equation (2).

$$AP_c = \frac{1}{N_c}\sum[TP(i)/(TP(i) + FP(i))] \tag{2}$$

where

i—intersection-over-union threshold;

TP—the number of true positives for the i-th threshold value;

FP—the number of false positives for the i-th threshold value;

N_c—the total number of class objects in the dataset.

In the computer vision tasks, the mAP is dependent on the intersection-over-union (IoU) metric.

As a loss function in TransCAM, weemployed multi-label soft margin loss (3).

$$loss(x, y) = -\frac{1}{C}\sum_{i} y[i]\log\left(\frac{1}{1+\exp(-x[i])}\right) + (1 - y[i])\log\left(\frac{\exp(-x[i])}{1+\exp(-x[i])}\right) \tag{3}$$

where

C—number of the classes;
y—target value;
x—input value.

3.4. Experiment Settings

In this study, a YOLOv8 instance segmentation network is utilized, which is known for its fast one-shot learning capabilities. YOLOv8 stands for "You Only Look Once version 8". It is an improved version of the YOLO (You Only Look Once) family of models. YOLOv8's architecture is based on a deep convolutional neural network. This model uses a single neural network to predict bounding boxes and class probabilities simultaneously. The network architecture consists of multiple convolutional layers, followed by fully connected layers. YOLOv8 utilizes anchor boxes to improve the accuracy of object detection. It uses a feature pyramid network to detect objects at different scales. YOLOv8 incorporates Darknet-53 as its backbone network. It achieves real-time object detection by dividing the input image into a grid and making predictions for each grid cell. YOLOv8 employs nonmaximum suppression to remove duplicate detections. Besides object detection, YOLOv8 can be used for the instance segmentation purpose. One of the interesting features of the YOLO family is the ability to calculate metrics such as mAP@0.5 and mAP@0.5:0.95 for both bounding boxes and segmentation masks. The hyperparameters for TransCAM and YOLOv8 are provided in Table 2 and Table 3, respectively. To account for the wide range of resolutions in the dataset, all images were resized to 640 × 640 resolution. The MiDaS architecture used in this work is MiDaS v2.1-small. The choice of the pretrained weights for the TransCAM and YOLOv8 models, as well as choice of the MiDaS architecture, was driven by limited computational resources at our disposal. The learning rate was tuned to provide smooth calculation of the loss function. Batch size choice was also driven by limited resources. Other parameters values were set by default.

Table 2. TransCAM parameters.

Parameter Name	Parameter Value
Batch size	8
Epochs	80
Optimizer	AdamW
Learning rate	2×10^{-6}
Weight decay	5×10^{-4}
Epsilon	1×10^{-8}
Image size	640 × 640
Pretrained model	Conformer-small-patch16

Table 3. YOLOv8 parameters.

Parameter Name	Parameter Value
Batch size	6
Epochs	80
Optimizer	Adam
Learning rate	4×10^{-3}
Image size	640 × 640
Pretrained model	yolov8n-seg

4. Results

Tables 4 and 5 compare the mAP@0.5 and mAP@0.5:0.95 of the predictions provided by YOLOv8 trained on different types of labels for the instance segmentation task. The results of both tables show the relative percentage gain in metrics. The gray color in these tables represents the baseline case where the bounding box is used as the segmentation mask.

Table 4. Instance segmentation metrics for object parts case with using different types of segmentation masks.

Object Parts	Annotation Source							
	Bounding-Box-Based		TransCAM+MiDaS-Based		SAM-Based		Ground Truth	
	mAP@50	mAP@50:95	mAP@50	mAP@50:95	mAP@50	mAP@50:95	mAP@50	mAP@50:95
All	13 ± 0.3	4.5 ± 0.2	13.6 ± 0.7 (+4%)	5.4 ± 0.13 (+20%)	16 ± 0.4 **(+23%)**	7.5 ± 0.2 **(+66%)**	16.7 ± 1 (+28%)	8.4 ± 0.4 (+86%)
Stem	0.08 ± 0.014	0.02 ± 0.004	0.05 ± 0.008 (−33%)	0.01 ± 0.01 (-26%)	2 ± 0.5 **(+2630%)**	0.55 ± 0.15 **(+2821%)**	1.6 ± 0.1 (+1914%)	0.4 ± 0.02 (+2199%)
Leaf	11 ± 0.7	2.6 ± 0.2	7.4 ± 2 (−33%)	2 ± 0.4 (−24%)	20 ± 1.6 **(+73%)**	8 ± 0.6 **(+190%)**	19 ± 2 (+75%)	8 ± 0.8 (+205%)
Fruit	47 ± 4	19.4 ± 2	49 ± 2.6 (+5%)	23 ± 1 (+18%)	49.5 ± 1.5 **(+5.5%)**	26 ± 1.5 **(+36%)**	50 ± 2.3 (+10%)	29 ± 1.3 (+53%)
Flower	19 ± 1.6	4.8 ± 0.6	24 ± 1.5 (+24%)	8 ± 0.6 (+60%)	24 ± 2.2 **(+28%)**	10 ± 1 **(+110%)**	25.4 ± 4 (+33%)	11.4 ± 2.5 (+136%)
Root	2.8 ± 1.7	0.8 ± 0.5	3 ± 2.4 **(+11%)**	0.7 ± 0.4 **(−19%)**	2 ± 1.6 (−22%)	0.4 ± 0.7 (+20%)	11 ± 6 (+308%)	5 ± 3 (+490%)

Note: Gray color in the table emphasizes the baseline case. Bold numbers demonstrate the best result.

Table 5. Instance segmentation metrics for full objects with using different types of segmentation mask.

Dataset	Annotation Source							
	Bounding-Box-Based		TransCAM+MiDaS-Based		SAM-Based		Ground Truth	
	mAP@50	mAP@50:95	mAP@50	mAP@50:95	mAP@50	mAP@50:95	mAP@50	mAP@50:95
All	12 ± 0.7	5 ± 0.5	16.8 ± 1.3 **(+40%)**	8.1 ± 0.75 **(+59%)**	12.1 ± 2.3 (+0.5%)	4.8 ± 1.3 (−4%)	24 ± 4 (+103%)	12.4 ± 2.5 (+144%)
Cassava plants	0.2 ± 0.17	0.07 ± 0.08	0.1 ± 0.1 (−43%)	0.05 ± 0.07 (−27%)	1.2 ± 1.6 **(+608%)**	0.2 ± 0.2 **(+148%)**	0.2 ± 0.16 (+4%)	0.06 ± 0.07 (−8%)
Corn leaves	49.7 ± 1.3	28.8 ± 7	41.6 ± 11.6 (−16%)	23 ± 9 **(−18%)**	44.8 ± 15 **(−10%)**	22 ± 10 (−23%)	59 ± 10 (+19%)	39 ± 9 (+37%)
Fruit Plants	0.06 ± 0.07	0.01 ± 0.01	0.03 ± 0.016 (−57%)	0.01 ± 0.005 (−10%)	0.1 ± 0.1 **(+42%)**	0.02 ± 0.02 **(+128%)**	0.02 ± 0.01 (−61%)	0.01 ± 0.007 (+29%)
Herbarium plants	0.22 ± 0.1	0.04 ± 0.03	0.66 ± 0.17 (+202%)	0.15 ± 0.05 (+317)	6.8 ± 2.6 **(+3068%)**	2 ± 1.1 **(+5414%)**	22 ± 6 (+10089%)	8 ± 2 (+21976%)
Leaves with pathology	27.7 ± 9	7 ± 4	48 ± 5 **(+73%)**	27 ± 7 **(+292%)**	22.5 ± 3.6 (−18%)	9 ± 2.2 (+38%)	52 ± 10 (+90%)	31 ± 6 (+355%)
Tomato plants	2.1 ± 0.6	0.3 ± 0.1	1.7 ± 0.4 (−17%)	0.3 ± 0.07 (+9%)	3.9 ± 4 **(+90%)**	1.1 ± 1 **(+314%)**	14.6 ± 1.8 (+603%)	3.9 ± 0.49 (+1289%)
Wild Edible Plants	0.88 ± 0.8	0.12 ± 0.1	7.8 ± 4.9 **(+789%)**	1.45 ± 0.8 **(+1074%)**	4 ± 1 (+361%)	0.8 ± 0.2 (+561%)	14.7 ± 6 (+1582%)	4.3 ± 1.9 (+3361%)
Flower plants	14.96 ± 12	4.15 ± 2.7	34.3 ± 8 **(+129%)**	13.6 ± 3 **(+229%)**	13 ± 4 (−12%)	3.2 ± 1 (−22%)	31 ± 14 (+106%)	12 ± 4.8 (+186%)

Note: Gray color in the table emphasizes the baseline case. Bold numbers demonstrate the best result.

Table 4 shows that for plant part segmentation, SAM-based annotations work better for the most of the tasks. This approach increases mAP from 13 to 16 compared with the baseline. This is close to the result of a model, trained on the real part segmentation masks. However, we must note that for very thin objects such as plant roots, the SAM-based approach is weaker than the baseline. Therefore, for this class it is more suitable to use TransCAM with MiDaS masks.

We can also observe that metrics for the stem category provided by the SAM method outperform the ground truth result. Figure 6 shows SAM pseudo-masks for the stem category.

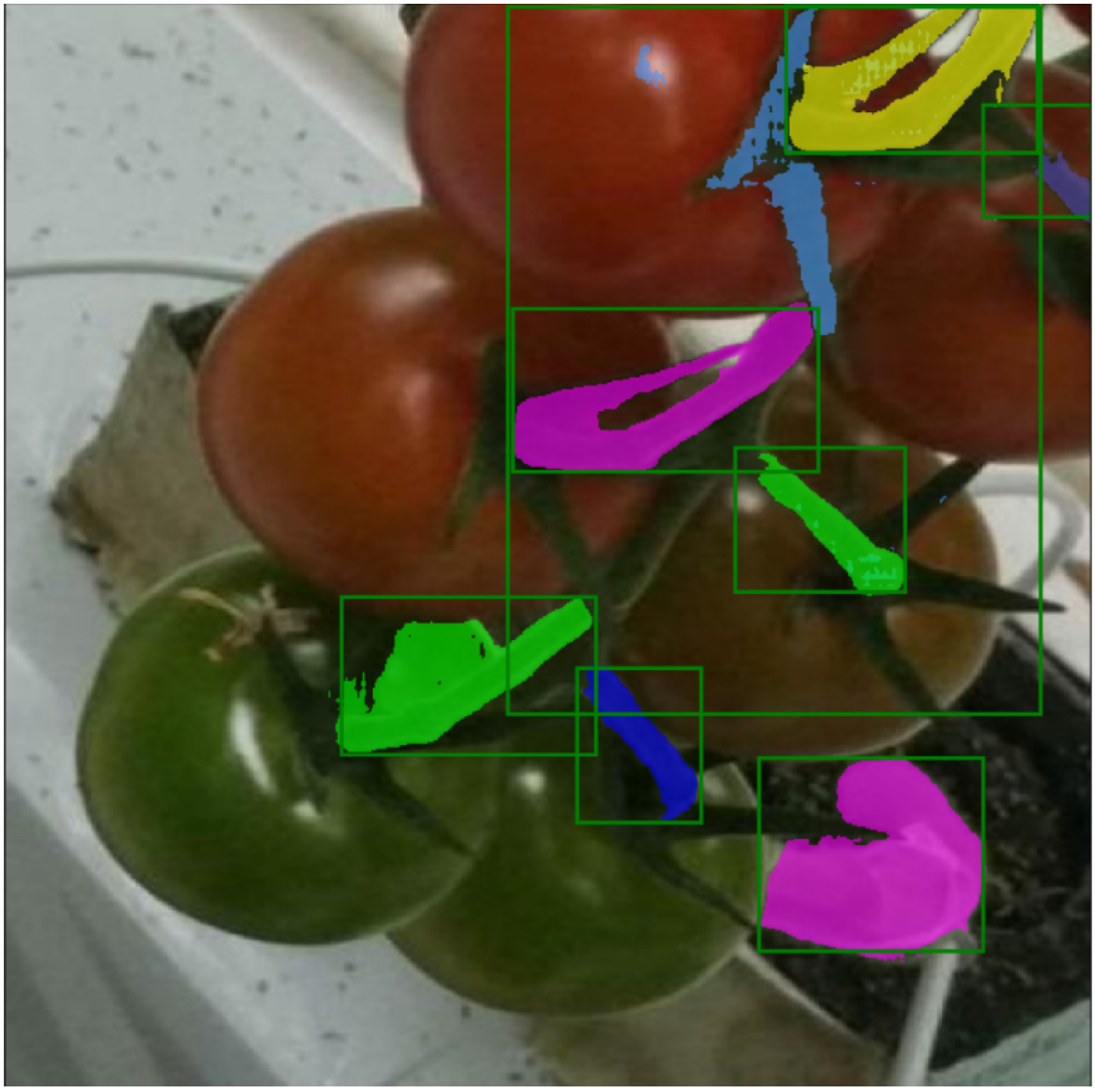

Figure 6. Pseudo-masks for the stem category provided by SAM.

Figure 6 reveals that, apart from the target objects, SAM also detects tomato stalks placed near to the stem due to their similar semantic structure to the stems. Consequently, the SAM pseudo-masks provide additional information about the target category by highlighting semantically similar objects within the image.

Table 5 shows the results of full plant segmentation. With this objective, a model trained with masks, obtained with TransCAM and MiDaS, generally works better. On average, it provides a 40% relative increase in mAP. This approach significantly beats SAM-based masks because MiDaS depth masks allow us to distinguish the borders of

overlapping objects better. The only exception here is the result on the Herbarium plants dataset. The reason for this is the simplicity of this dataset. It has a single plant on each image, and the background is always uniform.

Tables 6 and 7 display the results of combinations of object-based augmentation and SAM techniques applied to the object parts and full plants. In the brackets are shown gains in metrics related to the original SAM results from Tables 4 and 5.

Table 6. Instance segmentation metrics for object parts case using the OBA and SAM approach.

Object Parts	Annotation SOURCE			
	SAM without Object-Based Augmentation		SAM with Object-Based Augmentation	
	mAP@50	mAP@50:95	mAP@50	mAP@50:95
All	16 ± 0.4	7.5 ± 0.2	20.24 ± 1.26 **(+26%)**	10.1 ± 0.7 **(+34%)**
Stem	2 ± 0.5	0.55 ± 0.15	3.85 ± 0.32 **(+81%)**	1.14 ± 0.09 **(+107%)**
Leaf	20 ± 1.6	8 ± 0.6	24.1 ± 1.7 **(+25%)**	10 ± 0.9 **(+32%)**
Fruit	49 ± 1.5	26 ± 1.5	57 ± 1.6 **(+17%)**	33.3 ± 0.8 **(+26%)**
Flower	24 ± 2.2	10 ± 1	28.1 ± 3.5 **(+15%)**	11.8 ± 2.2 **(+16%)**
Root	2 ± 1.6	0.4 ± 0.7	7.8 ± 4.9 **(+809%)**	4.3 ± 4.4 **(+976%)**

Note: Percentage gain in the table is related to the SAM approach without augmentation.

Table 7. Instance segmentation metrics for the full plants in the case when using OBA and SAM approach.

Dataset	Annotation Source			
	SAM without Object-Based Augmentation		SAM with Object-Based Augmentation	
	mAP@50	mAP@50:95	mAP@50	mAP@50:95
All	12.8 ± 2.3	4.8 ± 1.3	20.7 ± 2.4 **(+72%)**	10.6 ± 2.1 **(+117%)**
Cassava plants	1.2 ± 1.6	0.2 ± 0.2	0.98 ± 0.76 (−18%)	0.47 ± 0.29 **(+179%)**
Corn leaves	44.8 ± 15	22 ± 10	62.5 ± 6.26 **(+39%)**	47 ± 6.4 **(+115%)**
Fruit Plants	0.1 ± 0.1	0.02 ± 0.02	0.34 ± 0.15 **(+285%)**	0.1 ± 0.05 **(+360%)**
Herbarium plants	6.8 ± 2.6	2 ± 1.1	12.5 ± 2.3 **(+82%)**	4.3 ± 0.7 **(+112%)**
Leaves with pathology	22.5 ± 3.6	9 ± 2.2	41.8 ± 12.9 **(+85%)**	9.5 ± 6 **(+95%)**
Tomato plants	3.9 ± 4	1.1 ± 1	7.6 ± 3.7 **(+92%)**	1.9 ± 0.9 **(+63%)**
Wild Edible Plants	4 ± 1	0.8 ± 0.2	8.7 ± 2.6 **(+116%)**	2.2 ± 0.87 **(+168%)**
Flower plants	13 ± 4	3.2 ± 1	31.1 ± 7.8 **(+137%)**	9.9 ± 4.2 **(+207%)**

Note: Percentage gain in the table is related to the SAM approach without augmentation.

In Tables 6 and 7 one can see that object-based augmentation boosts the performance of weakly supervised solutions even further. In these results, a 26% increase in mAP for the object part case and a 72% increase for the full plant case are observed. Obtained metrics verify that the OBA approach significantly improves the performance on instance segmentation task when using pseudo-masks and a small-sized dataset. However, it should be noted that for the plants with rich morphological structure and overlapping objects, such as plants from the Cassava plants dataset, the given approach is faced with challenges and provides lower metrics than the original approach.

The following dependencies were observed: TransCAM and MiDaS exhibit superior results for the whole plant case, while SAM performs well for the part plant case. These approaches have different strengths and weaknesses. Consequently, we decided to construct a meta-algorithm combining TransCAM and MiDaS with SAM to complement mutual weaknesses and achieve improved performance.

The core of the meta-model is the Passive Aggressive Classifier (PAC) from the Scikit-Learn package. We utilized SAM pseudo-instance masks and TransCAM heatmaps as input for PAC, with ground truth masks as the desired output.

For the dataset, we selected 50 images on which we conducted PAC training, using the hinge loss function. The proposed meta-algorithm was employed to generate annotations for the dataset. Subsequently, YOLOv8 was trained on the obtained annotations. The results of comparing the quality of the meta-algorithm with the TransCAM and MiDaS approach are presented in Table 8. The experiment was conducted on the full plant case, where the TransCAM with MiDaS approach demonstrated the best results.

Table 8. Metrics for the full plants segmentation using meta-model.

Dataset	Annotation Source			
	TransCAM and MiDaS		Meta-Model	
	mAP@50	mAP@50:95	mAP@50	mAP@50:95
All	16.8 ± 1.3	8.1 ± 0.75	17.76 ± 1.5 **(+5.5%)**	8.58 ± 1.2 **(+6%)**
Cassava plants	0.1 ± 0.1	0.05 ± 0.07	0.05 ± 0.04 (−48.6%)	0.04 ± 0.04 (−16.6%)
Corn leaves	41.6 ± 11.6	23 ± 9	42.4 ± 22 **(+2%)**	39.1 ± 16 **(+66%)**
Fruit Plants	0.03 ± 0.016	0.01 ± 0.005	0.02 ± 0.03 (−20%)	0.016 ± 0.03 **(+98%)**
Herbarium plants	0.66 ± 0.17	0.15 ± 0.05	5.3 ± 3 **(+709%)**	3.5 ± 3 **(+3.5%)**
Leaves with pathology	48 ± 5	27 ± 7	42.6 ± 5 (−11%)	30 ± 13 **(+10%)**
Tomato plants	1.7 ± 0.4	0.3 ± 0.07	6.7 ± 2.2 **(+287%)**	3 ± 2 **(+907%)**
Wild Edible Plants	7.8 ± 4.9	1.45 ± 0.8	5.5 ± 2.8 (−29%)	2.6 ± 2.6 **(+85%)**
Flower plants	34.3 ± 8	13.6 ± 3	31.4 ± 6 (−8.6%)	21.2 ± 13 **(+55%)**

Note: Percentage gain in the table is related to the TransCAM and MiDaS approach.

In Table 8, one can see that the meta-algorithm outperforms the TransCAM and MiDaS approach for the full plant case by 5%. Combining two algorithms to eliminate their weaknesses can be a promising approach. By leveraging the strengths of each algorithm and compensating for their limitations, the resulting combination has the potential to achieve improved performance and robustness. The proposed approach allows for a more comprehensive solution that addresses multiple aspects of the observed problems.

In order to prove statistical significance of the obtained metrics, we used the Kruskal–Wallis test [51]. This method is used to determine if there are significant differences in the medians of three or more independent groups. In our experiment, we divided our data into five folds and performed the training process five times in a row for every type of the mask, creating tables with metrics and averaged metrics value over five experiments.First of all, we calculated gain in mAP related to the bounding-box-based masks (baseline case) in every experiment. Then, we calculated p-value using Kruskal–Wallis test for these metrics gains for the following types of masks: TransCAM and MiDaS pseudo-masks, SAM pseudo-masks, and meta-model pseudo-masks. We wanted to prove that improvements in the metrics provided by the proposed and considered methods were not a coincidence. We set a significance level, also known as alpha, to the 0.05 value. It is a threshold used to reject the null hypothesis that considered data groups have the same median value. It represents the maximum acceptable probability of observing a result as extreme as, or more extreme than, the observed result, assuming the null hypothesis is true. The obtained p-value from the Kruskal–Wallis test is equal to 0.017 and it is lower than the significance level of 0.05. In the conditions of a small amount of statistical data, it is sufficient evidence that our results have different statistical parameters and the obtained metrics values for the considered methods relative to the baseline case are not a coincidence.

Figure 7 gives the comparative qualitative results obtained with different kinds of masks used.

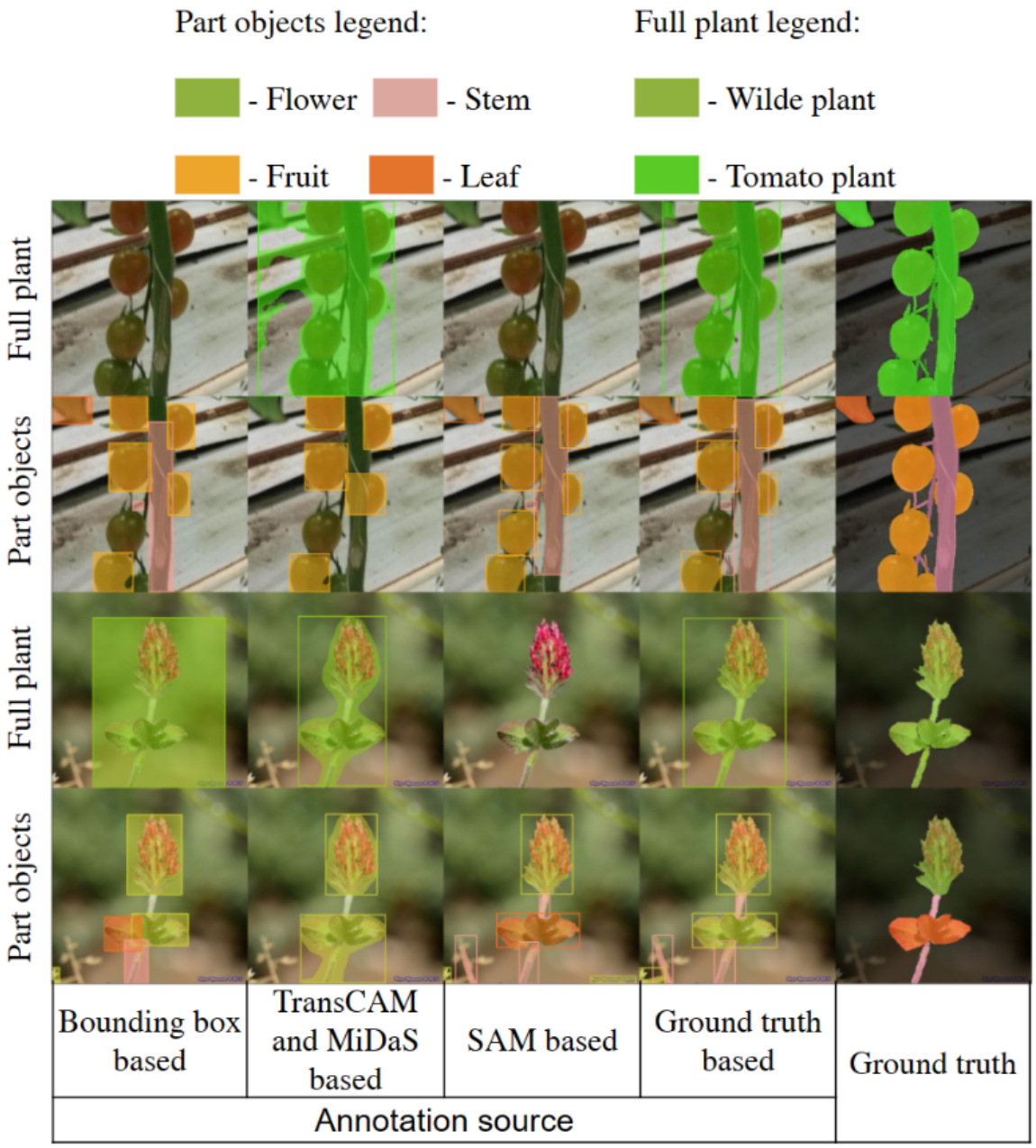

Figure 7. Predictions of the instance segmentation model trained on the different types of labels.

5. Discussion

In this paper, we show that one can segment plants and even plant parts without plant segmentation masks. The results obtained in this study demonstrate the effectiveness of pseudo-instance masks compared to bounding boxes. Almost all categories showed positive gains in the mAP metric when pseudo-instance masks were used. The results provided in Tables 6 and 7 exhibit an opportunity to use pseudo-masks and OBA to improve the quality of the training process in the conditions of a limited amount of annotations and a small-sized dataset.

However, some results also point to significant drawbacks of weak supervision. According to Tables 4 and 5, the most dramatic results are observed in the stems and cassava plants categories, which have extremely small mAP values. One of the reasons is that images related to the cassava plants category consist mainly of tiny and thin stems, which are challenging category objects. Weak supervision attention is focused on the most discriminative regions, such as fruits, and the thin stem is a less noticeable object. Furthermore,

cassava plant images have the largest amount of intersected masks, which significantly increases the complexity of this category.

It is also noticeable that not only stems and leaves have lower activation than fruits. In the case of fruits or flowers, the attention is mainly focused on more colorful instances, where red tomatoes, for example, receive more activation than pale ones. Additionally, in some images, stems are located very close to fruits or leaves, making them difficult to distinguish from other objects, leading to misclassifications.

Besides cases where the model recognizes an object from a different category, predictions can be completely wrong. As previously mentioned, weak supervision networks prefer discriminative regions with similar shapes, which can lead to attention being focused on repeated background patterns. This can degrade the quality of the predictions and even corrupt the pseudo-mask.

We want to note that segmentation and image processing [52–54], as well as signal processing [55,56], are important research areas that must receive enough attention in the current studies. These papers demonstrate the importance of further innovation in these areas in order to help in the development of artificial intelligence. By understanding these trends we plan to continue our research. In future study, the proposed approach can be combined with advanced techniques to retrieve and select more suitable background images for OBA. For instance, in [57], the authors suggested using the CLIP model or a diffusion model to retrieve and generate images that represent various backgrounds with changing surrounding conditions. Moreover, robust semantic segmentation models for plant monitoring can be further integrated into intelligent systems to predict and analyze plant growth [58].

Object detection annotation is more commonly available than semantic segmentation in many computer vision tasks. Therefore, the developed approach can be implemented in other specific domains of computer vision, such as remote sensing [59] or manufacturing [60], to simplify the data preparation process and improve model performance. Weak annotation improvement is another challenging task that can be addressed through the proposed approach and applied for environmental analysis [61].

6. Conclusions

In this paper, we showed how to train plant part segmentation models without given mask annotations. The findings of this study indicate that the utilization of weakly supervised segmentation methods can lead to a noteworthy enhancement in the performance of instance segmentation models, as opposed to relying solely on bounding box annotations. The suggested weak supervision framework exhibits a substantial improvement over the previously established baseline. By leveraging both the spatial information found in bounding boxes and the semantic information of pseudo-masks, the model is able to acquire a robust understanding of the underlying structures and patterns of objects, even in complex scenes. Moreover, SAM proves that with the use of only weak labels, the models can effectively tackle the segmentation task and, in terms of metrics quality, can approach the ideal scenario and even outperform it in certain cases. The provided approach demonstrates good quality for the full plants case compared to the SAM, and it can operate better with complex morphological plant structures and extract semantic information. Furthermore, we show that instance-level augmentation can utilize pseudo-masks to boost the performance of segmentation models.

In future work, one can use specialized multi-part augmentations to surpass current results.

Author Contributions: Conceptualization, S.N. and S.I.; methodology, A.S.; software, S.M.; validation, S.M., S.N. and S.I.; formal analysis, S.M.; investigation, S.M.; resources, A.S.; data curation, S.N.; writing—original draft preparation, S.M. and S.N.; writing—review and editing, S.I. and A.S.; visualization, S.M.; supervision, A.S.; project administration, S.N.; funding acquisition, S.I. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Ministry of Science and Higher Education grant No. 075-10-2021-068.

Data Availability Statement: The dataset is available upon request.

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

Abbreviations

The following abbreviations are used in this manuscript:

A2GNN	Affinity Attention Graph Neural Network
ACFN	Atrous Convolutional Feature Network
AdvCAM	Adversarial Class Activation Map
AV	Autonomous Vehicle
AP	Average Precision
AuxSegNet	Auxiliary Segmentation Network
BAP	Background Average Pooling
CCAM	Class-Agnostic Activation Map
CLIP	Contrastive Language–Image Pretraining
CMA	Collaborative Multi-Attention
CODNet	Co-attention Dictionary Network
CAM	Class Activation Map
CPN	Complementary Patch Network
DCAMA	Dense Pixel-Wise Cross-Query-and-Support Attention-Weighted Mask Aggregation
DINO	Self-DIstillatiON Loss
DA	Discriminative Activation
dCRF	Dense Conditional Random Field
EDAM	Embedded Discriminative Attention Mechanism
FP	False Positive
GPT	Generative Pretrained Transformer
GT	Ground Truth
GAP	Global Average Pooling
HDMNet	Hierarchically Decoupled Matching Network
HSSP	Hybrid Spatial Pyramid Pooling
H-DSRG	Hierarchical Deep Seeded Region Growing
ISIM	Image Segmentation with Iterative Masking
LOST	Localizing Objects with Self-Supervised Transformers
MSANet	Multi-Similarity and Attention Network
MiDaS	Monocular Depth Estimation
MIL	Multiple Instance Learning
MP	Multiple Point
mIoU	Mean Intersection Over Union
mAP	Mean Average Precision
NAL	Noise-Aware Loss
NSRM	Nonsalient Region Masking
PAC	Passive Aggressive Classifier
PPL	Progressive Patch Learning
POM	Potential Object Mining
PCM	Pixel Correlation Module
PRCM	Pixel-Region Correlation Module
OA-CAM	Online Accumulated Class Attention Map
OBA	Object-Based Augmentation
SVD	Singular Value Decomposition
SVF	Singular Value Fine-tuning
SegGPT	Segmentation Generative Pretrained Transformer
SLAM	Semantic Learning-Based Activation Map
SEAM	Self-Supervised Equivariant Attention Mechanism
SAM	Segment Anything from Meta

TransCAM	Transformer Class Activation Map
TP	True Positive
UAV	Unmanned Aerial Vehicle
VWE	Visual Word Encoder
WSSS	Weakly-Supervised Semantic Segmentation
WSIS	Weakly-Supervised Instance Segmentation
YOLO	You Only Look Once

References

1. Sorscher, B.; Geirhos, R.; Shekhar, S.; Ganguli, S.; Morcos, A. Beyond neural scaling laws: Beating power law scaling via data pruning. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 19523–19536.
2. Paton, N. Automating data preparation: Can we? should we? must we? In Proceedings of the 21st International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data, Lisbon, Portugal, 26 March 2019.
3. Lemikhova, L.; Nesteruk, S.; Somov, A. Transfer Learning for Few-Shot Plants Recognition: Antarctic Station Greenhouse Use-Case. In Proceedings of the 2022 IEEE 31st International Symposium on Industrial Electronics (ISIE), Anchorage, AK, USA, 1–3 June 2022; pp. 715–720. [CrossRef]
4. Nesteruk, S.; Shadrin, D.; Pukalchik, M.; Somov, A.; Zeidler, C.; Zabel, P.; Schubert, D. Image compression and plants classification using machine learning in controlled-environment agriculture: Antarctic station use case. *IEEE Sens. J.* **2021**, *21*, 17564–17572. [CrossRef]
5. Markov, I.; Nesteruk, S.; Kuznetsov, A.; Dimitrov, D. RusTitW: Russian Language Text Dataset for Visual Text in-the-Wild Recognition. *arXiv* **2023**, arXiv:2303.16531.
6. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 1–48. [CrossRef]
7. Illarionova, S.; Nesteruk, S.; Shadrin, D.; Ignatiev, V.; Pukalchik, M.; Oseledets, I. MixChannel: Advanced augmentation for multispectral satellite images. *Remote Sens.* **2021**, *13*, 2181. [CrossRef]
8. Illarionova, S.; Nesteruk, S.; Shadrin, D.; Ignatiev, V.; Pukalchik, M.; Oseledets, I. Object-based augmentation for building semantic segmentation: Ventura and santa rosa case study. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 1659–1668.
9. Ghiasi, G.; Cui, Y.; Srinivas, A.; Qian, R.; Lin, T.Y.; Cubuk, E.D.; Le, Q.V.; Zoph, B. Simple copy-paste is a strong data augmentation method for instance segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021; pp. 2918–2928.
10. Nesteruk, S.; Illarionova, S.; Akhtyamov, T.; Shadrin, D.; Somov, A.; Pukalchik, M.; Oseledets, I. Xtremeaugment: Getting more from your data through combination of image collection and image augmentation. *IEEE Access* **2022**, *10*, 24010–24028. [CrossRef]
11. Illarionova, S.; Shadrin, D.; Ignatiev, V.; Shayakhmetov, S.; Trekin, A.; Oseledets, I. Augmentation-Based Methodology for Enhancement of Trees Map Detalization on a Large Scale. *Remote Sens.* **2022**, *14*, 2281. [CrossRef]
12. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.
13. Xia, G.S.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A large-scale dataset for object detection in aerial images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, Utah, USA,18–22 June 2018; pp. 3974–3983.
14. Bouguettaya, A.; Zarzour, H.; Kechida, A.; Taberkit, A.M. Vehicle detection from UAV imagery with deep learning: A review. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 6047–6067. [CrossRef]
15. Feng, D.; Haase-Schütz, C.; Rosenbaum, L.; Hertlein, H.; Glaeser, C.; Timm, F.; Wiesbeck, W.; Dietmayer, K. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 1341–1360. [CrossRef]
16. Ruiz-del Solar, J.; Loncomilla, P.; Soto, N. A survey on deep learning methods for robot vision. *arXiv* **2018**, arXiv:1803.10862.
17. Illarionova, S.; Shadrin, D.; Tregubova, P.; Ignatiev, V.; Efimov, A.; Oseledets, I.; Burnaev, E. A Survey of Computer Vision Techniques for Forest Characterization and Carbon Monitoring Tasks. *Remote Sens.* **2022**, *14*, 5861. [CrossRef]
18. Zhang, B.; Xiao, J.; Jiao, J.; Wei, Y.; Zhao, Y. Affinity Attention Graph Neural Network for Weakly Supervised Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 8082–8096. [CrossRef] [PubMed]
19. Yao, Y.; Chen, T.; Xie, G.S.; Zhang, C.; Shen, F.; Wu, Q.; Tang, Z.; Zhang, J. Non-salient region object mining for weakly supervised semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021; pp. 2623–2632.
20. Wang, Y.; Zhang, J.; Kan, M.; Shan, S.; Chen, X. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 12275–12284.
21. Wu, T.; Huang, J.; Gao, G.; Wei, X.; Wei, X.; Luo, X.; Liu, C.H. Embedded discriminative attention mechanism for weakly supervised semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021. [CrossRef]
22. Bircanoglu, C.; Arica, N. ISIM: Iterative Self-Improved Model for Weakly Supervised Segmentation. *arXiv* **2022**, arXiv:2211.12455.

23. Zhang, F.; Gu, C.; Zhang, C.; Dai, Y. Complementary patch for weakly supervised semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 7242–7251.
24. Li, J.; Jie, Z.; Wang, X.; Zhou, Y.; Wei, X.; Ma, L. Weakly Supervised Semantic Segmentation via Progressive Patch Learning. *IEEE Trans. Multimed.* **2022**, *25*, 1686–1699. [CrossRef]
25. Oh, Y.; Kim, B.; Ham, B. Background-Aware Pooling and Noise-Aware Loss for Weakly-Supervised Semantic Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2021, Virtual, 19–25 June 2021. [CrossRef]
26. Ma, T.; Wang, Q.; Zhang, H.; Zuo, W. Delving Deeper Into Pixel Prior for Box-Supervised Semantic Segmentation. *IEEE Trans. Image Process.* **2022**, *31*, 1406–1417. [CrossRef]
27. Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A.C.; Lo, W.Y.; et al. Segment anything. *arXiv* **2023**, arXiv:2304.02643.
28. Xu, L.; Xue, H.; Bennamoun, M.; Boussaid, F.; Sohel, F. Atrous convolutional feature network for weakly supervised semantic segmentation. *Neurocomputing* **2021**, *421*, 115–126. [CrossRef]
29. Chen, J.; Zhao, X.; Liu, M.; Shen, L. SLAM: Semantic Learning based Activation Map for Weakly Supervised Semantic Segmentation. *arXiv* **2022**, arXiv:2210.12417.
30. Xu, L.; Ouyang, W.; Bennamoun, M.; Boussaid, F.; Sohel, F.; Xu, D. Leveraging Auxiliary Tasks with Affinity Learning for Weakly Supervised Semantic Segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, QC, Canada, 10–17 October 2021. [CrossRef]
31. Wan, W.; Chen, J.; Yang, M.H.; Ma, H. Co-attention dictionary network for weakly-supervised semantic segmentation. *Neurocomputing* **2022**, *486*, 272–285. [CrossRef]
32. Chong, Y.; Chen, X.; Tao, Y.; Pan, S. Erase then grow: Generating correct class activation maps for weakly-supervised semantic segmentation. *Neurocomputing* **2021**, *453*, 97–108. [CrossRef]
33. Ru, L.; Du, B.; Wu, C. Learning Visual Words for Weakly-Supervised Semantic Segmentation. In Proceedings of the 13th International Joint Conference on Artificial Intelligence, Virtual, 19–27 August 2021. [CrossRef]
34. Lee, J.; Kim, E.; Yoon, S. Anti-adversarially manipulated attributions for weakly and semi-supervised semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021; pp. 4071–4080.
35. Siméoni, O.; Puy, G.; Vo, H.V.; Roburin, S.; Gidaris, S.; Bursuc, A.; Pérez, P.; Marlet, R.; Ponce, J. Localizing objects with self-supervised transformers and no labels. *arXiv* **2021**, arXiv:2109.14279.
36. Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; Joulin, A. Emerging properties in self-supervised vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 9650–9660.
37. Wang, Y.; Shen, X.; Hu, S.X.; Yuan, Y.; Crowley, J.L.; Vaufreydaz, D. Self-supervised transformers for unsupervised object discovery using normalized cut. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18-24 June 2022; pp. 14543–14553.
38. Melas-Kyriazi, L.; Rupprecht, C.; Laina, I.; Vedaldi, A. Deep Spectral Methods: A Surprisingly Strong Baseline for Unsupervised Semantic Segmentation and Localization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18-24 June 2022; pp. 8364–8375.
39. Sauvalle, B.; de La Fortelle, A. Unsupervised Multi-object Segmentation Using Attention and Soft-argmax. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 2–7 January 2023; pp. 3267–3276.
40. Xie, J.; Xiang, J.; Chen, J.; Hou, X.; Zhao, X.; Shen, L. C2AM: Contrastive learning of Class-agnostic Activation Map for Weakly Supervised Object Localization and Semantic Segmentation. *arXiv* **2022**, arXiv:2203.13505.
41. Sun, Y.; Chen, Q.; He, X.; Wang, J.; Feng, H.; Han, J.; Ding, E.; Cheng, J.; Li, Z.; Wang, J. Singular Value Fine-tuning: Few-shot Segmentation requires Few-parameters Fine-tuning. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 37484–37496.
42. Iqbal, E.; Safarov, S.; Bang, S. MSANet: Multi-Similarity and Attention Guidance for Boosting Few-Shot Segmentation. *arXiv* **2022**, arXiv:2206.09667.
43. Shi, X.; Wei, D.; Zhang, Y.; Lu, D.; Ning, M.; Chen, J.; Ma, K.; Zheng, Y. Dense cross-query-and-support attention weighted mask aggregation for few-shot segmentation. In *Computer Vision–ECCV 2022, Proceedings of the 17th European Conference, Tel Aviv, Israel, 23–27 October 2022*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 151–168.
44. Peng, B.; Tian, Z.; Wu, X.; Wang, C.; Liu, S.; Su, J.; Jia, J. Hierarchical Dense Correlation Distillation for Few-Shot Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 23641–23651.
45. Wang, X.; Zhang, X.; Cao, Y.; Wang, W.; Shen, C.; Huang, T. SegGPT: Segmenting everything in context. *arXiv* **2023**, arXiv:2304.03284.
46. Li, R.; Mai, Z.; Zhang, Z.; Jang, J.; Sanner, S. TransCAM: Transformer Attention-based CAM Refinement for Weakly Supervised Semantic Segmentation. *J. Vis. Commun. Image Represent.* **2023**, *92*, 103800. [CrossRef]
47. Ranftl, R.; Lasinger, K.; Hafner, D.; Schindler, K.; Koltun, V. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 1623–1637. [CrossRef]
48. Rezaei, M.; Farahanipad, F.; Dillhoff, A.; Elmasri, R.; Athitsos, V. Weakly-supervised hand part segmentation from depth images. In Proceedings of the 14th PErvasive Technologies Related to Assistive Environments Conference, Virtual, 29 June–2 July 2021; pp. 218–225.

49. Ergül, M.; Alatan, A. Depth is all you Need: Single-Stage Weakly Supervised Semantic Segmentation From Image-Level Supervision. In Proceedings of the 2022 IEEE International Conference on Image Processing (ICIP), Bordeaux, France, 16–19 October 2022; pp. 4233–4237.
50. Khoreva, A.; Benenson, R.; Hosang, J.; Hein, M.; Schiele, B. Simple does it: Weakly supervised instance and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 876–885.
51. Ostertagova, E.; Ostertag, O.; Kováč, J. Methodology and application of the Kruskal-Wallis test. *Appl. Mech. Mater.* **2014**, *611*, 115–120. [CrossRef]
52. Zhang, Q.; Yang, M.; Zheng, Q.; Zhang, X. Segmentation of hand gesture based on dark channel prior in projector-camera system. In Proceedings of the 2017 IEEE/CIC International Conference on Communications in China (ICCC), Qingdao, China, 22–24 October 2017; pp. 1–6.
53. Zheng, Q.; Yang, M.; Tian, X.; Wang, X.; Wang, D. Rethinking the Role of Activation Functions in Deep Convolutional Neural Networks for Image Classification. *Eng. Lett.* **2020**, *28*.
54. Illarionova, S.; Shadrin, D.; Ignatiev, V.; Shayakhmetov, S.; Trekin, A.; Oseledets, I. Estimation of the Canopy Height Model From Multispectral Satellite Imagery with Convolutional Neural Networks. *IEEE Access* **2022**, *10*, 34116–34132. [CrossRef]
55. Zheng, Q.; Zhao, P.; Li, Y.; Wang, H.; Yang, Y. Spectrum interference-based two-level data augmentation method in deep learning for automatic modulation classification. *Neural Comput. Appl.* **2021**, *33*, 7723–7745. [CrossRef]
56. Zheng, Q.; Zhao, P.; Wang, H.; Elhanashi, A.; Saponara, S. Fine-grained modulation classification using multi-scale radio transformer with dual-channel representation. *IEEE Commun. Lett.* **2022**, *26*, 1298–1302. [CrossRef]
57. Nesteruk, S.; Zherebtsov, I.; Illarionova, S.; Shadrin, D.; Somov, A.; Bezzateev, S.V.; Yelina, T.; Denisenko, V.; Oseledets, I. CISA: Context Substitution for Image Semantics Augmentation. *Mathematics* **2023**, *11*, 1818. [CrossRef]
58. Nesteruk, S.; Shadrin, D.; Kovalenko, V.; Rodríguez-Sanchez, A.; Somov, A. Plant growth prediction through intelligent embedded sensing. In Proceedings of the 2020 IEEE 29th International Symposium on Industrial Electronics (ISIE), Delft, The Netherlands, 17–19 June 2020; pp. 411–416.
59. Illarionova, S.; Shadrin, D.; Shukhratov, I.; Evteeva, K.; Popandopulo, G.; Sotiriadi, N.; Oseledets, I.; Burnaev, E. Benchmark for Building Segmentation on Up-Scaled Sentinel-2 Imagery. *Remote Sens.* **2023**, *15*, 2347. [CrossRef]
60. Fu, Y.; Yao, X. A review on manufacturing defects and their detection of fiber reinforced resin matrix composites. *Compos. Part C Open Access* **2022**, *8*, 100276. [CrossRef]
61. Illarionova, S.; Trekin, A.; Ignatiev, V.; Oseledets, I. Tree species mapping on sentinel-2 satellite imagery with weakly supervised classification and object-wise sampling. *Forests* **2021**, *12*, 1413. [CrossRef]