

Leveraging Prompt-Based Segmentation Models and Large Dataset to Improve Detection of Trees

Vincent Grondin^{1 3}, Philippe Massicotte², Mohamed Gaha², François Pomerleau¹ and Philippe Giguère¹

Abstract—The abundance of unlabeled forest images on the web is a powerful yet untapped resource to train forestry vision models. Two key challenges limiting the use of these unlabeled images are *i*) collecting the images and *ii*) obtaining the labels, as supervised learning remains the prevailing approach for model training. In this work, we address the first issue by providing a dataset of 110 k forest images sourced from a repository of pictures taken by amateur photographers worldwide. To generate supplementary labels for supervised training, we propose a two-step approach. First, we train a network on a small labelled dataset, to generate pseudo-labels on the much larger, unlabelled one. Then, we leverage the zero-shot segmentation capability of the Segment Anything Model to improve the quality of these pseudo-labels. Our experiments demonstrate that both the proposed dataset and the pseudo-labeling method increase performance of a tree detector at no additional labeling cost. This performance increase is particularly significant in challenging scenarios, showing that training the model with better segmentation masks notably helps disentangle overlapping trees and detect odd-shaped ones, gaining between 3.3 AP^{bb}, 7.7 AP^{seg} or 1.6 AP^{bb}, 3.5 AP^{seg} percentage points depending on the burn-in model. Code and dataset links are available at <https://github.com/norlab-ulaval/PercepTreeV1>.

I. INTRODUCTION

Computer vision is a promising way to automate various tasks traditionally performed by humans. Within forestry, the necessity for accurate tree detection prevails across numerous applications such as species classification, disease detection, field surveying, and navigation [1]. To address these needs, many recent research studies are using self-supervised or semi-supervised learning to leverage data *at scale*. Semi-supervised learning converts large image collections into a training dataset that can surpass the diversity achieved by a small, labeled counterpart. In this sense, the foundational model Segment Anything Model (SAM) [2] has aroused the interest of the computer vision community by its impressive zero-shot generalization capabilities on new data distributions and the introduction of a new training task: promptable segmentation. However, obtaining high-quality pseudo labels, even with SAM, is still a complex problem. The authors notably report a decrease in Average Precision (AP) when prompting SAM with a box detector specifically designed and finetuned for object detection on COCO [3]. **This approach is problematic because SAM must output better segmentation masks than a finetuned detector and achieve this without additional training data or finetuning, i.e., zero-shot.**

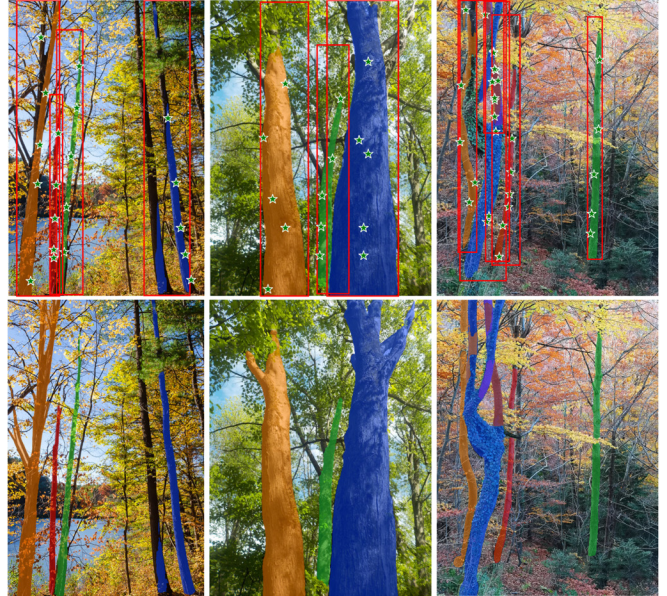


Fig. 1: High-quality pseudo-labels using the Segment Anything Model. **First row**, pseudo-labels (bounding box and segmentation mask predictions) from the burn-in model are depicted. **Second row** showcases pseudo-labels produced by SAM utilizing prompt inputs derived from the burn-in model. Stars correspond to the 5 point prompts from our prompting strategy.

Although SAM generally exhibits inferior performance compared to finetuned models, exploring its utility in pseudo-label generation remains a compelling research avenue for four key reasons. Firstly, the original experiments benchmarked SAM against a model finetuned on a dataset comprising over 100k labeled images. It is plausible that a model finetuned on a smaller dataset, as in our test case, might perform worse. Secondly, human evaluators consistently rated SAM’s segmentation masks higher than hand-labelled ground truth, demonstrating its potential to produce superior quality pseudo-labels. Thirdly, SA-1B [2], the largest image segmentation dataset with one B masks and 11 M images, is entirely generated by SAM itself, highlighting its efficiency in generating high-quality pseudo-labels for a substantial volume of unlabeled images. Fourthly, the intuition behind the promptable segmentation task of SAM was to enable solving a range of downstream segmentation problems on new data distributions through prompt engineering. In that sense, the authors encourage *composing* SAM in a broader system (e.g., where SAM acts as a combined component with an existing object detector to prompt it for segmentation [2]). Consequently, we developed a practical methodology to **streamline the generation of high-quality pseudo-labels**

*This work is supported by the FORAC Consortium.

¹ Department of Computer Science and Software Engineering, Université Laval, Canada.

² Institut de recherche électrique du Québec, Hydro-Québec, Canada.

with SAM on small labelled datasets, and applied it on a forestry problem.

However, **the crux of the challenge lies in providing SAM with input prompts that effectively represent the object of interest** without being too ambiguous or overly restrictive. Should the segmentation predicted by SAM differs significantly from the ground truth, it risks negatively impacting the segmentation performance. In order for SAM to generate better pseudo-labels than the ones from a finetuned model, the prompts must strike a delicate balance — precise enough to delineate the object of interest and simultaneously generic enough to refine the initial object segmentation by expanding or contracting the segmented pixels. For instance, point prompts excel in expanding initial detections but may fall short when the object is partially occluded or composed of distinct parts. For example, if a point prompt lies on a part or subpart, SAM might return the subpart, part, or whole object. Thus, it is preferable to use more than one point or combine it with other prompt inputs. Box prompts effectively delimit the object’s spatial extent but introduce ambiguity when the box encompasses multiple objects. Mask prompts yield a more comprehensive prompt, yet their standalone use is worse than box prompts [4].

To the best of our knowledge, no previous research has **built upon SAM to turn a large dataset of unlabeled images into a large pseudo-labeled dataset**. While numerous papers have investigated different applications for SAM, very few report performance improvements compared to finetuned models [4]–[6]. On medical images, studies reveal a notable disparity between SAM predictions and ground truth [6], [7]. When testing three modes of prompts, the finetuned model outperformed SAM detections for all prompt modes, indicating that the domain gap might be too substantial for SAM on medical images without undergoing a finetuning process [7]. Since no attempts to use this kind of approach on forestry images has been explored before, one of our goal is to assess how well SAM can perform tree trunk segmentation.

In this paper, we investigate how SAM can be used to produce high-quality segmentation masks on tree trunks, when the labelled dataset is small (100 images). Our method extends the groundwork established by Kirillov *et al.* [2] by prompting SAM with boxes, masks, and points derived from the predictions of an object detection model. Notably, the point prompts are strategically sampled to focus on locations with the highest confidence, thus having a greater likelihood of corresponding to a tree trunk. The proposed approach can generate high-quality pseudo-labels (see Figure 1) that are effective for model training, resulting in a noteworthy enhancement of segmentation performances by 7.7 AP^{seg} percentage points on our most varied test set. In specific instances where entangled trees were pseudo-labeled as a single one, our method can effectively disentangle them, showcasing improved detection, particularly in scenarios involving grouped trees. Furthermore, we found that pseudo-labels generated by SAM for trees with unconventional shapes were segmented qualitatively better.

In short, the main contributions of our paper are:

- 1) A highly diverse unlabeled dataset comprising 110,000 forest images with pseudo-labels and a subset of 100 human-labeled images;
- 2) A comparative analysis on prompting methods and the impact of dataset size on performances;
- 3) A strategy based-on SAM to successfully generate high-quality pseudo-labels.

II. RELATED WORKS

A. Semi-Supervised learning

The conventional framework for semi-supervised learning methods is a small set of labeled data and a more extensive set of unlabeled data. Most existing works do this in two stages [8]–[11]. The first stage is the burn-in, where an object detector is trained on the small labeled dataset. The mutual learning stage unfolds in the second stage, where the burn-in object detector is duplicated into two models: a student and a teacher. The student is trained on strong image augmentations and uses the predicted output of the teacher from the same but weakly augmented image as the ground truth. The teachers’ weights are generally updated using an exponential moving average, which helps stabilize training [12].

Although successful, the mutual learning stage of these approaches makes it difficult to leverage SAM’s zero-shot segmentation, as it would require three models running at once on the GPU. This is also why these approaches generally involve models with smaller backbones, such as ResNet-50 [13] with 23 M trainable parameters compared to ViT-H [14] with 632 M for SAM.

B. Pseudo-labeling with SAM

Foundational models such as SAM can be described as models "trained on broad data at scale and are adaptable to a wide range of downstream tasks" [15]. These models are thus particularly interesting for application in domains like forestry, where training data has been relatively scarce so far [16]–[19]. Because of its novel promptable segmentation task objective to return a valid segmentation mask for any given prompt, SAM can predict a plausible mask even when the prompt could refer to multiple objects (i.e., is ambiguous). Its zero-shot segmentation capabilities have made it a handy tool to label image datasets, including the dataset on which it underwent training, SA-1B. Although SAM was not entirely trained on mask labels generated by itself, the labels of SA-1B publicly released *only include automatically generated masks*, without human annotator input. In fact, the data engine operated in three stages: 1) model-assisted manual labeling, 2) semi-automatic label generation with model-assisted labels and automatically predicted masks, and 3) a fully automatic stage with no human intervention. In other words, the majority ¹ (99.1 %) of the 1.1 B segmentation masks from 11 M images are automatically generated by SAM, then retrained on them. This demonstrates that

¹The publicly available SA-1B ground truth annotations only includes automatically generated masks label, meaning the human-made labels are kept private.

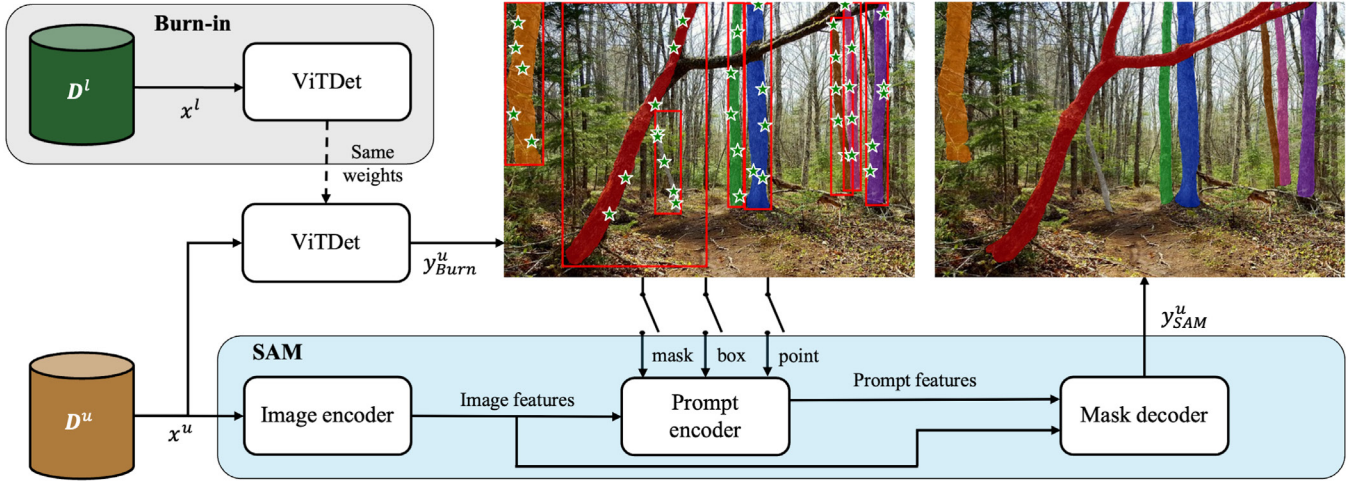


Fig. 2: Workflow of our proposed pseudo-label method using SAM. First, the burn-in stage involves training an object detector on a small labeled dataset \mathbf{D}_l of tree trunks (either CANATREE100 or FLICKRTREE100). In the second stage, images from the unlabeled dataset \mathbf{D}_u are inputted simultaneously to the burn-in model and SAM. The detections from the burn-in model serve as inputs to the prompt encoder before generating a potentially improved segmentation mask.

even though generated automatically, the mask labels are high quality and effective for model training. Similar to this retraining scheme, Cut and Learn [20] proposed a multi-round self-supervised training, where the model successfully learns from its own predictions.

Other works have shown interest in applying SAM’s zero-shot performance with the hope of surpassing the current image segmentation techniques in their respective domains. Cheng *et al.* [7] comprehensively studied three prompt modes for medical image segmentation on 12 different datasets. Among the auto-prompt, point, and box prompt modes, the authors identify the box prompt with zero jitters as the most effective, yielding results competitive with the state-of-the-art. However, box jitter can significantly degrade mask accuracy [4], [7]. For the work of Ke *et al.* [4], SAM often fails to segment intricate structures such as thin ropes, mesh, or poles. Using HQSeg-44K, a dataset regrouping six existing datasets totaling 44,320 meticulously annotated image masks, they propose to replace SAM’s original output token with a lightweight, high-quality output token trained on HQSeg-44K. The authors named the model HQ-SAM, and reported improvements on all the datasets they tested. This was obtained without compromising zero-shot segmentation, since they only finetuned output tokens: no other parts of the initial model weights were modified.

In our case, the labeled datasets dedicated to tree detection are significantly smaller in scale compared to HQSeg-44K, i.e. 100 instead of 44,320 annotated images. This limits our ability to train the lightweight, high-quality output token module of HQ-SAM. Instead, we drew inspiration from the methodology employed in the (mostly) automated labeling of SA-1B: a burn-in stage to train a detector to subsequently pseudo-label a large dataset. Since we do not have the resources to have a human-in-the-loop correcting pseudo-labels for our 110k images as in the work of Kirillov *et al.* [2], we restricted ourselves to only one iteration of pseudo-labels, without human-in-the-loop.

III. METHODOLOGY

Our semi-supervised object detection approach is depicted in Figure 2. It uses either of two small labeled datasets (CANATREE100, FLICKRTREE100) of forest images with box and segmentation mask annotations for trunks, as well as a large unlabeled dataset of forests (FLICKRTREE110K) described in Section III-B. Next, a model is trained on one of the labeled datasets, which we refer to as the burn-in stage in Section III-C. Starting from the predictions of the burn-in model, we aim to prompt SAM with said predictions to produce a better segmentation mask and bounding box coordinates on new data distribution. Using various prompting configurations described in Section III-E, pseudo-labels are generated for FLICKRTREE110K. The last step is to finetune a detection model on FLICKRTREE110K’s pseudo-labels.

A. Preliminary

Our objective is to obtain high-quality pseudo-labels \mathbf{y}^u for a large dataset of unlabeled images $\mathbf{D}_u = \{\mathbf{x}_i^u\}_{i=1}^{N^u}$. A small labeled dataset $\mathbf{D}_l = \{\mathbf{x}_i^l, \mathbf{y}_i^l\}_{i=1}^{N^l}$ is used in the burn-in stage (Section III-C). For each labeled image \mathbf{x}^l , the annotations \mathbf{y}^l contain categories, bounding boxes, and segmentation masks of all trees of interest in the images. N^l and N^u refer to the number of labeled and unlabeled images, respectively.

B. Datasets

The extensive variation in geographical locations, species composition, weather conditions, seasons, sensors, and scales introduces significant distribution shifts in forest datasets. Some of these variations can be captured in synthetic forest datasets [21], [22], but the scarcity of large, ground-based labeled datasets documented in the literature still poses a significant challenge. Our FLICKRTREE110K unlabelled dataset addresses this diversity issue, as can be seen in Figure 3 for a number of randomly-selected samples. On top of this, we supplemented our existing labelled dataset CANATREE100 [18] by curating and labeling an additional 100



Fig. 3: Randomly sampled images from our unlabeled dataset FLICKRTREE110K.

images extracted from FLICKRTREE110K, and nicknamed it FLICKRTREE100 (see Table I). Doing so allowed us to test the generalization capabilities of our approach, for instance when the burn-in is performed strictly on CANATREE100 or strictly on FLICKRTREE100.

TABLE I: Characteristics of the datasets used in our experiments.

Dataset	# Images	Size	# Labels	Location
CANATREE100	100	1280×720	920	Québec
FLICKRTREE100	100	$\leq 800 \times 800$	1463	World
FLICKRTREE110K	110 937	$\leq 800 \times 800$	N/A	World

In details, CANATREE100 [18] comprises 100 images collected from public, private, and commercial forests in Quebec, Canada. It contains 920 labeled trees, and the images were sampled from videos at a lower framerate than other annotated tree datasets such as [16], [17], [19]. In the case of FLICKRTREE100, it contains 100 images sourced from Flickr². The Flickr dataset exhibits various images collected by amateur photographers, encompassing different camera models, resolutions, geographical regions, and other parameters. Unlike CANATREE100, the images in FLICKRTREE100 are not sourced from videos, resulting in a much broader diversity and absence of motion blur. It showcases complex vegetation structures, intertwined and overlapping trunks or crowns, and a wide diversity of species and tree sizes (see Figure 3).

Our large dataset of images \mathbf{D}_u is also sourced from Flickr. Using their API, over 110k images with the keywords "forest" and "tree" were downloaded. Note that as these images were not subjected to human verification post-download, some may not necessarily feature trees, rendering

them irrelevant for tree detection purposes. The licenses for images on Flickr can vary, as users can choose from a range of licenses when uploading their content. Consequently, the dataset is not self-contained, and only links to the image URLs will be provided to the research community³.

C. Burn-in stage

The burn-in stage involves training the object detector model ViTDet [23] on the small, human-labeled dataset \mathbf{D}_l . The object detector is initialized from a COCO checkpoint and trained with the three standard supervised losses for the class, box, and mask predictions :

$$L_{sup} = L_{cls} + \lambda_{box} L_{box} + \lambda_{mask} L_{mask}. \quad (1)$$

D. Pseudo-label generation from burn-in model

After burn-in, the trained detector presented above is used to generate pseudo-labels \mathbf{y}_{Burn}^u for our FLICKRTREE110K unlabelled dataset \mathbf{D}_u . For each unlabeled image $x^u \in \mathbf{D}_u$, the image is processed through the burned-in model, yielding a pseudo-label y_{burn}^u with an associated confidence score δ . Based on previous research findings [8] and our own experiments, a confidence threshold $\delta^{th} = 0.5$ is set in order to reduce the negative impact of noisy pseudo-labels. In general, low confidence predictions $\delta < 0.5$ are more likely to be associated with false positive. Subsequent non-maximum suppression is also performed to remove overlapping pseudo-labels.

E. Pseudo-label generation from SAM

One key aspect of our work is demonstrating how we can improve the quality of these pseudo-labels y_{burn}^u . To

²<https://www.flickr.com/>

³Note that there is no guarantee that the URL links will exist or remain in the future.

this effect, we used SAM to segment trees, hoping to attain a greater accuracy across geographically diverse forests compared to what can be achieved by a burn-in model trained on a small amount of domain specific data (i.e., Canadian forests in the case of CANATREE100). We refer to a pseudo-label generated by SAM as y_{SAM}^u . Since we do it on the entire dataset FLICKRTREE110K, the resulting pseudo-labels are denoted y_{sam}^u .

Different prompt configurations can be used to input SAM with a prompt representative of the target object to segment. We explored three strategies:

- 1) Bounding box prompt from a detector;
- 2) Bounding box, segmentation mask from a detector;
- 3) Bounding box, segmentation mask from a detector, and five point prompts.

The first strategy is the same as the one used in [2], [7], where a tree detector is used to feed the predicted bounding box as prompts. The second strategy adds the segmentation mask logits predicted by the tree detector as prompts. The third strategy combines the first two, by adding point prompts sampled from the predicted mask logits. To sample point on regions that are likely trees, we use the max logits of the mask. We posit that they correspond to areas with high likelihood of being a tree according to the burn-in model. The sampling strategy can be observed in Figure 4, the vertical split was beneficial as it forces SAM to also segment small regions at the tree’s end. In our experiments, using a bounding box, mask, and five points per detected tree is the best strategy to generate high-quality pseudo-labels with SAM, and this strategy will be used in Section IV-B.

Ablation results for these three strategies are presented in Section IV-C.

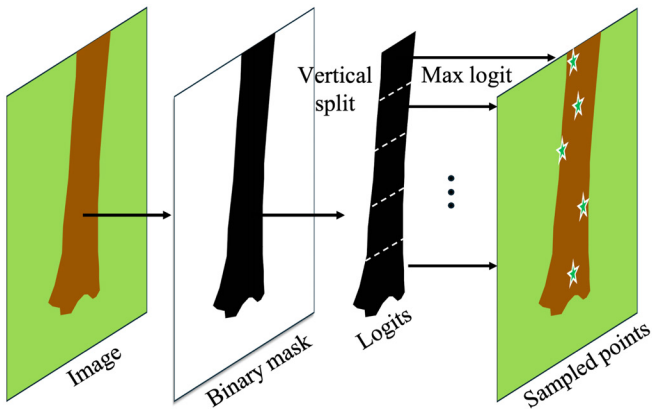


Fig. 4: Our proposed point prompt sampling strategy. The binary mask of the image is split into five equal vertical areas to encourage SAM to segment the tree trunk in its entirety, not just at a local area. A point is sampled from each area based on the maximal logit value in this area.

F. Finetuning on pseudo-labels

Once pseudo-labels are generated, a ViTDet architecture initialized on COCO is trained on \mathbf{D}_u with either the set of pseudo-labels y_{burn}^u or y_{sam}^u . The pseudo-label can be directly from the burn-in model or generated by SAM.

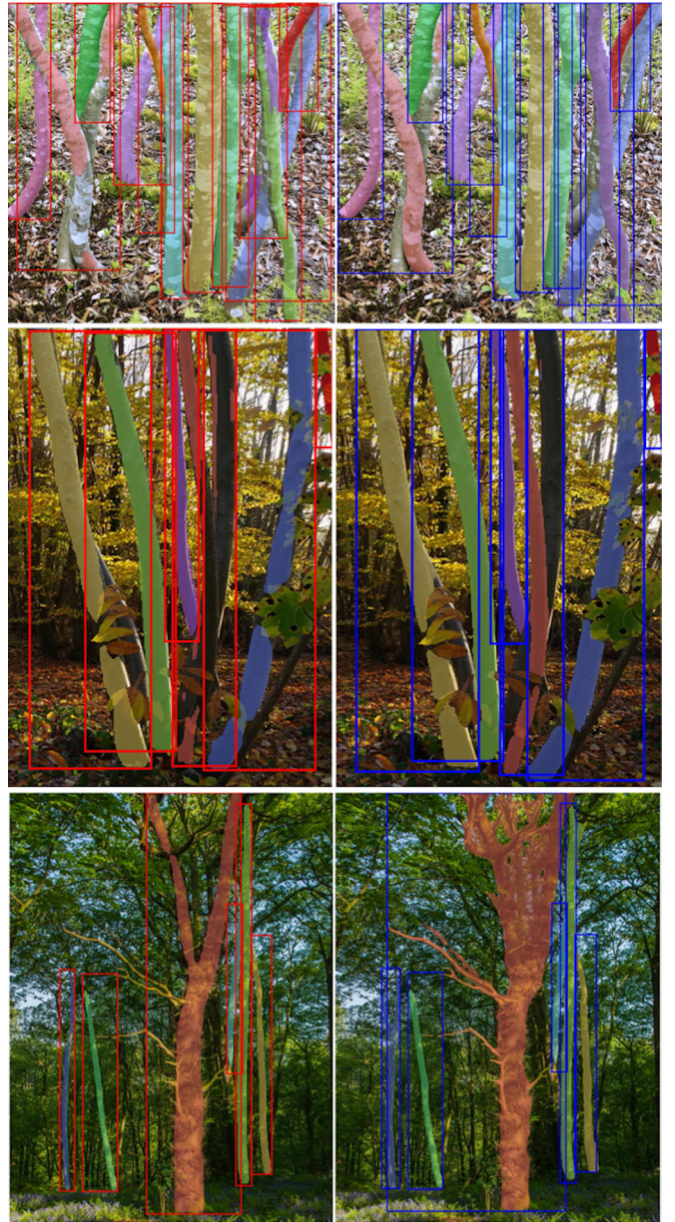


Fig. 5: Examples of mask and box pseudo-labels generated by SAM using our proposed prompting method with box, training solely on labeled and points. **Top, middle:** SAM exhibits the capability to disentangle some of the wrongly segmented trees by re-mapping the mask used as a prompt to the correct tree. This can turn false positives into true positives. Bounding box coordinates are also readjusted to fit the entire trunk detected by SAM if it was not the case. **Bottom:** Error case where SAM wrongly associates background branches to the tree. It also correctly associates some of the lower branches, causing the bounding box to cover a much larger area, which can hurt the performances on the test set if the labels do not include branches.

IV. EXPERIMENTS

A. Implementation details

The ViTDet implementation and the initial COCO checkpoint used are from the Detectron2 [24] library. We chose a ViT-H backbone architecture since it is the largest and highest scoring for object detection available from the checkpoint list. It is also the same architecture used by SAM. The burn-in model is trained on the *train set* of 60 images from CANATREE100, referred to as CT-Train, and the *validation*

TABLE II: Experimental results on our two test sets. It shows performances without pseudo-labels (\mathbf{y}^l), with pseudo-labels (\mathbf{y}_{burn}^u), and with pseudo-labels refined by SAM (\mathbf{y}_{sam}^u). Abbreviations for CanaTree (CT) and FlickrTree (FT) are used. The prompt strategy to generate SAM’s pseudo-labels was a bounding box and a segmentation mask predicted by the burn-in detector, along with five point prompts

Test set	Train set	AP _{50:95} ^{bb}	AP ₅₀ ^{bb}	AP _{50:95} ^{seg}	AP ₅₀ ^{seg}	AR _{50:95} ^{bb}	AR ₅₀ ^{bb}	AR _{50:95} ^{seg}	AR ₅₀ ^{seg}
CT-Test	\mathbf{y}_{ct}^l	69.6	91.4	64.2	89.3	78.1	96.9	70.5	95.0
	$\mathbf{y}_{ct burn}^u$	70.4	92.8	64.3	92.0	78.4	97.8	69.4	96.6
	$\mathbf{y}_{ct sam}^u$	69.6	92.4	63.8	91.4	78.3	97.5	69.5	96.4
	\mathbf{y}_{ft}^l	65.4	89.4	57.9	88.2	76.5	97.5	65.6	95.5
	$\mathbf{y}_{ft burn}^u$	66.0	90.1	52.4	87.9	76.8	98.3	60.7	95.5
	$\mathbf{y}_{ft sam}^u$	66.0	89.9	57.0	87.8	76.8	98.0	64.5	95.3
FT-Test	\mathbf{y}_{ct}^l	46.5	75.0	34.5	65.0	55.6	81.3	43.8	73.8
	$\mathbf{y}_{ct burn}^u$	54.6	80.9	45.4	76.2	63.6	87.8	53.8	83.3
	$\mathbf{y}_{ct sam}^u$	57.9	81.8	53.1	80.7	66.8	88.4	61.6	87.1
	\mathbf{y}_{ft}^l	61.8	87.3	56.5	85.8	69.8	92.9	65.3	92.7
	$\mathbf{y}_{ft burn}^u$	63.0	89.9	57.1	87.5	71.4	95.8	64.2	93.7
	$\mathbf{y}_{ft sam}^u$	64.6	90.1	60.6	88.7	72.3	95.8	66.9	94.2

TABLE III: Comparison of prompt methods used to generate SAM’s pseudo-labels, evaluated on FT-Test.

Prompt	AP ₅₀ ^{bb}	AP _{50:95} ^{bb}	AP ₅₀ ^{seg}	AP _{50:95} ^{seg}
Box	54.2	79.7	43.1	77.8
Box, mask	54.7	81.3	45.9	78.2
Box, mask, 5 points	57.9	81.8	53.1	80.7

set is used to monitor the model training. The burn-in model is trained for 4500 iterations, with a batch size of 4. The learning rate is 1e-4, multiplied by a 0.1 factor at steps [800, 2600]. We used the same data augmentation as in the original implementation of ViTDet [23].

To generate SAM’s pseudo-labels, the HQ-SAM variant of SAM was used as it Ke *et al.* [4] empirically demonstrated better segmentation mask and robustness to noisy box prompt inputs. For the sake of brevity, we refer to HQ-SAM as SAM in our results. The model’s parameters used for SAM’s pseudo-labels are a threshold of 86 for the intersection over union (IoU) prediction score, single mask output, and no post-mask refinement.

To measure the impact of training on pseudo-labels, we re-initialize ViTDet from the COCO checkpoint. It is trained for 50k iterations, with a batch size of 4, a learning rate of 1e-4 multiplied by a 0.1 factor at steps [2000, 10000].

B. Results

We measure the quality of a pseudo-labeling method by evaluating performance of an object detector model (ViTDet) trained on either \mathbf{y}_{Burn}^u or \mathbf{y}_{SAM}^u pseudo-labels. Table II shows the detection results when *i*) only training on a labeled dataset, *ii*) training on pseudo-labels \mathbf{y}_{Burn}^u generated by the burn-in model, and *iii*) training on pseudo-labels \mathbf{y}_{SAM}^u refined by SAM. For this experiment, the prompt strategy to generate \mathbf{y}_{SAM}^u was a bounding box and a segmentation mask predicted by the burn-in detector, along with five point prompts sampled from the mask as described in Section III-E and Figure 4.

From Table II, we observe that training solely on labeled data yields competitive results on CT-Test, albeit slightly underperforming the other training methods by less than 0.9% AP/AR. We did not see a significant performance

TABLE IV: Effect on performance for reduced dataset size, evaluated on FT-Test. It shows performances with pseudo-labels (\mathbf{y}_{burn}^u), and with pseudo-labels refined by SAM (\mathbf{y}_{sam}^u)

Size	Train set	AP _{50:95} ^{bb}	AP ₅₀ ^{bb}	AP _{50:95} ^{seg}	AP ₅₀ ^{seg}
100 %	$\mathbf{y}_{ct burn}^u$	54.6	80.9	45.4	76.2
	$\mathbf{y}_{ct sam}^u$	57.9	81.8	53.1	80.7
10 %	$\mathbf{y}_{ct burn}^u$	54.2	80.6	45.5	76.7
	$\mathbf{y}_{ct sam}^u$	57.5	81.5	51.8	79.5
5 %	$\mathbf{y}_{ct burn}^u$	54.6	80.7	45.4	76.3
	$\mathbf{y}_{ct sam}^u$	57.5	81.1	51.8	78.5
1 %	$\mathbf{y}_{ct burn}^u$	53.3	77.7	44.3	74.5
	$\mathbf{y}_{ct sam}^u$	55.4	77.7	50.4	76.8

improvement on CT-Test, possibly because CANATREE100 mainly contains straight tree trunks with minimal branches (i.e., easy FLICKRTREE110K samples).

However, when evaluated on a test set containing an array of diverse and complex images such as Flickr-Test, we observe a significant gain in detection and segmentation performance for models trained on pseudo-labels generated with or without SAM. Quantitatively, we observe that models burned-on CANATREE100 and trained on the pseudo-labels $\mathbf{y}_{ct sam}^u$ generated by SAM see an improvement of 3.3 AP_{50:95}^{bb} and 7.7 AP_{50:95}^{seg} when tested on FT-Test. For the models burned-on FLICKRTREE100, the improvement is 1.6 AP_{50:95}^{bb} and 3.5 AP_{50:95}^{seg} when tested on FT-Test.

An explanation for this increase in performance on FLICKRTREE100 and not on CANATREE100 is that the labels generated by SAM for straight, normal-looking trees are often very similar to those without SAM. As can be observed in Figure 5, the main difference arises when trees are close to each other in a bunch, but also when they are inclined like in Figure 1. In these cases, it appears that the model learned from SAM’s pseudo-labels \mathbf{y}_{SAM}^u new correlation and patterns that generalize better than \mathbf{y}_{Burn}^u generated by the burn-in model.

These results indicate that training on pseudo-labels \mathbf{y}_{SAM}^u generated by SAM can improve detection results, highlighting the benefits of foundational models to generalize on diverse images, including forests. Another takeaway is the importance of the burn-in dataset, where burning a model

on FLICKRTREE100 generalizes significantly better. In fact, a model burned on CT-Train drops from 69.6 AP to 46.5 AP when tested on CT-Test and FT-Test, respectively. That is a 23.1 Δ AP percentage points drop compared to a 3.8 Δ AP percentage points drop (from 65.4 to 61.8 AP) for the model burned on FT-Train.

C. Prompting method

To demonstrate the effectiveness of our prompting method, we compared it with two other variants. The first is prompting SAM with the detected box only, and the second is with the box and detected mask.

Table III shows the importance of using point prompts in addition to the box and mask prompt, gaining 3.2 $AP_{50:95}^{bb}$ and 7.2 $AP_{50:95}^{seg}$ percentage points over the second best method, box and mask. We conjecture that point prompts guide SAM into segmenting the specific object of interest inside the bounding box without being as restrictive as the mask prompt. As Ke *et al.* [4] noted before us, more prompt information means less object ambiguity, and leads to performance increase.

D. Impact of pseudo-labeled dataset size

To evaluate sample efficiency, FLICKRTREE110K is split into subsets of varying proportions [10, 5, 1]%, with their corresponding pseudo-labels y_{Burn}^u or y_{SAM}^u . As before, a ViTDet detector is initialized from a COCO checkpoint but is trained on each subset's pseudo-labels instead of the entire pseudo-labeled dataset.

From the results in Table IV, we see that even when reducing to 1% of FLICKRTREE110K dataset size, the models trained on SAM's pseudo-labels consistently outperform the ones without. Interestingly, the impact of reduced dataset size is more pronounced for segmentation mask predictions. In fact, the model performances **scale significantly better with dataset size when the pseudo-labels are provided by SAM**. When using 100% of the labeled data versus 1%, we observe an improvement of 2.5 $AP_{50:95}^{bb}$ and 2.7 $AP_{50:95}^{seg}$, compared to gains of 1.3 AP_{50}^{bb} and 1.1 AP_{50}^{seg} percentage points (on scores that are already worse) for pseudo-labels generated without SAM.

V. CONCLUSION AND FUTURE WORKS

To the best of our knowledge, the proposed dataset of 110k images, along with our pseudo-labeling method based on SAM, is the first attempt to leverage unlabeled images in forests for tree detection and segmentation. Our principal contributions are the unlabeled dataset FLICKRTREE110K, FLICKRTREE100, along with FLICKRTREE100, which is a subset of 100 manually labeled images, a practical point prompt sampling strategy that improves the quality of pseudo-labels generated by SAM, and a comparative analysis of different prompting strategy and unlabelled dataset size. Overall, both the dataset and pseudo-label generation method significantly impacted the zero-shot detection and segmentation performances of the ViTDet models trained on them. We believe that our datasets will be a great addition to

the research community as abundant training data in forests, even unlabeled, is lacking.

In future works, we plan to improve on the pseudo-labels released with the dataset or even provide additional human-labeled images so future research can benchmark their results on a large, statistically significant test set. It would also be interesting to finetune HQ-SAM on forest images so it learns to output correctly segmented trees even with noisy prompt inputs.

VI. ACKNOWLEDGEMENTS

The authors would like to thank FORAC Research Consortium for the grant [CRDPJ 538321 - 18], Subventions Alliance (ALLRP), proposition intitulée « Classification des essences d'arbres par image plein pied », ALLRP 586940 - 23, and CFI Innovation Fund for grant number 39709.

REFERENCES

- [1] A. Ouaknine, T. Kattenborn, E. Laliberté, and D. Rolnick, "Openforest: A data catalogue for machine learning in forest monitoring," *arXiv preprint arXiv:2311.00277*, 2023.
- [2] A. Kirillov, E. Mintun, N. Ravi, *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 4015–4026.
- [3] T.-Y. Lin, M. Maire, S. Belongie, *et al.*, "Microsoft coco: Common objects in context," in *European Conference on Computer Vision (ECCV)*, Springer, 2014.
- [4] L. Ke, M. Ye, M. Danelljan, *et al.*, "Segment anything in high quality," in *NeurIPS*, 2023.
- [5] J. Hu, C. Chen, L. Cao, *et al.*, "Pseudo-label alignment for semi-supervised instance segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.
- [6] Z. Huang, H. Liu, H. Zhang, *et al.*, "Push the boundary of sam: A pseudo-label correction framework for medical segmentation," *arXiv preprint arXiv:2308.00883*, 2023.
- [7] D. Cheng, Z. Qin, Z. Jiang, S. Zhang, Q. Lao, and K. Li, "Sam on medical images: A comprehensive study on three prompt modes," *arXiv preprint arXiv:2305.00035*, 2023.
- [8] Y.-C. Liu, C.-Y. Ma, Z. He, *et al.*, "Unbiased teacher for semi-supervised object detection," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.
- [9] Q. Zhou, C. Yu, Z. Wang, Q. Qian, and H. Li, "Instant-teaching: An end-to-end semi-supervised object detection framework," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 4081–4090.
- [10] M. Xu, Z. Zhang, H. Hu, *et al.*, "End-to-end semi-supervised object detection with soft teacher," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

- [11] Y.-C. Liu, C.-Y. Ma, and Z. Kira, “Unbiased teacher v2: Semi-supervised object detection for anchor-free and anchor-based detectors,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [12] A. Tarvainen and H. Valpola, “Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results,” *Advances in neural information processing systems (NIPS)*, 2017.
- [13] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [14] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.
- [15] R. Bommasani, D. A. Hudson, E. Adeli, *et al.*, “On the opportunities and risks of foundation models,” *arXiv preprint arXiv:2108.07258*, 2021.
- [16] B. H. Wang, C. Diaz-Ruiz, J. Banfi, and M. Campbell, “Detecting and mapping trees in unstructured environments with a stereo camera and pseudo-lidar,” in *Proceedings of the IEEE International conference on robotics and automation (ICRA)*, IEEE, 2021.
- [17] D.-Q. da Silva, F.-N. Dos Santos, A.-J. Sousa, and V. Filipe, “Visible and thermal image-based trunk detection with deep learning for forestry mobile robotics,” *Journal of Imaging*, 2021.
- [18] V. Grondin, J.-M. Fortin, F. Pomerleau, and P. Giguère, “Tree detection and diameter estimation based on deep learning,” *Forestry*, 2023.
- [19] J. Lagos, U. Lempiö, and E. Rahtu, “Finnwoodlands dataset,” in *Scandinavian Conference on Image Analysis*, Springer, 2023.
- [20] X. Wang, R. Girdhar, S. X. Yu, and I. Misra, “Cut and learn for unsupervised object detection and instance segmentation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [21] V. Grondin, F. Pomerleau, and P. Giguère, “Training deep learning algorithms on synthetic forest images for tree detection,” *arXiv preprint arXiv:2210.04104*, 2022.
- [22] Y. Lu, Y. Huang, S. Sun, *et al.*, “M2fnet: Multi-modal forest monitoring network on large-scale virtual dataset,” *arXiv preprint arXiv:2402.04534*, 2024.
- [23] Y. Li, H. Mao, R. Girshick, and K. He, “Exploring plain vision transformer backbones for object detection,” in *European Conference on Computer Vision (ECCV)*, Springer, 2022.
- [24] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, *Detectron2*, <https://github.com/facebookresearch/detectron2>, 2019.