AdaCropFollow: Self-Supervised Test-Time Adaptation for Visual Under-Canopy Navigation

Arun N. Sivakumar[1], Federico Magistri[2], Jens Behley[2], Cyrill Stachniss[2,3], Girish Chowdhary[1]

***Abstract*— Under-canopy agricultural robots can enable various applications like precise monitoring, spraying, weeding, and plant manipulation tasks throughout the growing season. Autonomous navigation under the canopy is challenging due to the degradation in accuracy of RTK-GPS and the large variability in the visual appearance of the scene over time. Prior work commonly relies on supervised learning-based perception systems, such as semantic keypoint representations, but many failures arise from the model's inability to adapt to domain shifts during deployment in diverse field conditions. We propose a self-supervised test-time adaptation method that leverages vision foundation model features and a geometric prior-based pseudo-labeling criterion to adapt the keypoint representation. Our experiments show that with a small amount of unlabeled data, the keypoint prediction model from the source domain can be adapted in a self-supervised manner to various challenging target domains using our method on an embedded computer.**

I. Introduction

Agriculture faces major challenges due to the increasing demand for food, fiber, and energy from the growing world population, the shrinking availability of farmland and labor, and the environmental concerns due to over-application of farm inputs. Autonomous robots can help tackle these challenges by sustainably improving productivity in agricultural production as well as in crop improvement research. In particular, compact and low-cost under-canopy agricultural robots can enable various applications such as precise monitoring, high throughput phenotyping, spraying, weeding, and plant manipulation throughout the growing season.

Under-canopy navigation is challenging due to factors such as degradation in RTK-GPS accuracy under the canopy, limited visibility in this scenario, and less margin for error due to the limited traversable space available for the robot. Prior work [1] developed a vision and learning-based modular navigation system with semantic keypoint representation and deployed this system over large distances with significantly better performance in terms of the number of collisions than the prior state of the art [2]. However, the keypoint prediction model was trained with supervised learning using a large labeled dataset of crop row lines and used as a frozen model with no continuous adaptation during deployment. A large number of failures observed during the deployment can be attributed to the errors in keypoint prediction stemming from the inability to adapt to novel scenarios resulting from changes in visual appearance and lighting conditions.

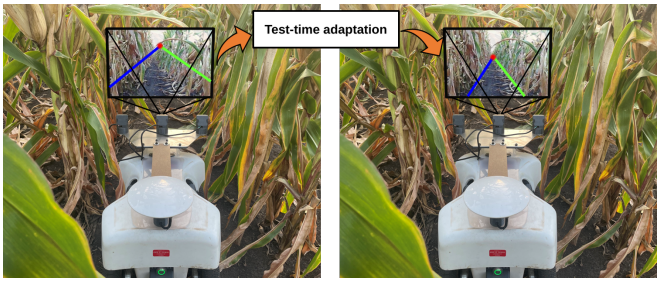

Fig. 1: Under-canopy robots with learning-based visual navigation systems encounter domain shifts due to large visual variations. We propose a self-supervised test-time adaptation method for semantic keypoints: vanishing point (red), left intercept point (blue), and right intercept point (red).

In this paper, we investigate the problem of self-supervised test-time adaptation of the keypoint model for under-canopy navigation. This implies that by using a small amount of unlabeled data from a target domain, we update the keypoint model with the compute available onboard the robot in a self-supervised manner, i.e., without additional manual labeling effort. The main research challenges associated with this problem include identifying an appropriate self-supervised loss function for the task and finding the small number of parameters that can be updated to adapt the network while considering the compute constraints of the robot.

A common approach for self-supervised adaptation is to use entropy minimization based on pseudo-labeling [3], [4]. However, the approach of choosing the pseudo-labels just based on the entropy of the predictions results in noisy labels when the domain shift is significant [5]. Instead we assume that we have access to a set of unlabeled images from the target domain during adaptation and also that the information on camera intrinsics and extrinsics, as well as an estimate of the lateral distance between the crop rows, is known. With this information, we define a geometric prior that is used as the criterion to select the pseudo-labels for model adaptation. Also, we use the vision foundation model DINOv3 [6] as our frozen feature encoder which enables updating only the parameters in the convolutional layers of the keypoint decoder to adapt the network.

The main contribution of this paper is a novel method for self-supervised test-time adaptation of an under-canopy keypoint prediction model that requires only small amounts

[1]Field Robotics Engineering and Sciences Hub (FRESH), University of Illinois Urbana-Champaign (UIUC), IL

[2]Center for Robotics, University of Bonn, Germany

[3]Lamarr Institute for Machine Learning and Artificial Intelligence, Germany

This work was supported in part by AIFARMS #1024178, NSF-USDA COALESCE #2021-67021-34418, NSF NRI 2.0 NIFA #2021-67021-33449, USDA grants iCOVER(#NR233A750004G066), iFARM(#2022-77038-37306), and NSF STTR #1951250.

of unlabeled data from the target domain and little compute available in an embedded computer to generalize to the target domain. We formulate a geometric prior based on known crop row structure in the environment and use this as the criterion for pseudo-label generation. We also leverage the powerful feature extractor from a transformer-based vision foundation model. Such an approach enables us to adapt the keypoint prediction model to diverse, challenging target domains without additional human labels.

In sum, we make three key claims: (i) our approach minimizes the error in prediction of semantic keypoints in different challenging and previously unseen under-canopy target domains without requiring additional human labels; (ii) our formulation of geometric prior for pseudo-labeling based on crop row spacing and camera properties is critical for achieving good performance; (iii) using frozen backbone from a transformer-based vision foundation model is necessary to achieve this performance across crop variants. These claims are backed up by the paper and our experimental evaluation.

II. Related Work

Vision-based navigation in agricultural environments has evolved significantly over the past three decades, beginning with classical computer vision techniques that relied on edge detection, color segmentation, and Hough transforms to extract linear crop-row geometry [7], [8], [9]. As the demand for robustness across varying growth stages and complex lighting conditions grew, the field transitioned toward learning-based perception systems but semantic segmentation of crop pixels from background remained as the output representation [10]. However, all these works focused on a viewpoint above the canopy. For under-canopy environments that are characterized by clutter and occlusion, Sivakumar et al. [2] proposed an end-to-end perception system that directly predicts the angle and offset of the robot relative to the crop rows from RGB image input. However, this end-to-end perception representation is sensitive to variations in camera extrinsics and lacks interpretability for humans. Later, Sivakumar et al. [1] proposed using the semantic keypoint representation similar to object pose literature in manipulation to overcome the above limitations and demonstrated large-scale field deployment with this system.

Deploying learning-based visual navigation systems in outdoor environments, like agricultural fields, results in severe domain shifts due to variations in lighting, weather, and crop varieties. To mitigate domain shift, self-supervised visual navigation approaches often utilize proprioceptive feedback or predictive costmaps to learn traversability directly from robot experience in the wild [11], [12]. However, these methods are largely designed for open unstructured off-road terrains and they do not take advantage of the known geometric information like in our target environment of crop rows. Consequently, test-time adaptation (TTA) has emerged as a mechanism to continually refine network weights using unlabeled deployment data. While standard TTA methods often rely on entropy minimization for pseudo-labeling [13], such techniques suffer from noisy labels when the domain shift is significant [5]. Other works have demonstrated that cross-view and cross-modal geometric consistency provides a much more robust self-supervisory signal for adaptation [14], [15], [16], [17]. Building on this insight, our approach departs from entropy-based filtering and instead defines a geometric prior from crop row spacing and camera properties to select reliable pseudo-labels for self-supervised TTA.

Finally, the recent emergence of Vision Foundation Models (VFMs) has provided a powerful mechanism to combat perceptual domain shifts. While several works attempt to train generalist vision-language-action policies across diverse robotic embodiments [18], [19], these generalist models often struggle in specialized field applications requiring precision, such as navigating tight crop rows. Several works have shown that with pre-trained vision encoders, such as DINOv2 [20], as frozen feature extractors we can finetune small, task-specific downstream modules for traversability, planning, or mapping [21], [22]. In contrast, our method exploits the rich, task-agnostic representations of a frozen transformer-based VFM, adapting only the lightweight convolutional decoder without any human annotations.

III. Our Approach to Self-Supervised Adaptation of Semantic Keypoints

Given a machine learning model trained with supervised learning for under-canopy visual row following in a source domain, our objective is to adapt it using a small amount of unlabeled data to target domains that exhibit substantial variation in visual appearance. The model takes an RGB image as input and predicts three semantic keypoints relevant to this task: the vanishing point, the left intercept point, and the right intercept point. To address the domain shift between source and target environments, we propose a self-supervised adaptation method based on pseudo-labeling. Specifically, we identify target-domain images where the source-trained model produces geometrically consistent predictions and use the geometry of the crop rows to construct supervisory signals from them to iteratively adapt the model. Finally, inspired by the recent success of vision foundation models in outdoor visual navigation, we leverage them as frozen feature extractors. Fig. 2 illustrates the overall method.

A. Source Domain Pretraining

Given a source-domain dataset with left and right crop rows annotated as lines, we train a semantic keypoint prediction model. Given an RGB image $I \in \mathbb{R}^{3\times H\times W}$, where H and W are the height and width of the image, the model predicts semantic keypoint heatmaps $\hat{Y} \in \mathbb{R}^{3\times H_h\times W_h}$ corresponding to the vanishing point (VP), left intercept point (LP), and right intercept point (RP), similar to CropFollow++ [1]. H_h and W_h denote the height and width of the heatmaps.

The network consists of a frozen DINOv3 ViT-S/16 backbone $\phi_\theta : \mathbb{R}^{3\times H\times W} \rightarrow \mathbb{R}^{C\times H_f\times W_f}$ that extracts high-level features $F = \phi_\theta(I)$, where C, H_f, and W_f denote the dimensions of the output feature map. We use ViT-S/16,

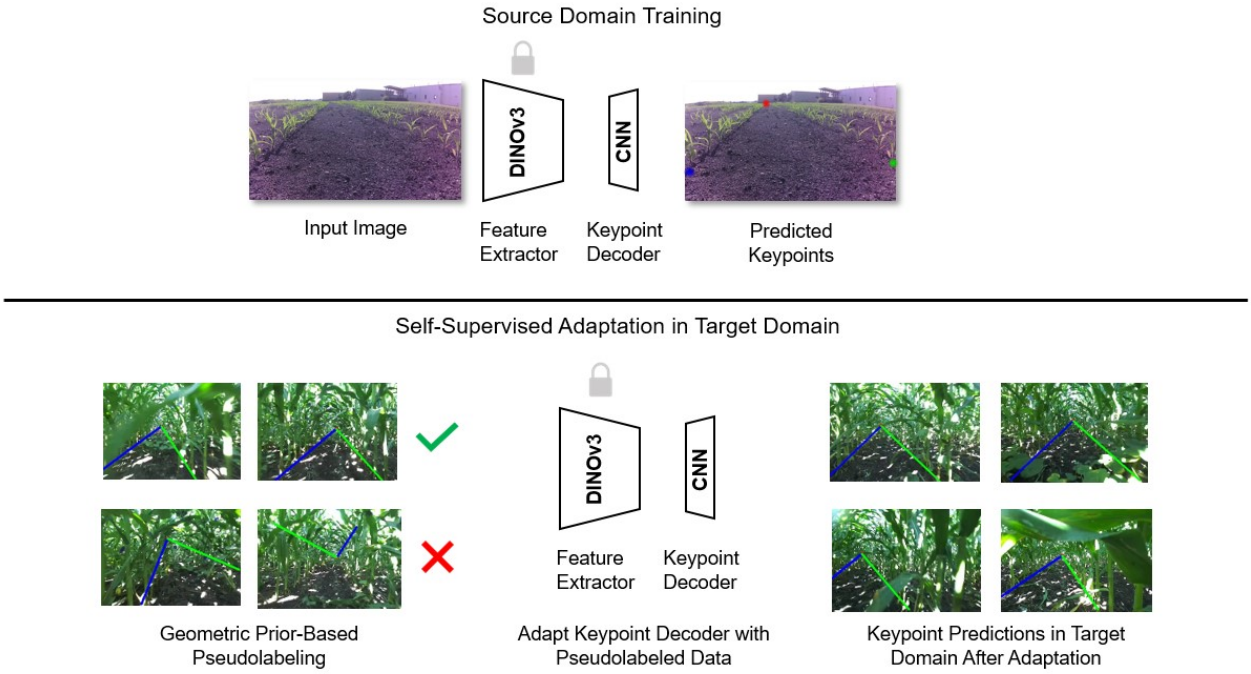

Fig. 2: AdaCropFollow overview. Given a trained model in the source domain (upper part) that is trained to predict semantic keypoints: vanishing point (red), left intercept point (blue), and right intercept point (red), we adapt the model in a target domain (lower part). To this end, we use geometric priors to identify reliable pseudo-labels, which are used to adapt the keypoint decoder to the target domain.

the smallest DINOv3 variant (≈21 M parameters), to keep the memory footprint compatible with resource-constrained embedded hardware. A lightweight three-layer convolutional decoder $g_{\mathbf{w}} : \mathbb{R}^{C\times H_f\times W_f} \to \mathbb{R}^{3\times H_h\times W_h}$ maps these features to the predicted heatmaps $\hat{Y} = g_{\mathbf{w}}(F)$.

From the labeled crop row lines, the vanishing point is calculated as their intersection. The left and right intercept points are defined by the intersection of their respective crop row lines with the bottom edge of the image. Since both intercepts are obtained by extending the crop row lines to the image boundary, they remain well defined even when the canopy occludes the ground and the base of the plants. Target heatmaps $Y \in \mathbb{R}^{3\times H_h\times W_h}$ are generated by placing Gaussian kernels of standard deviation $\sigma = 1$ at the labeled vanishing point, left intercept point, and right intercept point. During training, the backbone ϕ_θ is frozen, and the decoder parameters $\mathbf{w}$ are optimized to minimize the channel-wise Kullback-Leibler divergence KL between target Y_k and predicted heatmaps $\hat{Y}_k$, where each channel is normalized with a spatial softmax:

$$\mathcal{L}_{\mathrm{KL}} = \sum_{k\in\{\mathrm{VP,LP,RP}\}} \mathrm{KL}\big(Y_k \,\|\, \hat{Y}_k\big) \tag{1}$$

For each keypoint k, we extract the discrete location with the highest intensity in the softmax-normalized heatmap, $(\hat{\imath}_k, \hat{\jmath}_k) = \arg\max_{i,j} \hat{Y}_{k,i,j}$, with confidence $\hat{s}_k = \max(\hat{Y}_k)$, and linearly scale these indices by the ratio of image to heatmap dimensions to obtain the pixel coordinates $(\hat{u}_k, \hat{v}_k)$ in the full image frame.

B. Geometric Prior for Keypoints from Crop Row Structure

Let D_{row} denote the physical spacing between the left and right crop rows, and h_{cam} the camera height above the ground. We define the canonical orientation of the robot to be the case where the roll and pitch are aligned with the ground plane and the yaw is along the crop row direction. In the canonical orientation, the ground plane at the bottom edge of the image ($v = H$) lies at a constant depth $Z_0 = f h_{\mathrm{cam}}/(H - c_v)$ from the camera, where f is the focal length in pixels and (c_u, c_v) is the principal point. The horizontal pixel distance between the left and right crop row intercepts at this edge, denoted u_{row}, is therefore independent of the robot's lateral placement:

$$u_{\mathrm{row}} = \frac{f\, D_{\mathrm{row}}}{Z_0} = \frac{D_{\mathrm{row}}\,(H - c_v)}{h_{\mathrm{cam}}} \tag{2}$$

We define the robot's lateral placement relative to the crop rows using the dimensionless distance ratio $d \in [0, 1]$, where 0 corresponds to the camera centered over the left crop row and 1 to the right crop row. The canonical left and right intercept coordinates u_{LP} and u_{RP} relate to the distance ratio d as:

$$u_{\mathrm{LP}}(d) = c_u - d\, u_{\mathrm{row}}, \quad u_{\mathrm{RP}}(d) = c_u + (1 - d)\, u_{\mathrm{row}} \tag{3}$$

For a robot with an allowable operational lateral placement range of $[d_{\min}, d_{\max}]$, we establish the canonical geometric bounds for the left and right intercept points directly in the

image plane:

$$u_{\mathrm{LP}}^{\min} = c_u - d_{\max}\, u_{\mathrm{row}} \tag{4}$$
$$u_{\mathrm{LP}}^{\max} = c_u - d_{\min}\, u_{\mathrm{row}} \tag{5}$$
$$u_{\mathrm{RP}}^{\min} = c_u + (1 - d_{\max})\, u_{\mathrm{row}} \tag{6}$$
$$u_{\mathrm{RP}}^{\max} = c_u + (1 - d_{\min})\, u_{\mathrm{row}} \tag{7}$$

The bounds $u_{\mathrm{LP}}^{\min}, u_{\mathrm{RP}}^{\min}, u_{\mathrm{LP}}^{\max}, u_{\mathrm{RP}}^{\max}$, and the width u_{row} are defined in the canonical ground-aligned coordinate space. We use these values as a geometric prior to formulate the criteria for pseudo-labeling in the target domain.

C. Pseudo-label Generation in the Target Domain

Let $\mathcal{I}_{\mathrm{target}} = \{I_1, \ldots, I_N\}$ denote the set of unlabeled images from the target domain. For each image $I \in \mathcal{I}_{\mathrm{target}}$, we use the model trained in the source domain to predict the pixel coordinates $(\hat{u}_k, \hat{v}_k)$ and their associated confidences $\hat{s}_k$ for each keypoint $k \in \{\mathrm{VP}, \mathrm{LP}, \mathrm{RP}\}$. Our objective is to identify a subset of images $\mathcal{I}_{\mathrm{pseudo}} \subseteq \mathcal{I}_{\mathrm{target}}$ whose keypoint predictions are sufficiently reliable to construct pseudo-labels for domain adaptation.

For each predicted keypoint $\hat{\mathbf{p}}_k = [\hat{u}_k, \hat{v}_k]^\top$, we define its homogeneous representation as $\hat{\tilde{\mathbf{p}}}_k = [\hat{u}_k, \hat{v}_k, 1]^\top$. Given the predicted vanishing point $\hat{\mathbf{p}}_{\mathrm{VP}}$ and IMU measurements of roll ϕ and pitch θ, we first transform the VP to the ground-aligned image plane via the roll-pitch homography $\mathbf{H}_{\phi,\theta}$:

$$\tilde{\mathbf{p}}'_{\mathrm{VP}} = \mathbf{H}_{\phi,\theta}\hat{\tilde{\mathbf{p}}}_{\mathrm{VP}}, \text{ where } \mathbf{H}_{\phi,\theta} = \mathbf{K}\,\mathbf{R}_x(\theta)\,\mathbf{R}_z(\phi)\,\mathbf{K}^{-1} \tag{8}$$

Here, $\mathbf{R}_z(\phi)$ and $\mathbf{R}_x(\theta)$ denote rotations about the optical axis and the horizontal image axis, compensating roll and pitch sequentially, and $\mathbf{K} \in \mathbb{R}^{3\times3}$ is the camera intrinsic matrix. The transformed 2D coordinates $\hat{\mathbf{p}}'_{\mathrm{VP}} = [\hat{u}'_{\mathrm{VP}}, \hat{v}'_{\mathrm{VP}}]^\top$ are obtained by dividing the first two components of $\tilde{\mathbf{p}}'_{\mathrm{VP}}$ by its third component. The heading angle ψ is then computed relative to the principal point:

$$\psi = \arctan\left(\frac{\hat{u}'_{\mathrm{VP}} - c_u}{f}\right) \tag{9}$$

Using the estimated heading ψ and IMU orientation, we define a canonical homography that aligns the image plane orientation with the crop rows and the ground:

$$\mathbf{H}_{\mathrm{can}} = \mathbf{K}\,\mathbf{R}_y(-\psi)\,\mathbf{R}_x(\theta)\,\mathbf{R}_z(\phi)\,\mathbf{K}^{-1} \tag{10}$$

We use this homography to calculate the canonical keypoints:

$$\hat{\tilde{\mathbf{p}}}_{k,\mathrm{can}} = \mathbf{H}_{\mathrm{can}}\hat{\tilde{\mathbf{p}}}_k \tag{11}$$

The canonical horizontal intercepts $\hat{u}_{\mathrm{LP,can}}$ and $\hat{u}_{\mathrm{RP,can}}$ are calculated by extending the lines connecting the canonical vanishing point with the canonical left and right intercept points to the bottom of the image. A target image is accepted for pseudo-labeling if it satisfies the following geometric constraints:

1) **VP Confidence:** The VP confidence exceeds a threshold:
$$C_1(I) : \hat{s}_{\mathrm{VP}} \geq \tau \tag{12}$$
2) **VP Alignment:** The roll-pitch rectified VP lies within a vertical tolerance ϵ_{VP} scaled by image height H:
$$C_2(I) : |\hat{v}'_{\mathrm{VP}} - c_v| \leq \epsilon_{\mathrm{VP}} \cdot H \tag{13}$$
3) **Intercept Bounds:** The canonical left and right intercept points lie within the bounds defined by the geometric prior:
$$C_3(I) : \begin{matrix} u_{\mathrm{LP}}^{\min} \leq \hat{u}_{\mathrm{LP,can}} \leq u_{\mathrm{LP}}^{\max}, \\ u_{\mathrm{RP}}^{\min} \leq \hat{u}_{\mathrm{RP,can}} \leq u_{\mathrm{RP}}^{\max} \end{matrix} \tag{14}$$
4) **Projected Row Width:** The distance between the canonical intercepts lies within scaled bounds of the known horizontal pixel distance u_{row}:
$$C_4(I) : \epsilon_{\min} u_{\mathrm{row}} \leq (\hat{u}_{\mathrm{RP,can}} - \hat{u}_{\mathrm{LP,can}}) \leq \epsilon_{\max} u_{\mathrm{row}} \tag{15}$$

The final pseudo-label set is defined as:

$$\mathcal{I}_{\mathrm{pseudo}} = \left\{ I \in \mathcal{I}_{\mathrm{target}} \mid \bigwedge_{j=1}^{4} C_j(I) \right\} \tag{16}$$

Rather than directly reusing the raw predicted keypoints of accepted images as supervision, we use the geometric prior to construct corrected pseudo-labels. For each accepted image, we estimate the distance ratio by inverting Eq. (3) at the two canonical intercepts and averaging:

$$\hat{d} = \frac{1}{2}\left(\frac{c_u - \hat{u}_{\mathrm{LP,can}}}{u_{\mathrm{row}}} + 1 - \frac{\hat{u}_{\mathrm{RP,can}} - c_u}{u_{\mathrm{row}}}\right) \tag{17}$$

Given $\hat{d}$, we place ideal keypoints in the canonical view, with the vanishing point at the principal point and the intercepts at $u_{\mathrm{LP}}(\hat{d})$ and $u_{\mathrm{RP}}(\hat{d})$ on the bottom edge, and map them back to the original image via the inverse homography $\mathbf{H}_{\mathrm{can}}^{-1}$. The resulting keypoints serve as the pseudo-label for the image.

D. Model Adaptation with Pseudo-labels

Starting from the source-trained model described earlier, we adapt the keypoint decoder to the target domain using the pseudo-labeled images $\mathcal{I}_{\mathrm{pseudo}}$. From the constructed pseudo-label keypoint locations, keypoint heatmaps are created in the same manner as in the source domain. Training proceeds by minimizing the KL divergence in Eq. (1) between predicted heatmaps and pseudo-label heatmaps and updating the parameters of the decoder, as in pre-training. This process of generating pseudo-labels is repeated for five iterations. In each iteration, the entire target dataset $\mathcal{I}_{\mathrm{target}}$ is re-evaluated using the updated model to form a new set $\mathcal{I}_{\mathrm{pseudo}}$. This allows the model to gradually adapt to the visual characteristics of the target domain by incorporating images that may not have met the reliability criteria in earlier iterations.

E. Implementation Details

We use an input image size of 224×320 pixels. The output heatmap resolution is set to 56×80. We use a horizontal flip as an augmentation. The network is trained with a batch size of eight using the Adam optimizer [23] with a learning rate of

10^{-4}, a weight decay of 10^{-5}, and a learning rate scheduler coefficient of 0.9. These hyperparameters remain unchanged during both source domain training and target domain adaptation. The network was trained for 50 epochs during source training. During adaptation, five pseudo-labeling iterations were performed with only one epoch of training in each iteration. The DINOv3 ViT-S/16 backbone remains frozen during both stages. Only the parameters of the convolutional decoder are updated during source training and adaptation. For pseudo-label generation, the VP confidence threshold τ is 0.5, the VP alignment factor ϵ_{VP} is 0.25, and the projected row width factors $\epsilon_{\min}$ and $\epsilon_{\max}$ are 0.75 and 1.25, respectively.

IV. Experimental Evaluation

The main focus of this work is a method for self-supervised adaptation of semantic keypoints to minimize errors in perception from domain shift for visual navigation in challenging under-canopy environments. We present our experiments to show the capabilities of our method. The results of our experiments support our key claims, which are: (i) our approach minimizes the error in prediction of semantic keypoints in different challenging and previously unseen under-canopy target domains without requiring additional human labels; (ii) our formulation of geometric prior for pseudo-labeling based on crop row spacing and camera properties is critical for achieving good performance; (iii) using frozen backbone from a transformer-based vision foundation model is necessary to achieve this performance across crop variants.

A. Experimental Setup

We use datasets from prior works [2], [25] to evaluate our approach. From the large under-canopy dataset of Sivakumar et al. [2], we create a subset representing only the early growth stage of corn. This subset serves as our source domain for supervised training of the semantic keypoint prediction model. The dataset contains 6733 images of early season corn (as seen in the top row of Fig. 2) collected across 15 trajectories from different field and lighting conditions in the same early season growth stage.

For self-supervised adaptation experiments, we use the dataset introduced by Cuaran et al. [25], which provides synchronized stereo RGB images and IMU data collected with a ZED camera in challenging late-season corn environments. From this dataset, we select three distinct target environments, each treated as a separate target domain. To further test the generality of our method, we additionally collected a new dataset in an orchard farm using the same ZED camera setup, which we treat as a fourth target domain.

For each target domain, we sample 1000 unlabeled images (corresponding to less than one minute of driving) for adaptation. We annotate 100 frames from the remaining trajectory in each domain to serve as ground truth for evaluation only. Note that the unlabeled data used for adaptation and data labeled for evaluation in each target domain belong to non-overlapping segments of the trajectory. Using synchronized IMU roll angle and known camera intrinsics, we also compute the robot's heading and distance ratio relative to the crop rows, as described in Sec. III.

Evaluation is performed using the mean absolute error (MAE) of the normalized pixel coordinates of the predicted keypoints with respect to the annotated keypoints, as well as the MAE of the heading and distance ratio derived from the predicted keypoints with respect to those derived from the annotations. Heading and distance ratio are the state estimates consumed directly by the downstream row-following controller in CropFollow++ [1], which has been validated in large-scale closed-loop field deployments; reductions in these errors therefore translate directly to the navigation task rather than reflecting pixel accuracy alone. To assess the computational feasibility of our approach for edge deployment, all adaptation experiments are executed on an NVIDIA Jetson AGX Orin 64GB embedded device, and we report the average adaptation time.

B. Main Experiment

The first experiment evaluates the performance of our approach and the results shown in Tab. I support our claim that our approach minimizes the error in prediction of semantic keypoints considerably in different under-canopy target domains without requiring human labels. This experiment also shows that our formulation of geometric prior for pseudo-labeling based on crop row spacing and camera properties is critical for achieving good performance.

We compare our method against two task-agnostic pseudo-labeling baselines: TENT [4] and MixMatch [24]. In our implementation of TENT, we check whether the confidence of all three keypoints is above the threshold as the criterion to pick pseudo-labels. For MixMatch, we use horizontal flip as the augmentation since that is the relevant geometric augmentation for our task. We check whether the confidence of the vanishing point is above the threshold and if the predicted intercept locations in the original image and the mirrored counterpart in the flipped image fall within the defined bounds. We also compare against the zero-shot generalization of CropFollow [2] and CropFollow++ [1] models trained on the source domain without target adaptation. Since CropFollow directly predicts heading and distance ratio without intermediate semantic keypoints, only those navigational errors are reported.

Across all target domains our approach consistently shows lower MAE in normalized pixel coordinates of keypoints than baselines. In the case of vanishing point, our approach as well as the baselines with adaptation use prediction confidence as the selection criterion, resulting in comparable vanishing point error for all approaches. Since heading calculation depends solely on the vanishing point and IMU roll angle, similar results are observed in heading MAE. Compared to the methods without adaptation (CropFollow and CropFollow++), our heading error remains consistently low at roughly 0.03 rad across all four target domains, which is a decrease of 34% to 82% compared to the better-performing zero-shot baseline in each domain.

TABLE I: Mean L_1 error metrics across 4 target domains. Keypoints (VP, LP, RP) are in normalized pixels, heading is in radians, and distance ratio is unitless. The ↓ indicates that lower values are better.

Approach	Keypoints ↓ (normalized pixels)			Heading ↓ (rad)	Distance Ratio ↓	Keypoints ↓ (normalized pixels)			Heading ↓ (rad)	Distance Ratio ↓
	VP	LP	RP			VP	LP	RP		
	Corn 1					**Corn 2**				
TENT [4]	0.025	0.218	0.189	**0.026**	0.208	**0.025**	0.103	0.140	**0.026**	0.133
MixMatch [24]	0.027	0.214	0.188	0.027	0.208	0.028	0.065	0.093	0.031	0.074
AdaCropFollow (Ours)	**0.014**	**0.027**	**0.038**	0.027	**0.080**	0.027	**0.033**	**0.030**	0.027	**0.051**
CropFollow [2]	-	-	-	0.093	0.133	-	-	-	0.109	0.153
CropFollow++ [1]	0.053	0.112	0.216	0.119	0.263	0.051	0.099	0.109	0.113	0.179
	Corn 3					**Orchard**				
TENT [4]	0.023	0.142	0.130	0.030	0.110	0.031	0.009	0.087	0.043	0.128
MixMatch [24]	0.021	0.135	0.140	**0.025**	0.105	0.029	**0.009**	0.038	0.034	**0.054**
AdaCropFollow (Ours)	**0.020**	**0.039**	**0.033**	0.033	**0.076**	**0.013**	0.013	**0.022**	**0.033**	0.087
CropFollow [2]	-	-	-	0.050	0.152	-	-	-	0.180	0.186
CropFollow++ [1]	0.043	0.174	0.150	0.058	0.232	0.069	0.289	0.262	0.185	0.452

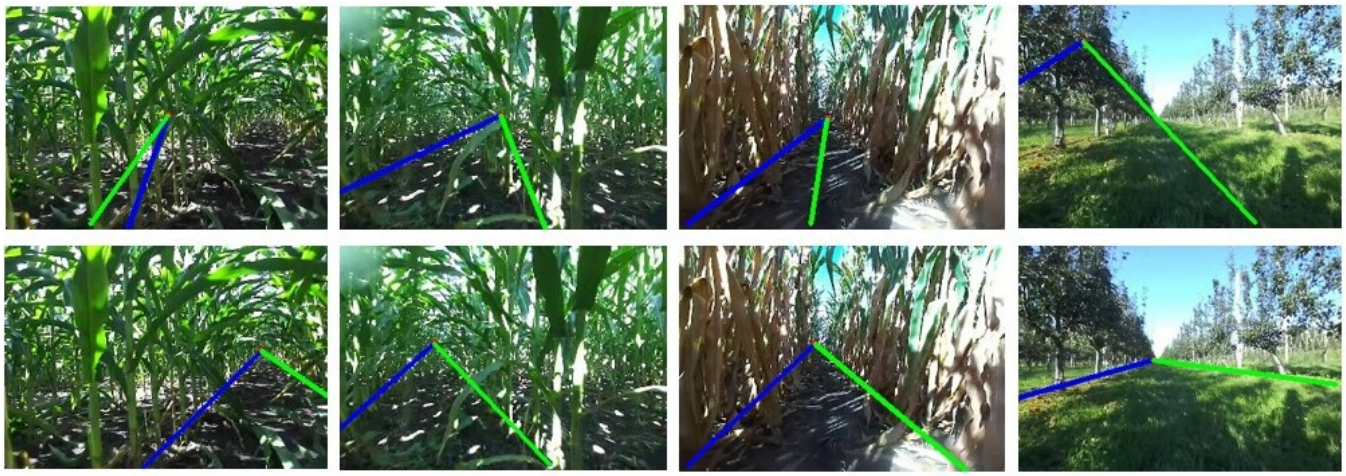

Fig. 3: Visualization of predicted keypoints in each target domain. Top row shows predictions from CropFollow++ trained only on the source domain without adaptation. Bottom row shows predictions after self-supervised adaptation using our method.

We can see the large difference between the performance of our approach and other baselines with adaptation in the pixel norm error of the left and right intercept keypoints. Unlike the baselines, we use the geometric prior of the crop row structure both as the criterion to select pseudo-labels for the intercept keypoints and to construct the pseudo-labels themselves (Sec. III), and this helps considerably across all 4 target domains. Our approach achieves an intercept pixel norm error less than 0.04 in all target domains whereas the lowest baseline errors in the corn domains hover between 0.065 and 0.188. This shows that, other than the Orchard dataset where the error is comparable, our geometric prior guided pseudo-labeling criterion enables more effective adaptation of the keypoints. We see the same difference in distance ratio prediction, where our approach achieves an error between 0.051 and 0.08 in the corn domains. This is a decrease of 28% to 62% compared to the best adaptation baseline, and a 40% to 67% reduction compared to CropFollow without adaptation. The considerable gap in heading and distance error between our approach and the method without adaptation shows the large domain gap between the source domain and the target domains and the need for self-supervised adaptation when deploying in real field conditions. Considering both the error in heading and distance ratio as the metric, our approach considerably outperforms the baselines with and without adaptation in nearly all the target domains. Unlike the late-season under-canopy environment, which is characterized by significant occlusion from leaves, the orchard environment exhibits clear visual difference between the ground and the crop row line which could be the reason for the flip augmentation baseline achieving comparable performance to our approach in that environment but not in other environments.

Fig. 3 shows qualitative keypoint predictions across challenging viewpoints and lateral robot displacements in each target domain. Compared to CropFollow++ without adaptation, our method enables very precise prediction of the keypoints after adaptation across all datasets.

C. *Effect of Different Backbones as Feature Extractors*

The second experiment evaluates the adaptation performance with different feature extractors for the keypoint prediction model and illustrates that using frozen backbones from transformer-based vision foundation models is necessary to achieve the best performance across diverse crop conditions. We train different keypoint prediction models

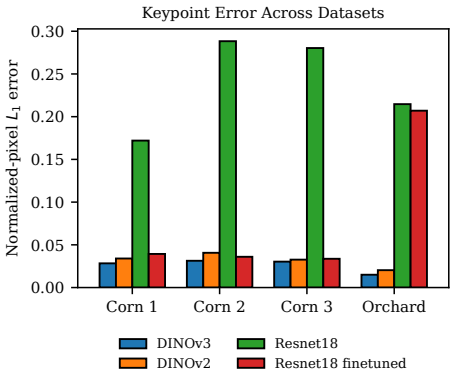

Fig. 4: Comparison of keypoint pixel norm L_1 error with different feature backbones.

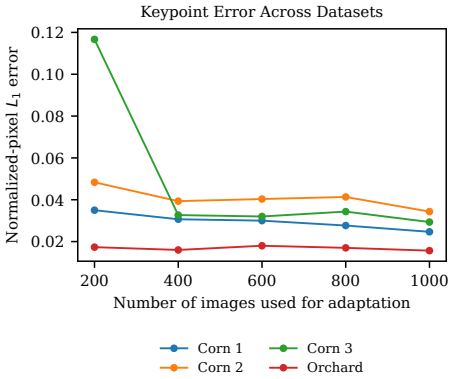

Fig. 5: Comparison of keypoint pixel norm L_1 error with different number of images used for adaptation.

with 4 different pretrained vision backbones as feature extractors: 1) ResNet18 backbone that is frozen during source domain training and adaptation 2) ResNet18 backbone that is finetuned during source domain training but is not changed during target domain adaptation 3) Vision foundation model DINOv2 [20] that is frozen during source domain training and adaptation 4) Vision foundation model DINOv3 [6] that is frozen during source domain training and adaptation. While prior works on vision-based under-canopy navigation used ResNet18 backbone that is finetuned during training, recent success of the frozen DINOv2 foundation model in vision-based outdoor navigation tasks motivated our choice. We use the mean absolute error in pixel norm across all three keypoints as the metric for this experiment. The results are shown as a bar chart in Fig. 4. In all 4 target domains, the frozen DINOv3 backbone results in the lowest error. Overall both frozen DINOv2 and DINOv3 backbones show consistently lower prediction error across all target domains than ResNet18 backbones. The difference in error is considerably large in Orchard domain where the pixel norm error decreases by more than 90% when we choose the frozen DINO models instead of ResNet18 as the feature extractor. In Corn 1, Corn 2 and Corn 3, the pixel norm error decreases by 27.96%, 12.96% and 9.9% respectively when we choose the frozen DINOv3 model instead of ResNet18 model that is finetuned in the source domain.

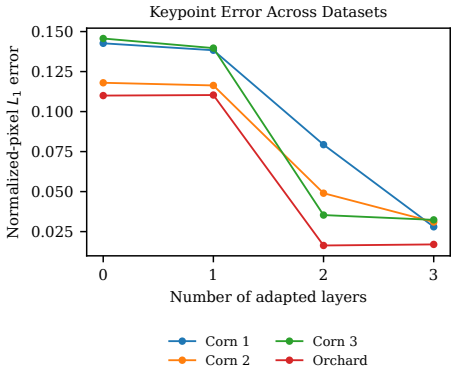

Fig. 6: Comparison of keypoint pixel norm L_1 error by changing the number of layers adapted in the decoder.

D. Ablation Study

The last experiment evaluates the impact of the size of data used for adaptation and the number of parameters modified during adaptation on the mean absolute pixel norm error. To deploy this system in real field conditions, it is essential to adapt to different target domains using a small number of unlabeled images. Also considering the computational constraints on embedded computers used in mobile robots, updating a small number of parameters during adaptation is desired. These two important considerations motivate this ablation study. Fig. 5 shows the change in mean absolute error in pixel norm across all three keypoints as we increase the number of unlabeled images used for adaptation from 200 to 1000. Other than Corn 3, the pixel norm error reduces to less than 0.04 with just 200 images used for adaptation and we reach this similar error with 400 images in Corn 3. Note that using 200 images corresponds to less than 15 s of video data from the target domain in our case. This indicates the potential for rapid adaptation in challenging target domains with our method.

Fig. 6 shows the change in mean absolute error in pixel norm as we change the number of layers in the convolutional decoder that is updated during the adaptation phase. In this case, zero layers refers to only updating the batch norm parameters across the entire decoder but not weight and bias parameters in any layer. In total there are 768 learnable parameters if updating only batch norm. Updating one layer refers to updating just the final head layer which has 195 parameters. In the case of two and three layers, we update more convolutional layers corresponding to 110,043 and 701,379 parameters respectively. While updating just batch norm parameters or the final layer is not enough, updating only two layers minimizes the pixel norm error to be lower than 0.05 in Corn 2, Corn 3 and Orchard. This indicates that when updating all decoder layers is not feasible, updating only the last two layers can result in considerably lower error in several target domains.

We performed all our experiments on an NVIDIA Jetson AGX Orin embedded computer which is commonly used

in mobile robots. Our method on average requires 312 s to finish the adaptation process when updating all three layers in the convolutional decoder and using 1000 unlabeled images from the target domain. This shows that our method is suitable for test-time adaptation in real field conditions on board the robot. Moreover, frozen ViT-S encoders of the same scale as our backbone have been demonstrated to support real-time, onboard closed-loop navigation on the same embedded platform [21], indicating that the inference cost of our architecture is compatible with deployment during navigation.

V. Conclusion

In this paper, we presented a novel approach to self-supervised adaptation of semantic keypoints to unseen environments for visual under-canopy navigation. Our approach uses pseudo-labeling with a novel geometric prior based on the known crop row structure, combined with frozen features from a transformer-based vision foundation model, allowing adaptation to very diverse target domains without human labels. We evaluated our approach on diverse datasets against existing techniques and supported all claims made in this paper. The experiments suggest that our adaptation approach yields more accurate prediction of semantic keypoints across different field conditions and crop types.

Despite these encouraging results, some limitations remain. Our geometric prior assumes that crop rows project as straight lines through a common vanishing point. Per-frame roll and pitch correction makes this robust to local terrain unevenness; however, fields with continuously varying slopes, where crop rows appear as curves in the image, remain an edge case to be evaluated. In addition, adaptation requires that a minimal set of source-model predictions pass the reliability criteria in the first iteration, which may fail under extreme domain shift. As future work, incorporating temporal consistency of the predicted keypoints over a short horizon into the pseudo-labeling criterion can improve pseudo-label reliability and yield in early iterations, enabling faster adaptation with less data.

References

[1] A. N. Sivakumar, M. V. Gasparino, M. McGuire, V. A. H. Higuti, M. U. Akcal, and G. Chowdhary, "Demonstrating CropFollow++: Robust Under-Canopy Navigation with Keypoints," in *Proc. of Robotics: Science and Systems (RSS)*, Delft, Netherlands, July 2024.

[2] A. N. Sivakumar, S. Modi, M. V. Gasparino, C. Ellis, A. E. Baquero Velasquez, G. Chowdhary, and S. Gupta, "Learned Visual Navigation for Under-Canopy Agricultural Robots," in *Proc. of Robotics: Science and Systems (RSS)*, Virtual, July 2021.

[3] D.-H. Lee *et al.*, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Proc. of ICML Workshop on challenges in representation learning*, 2013.

[4] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," in *Proc. of Intl. Conf. on Learning Representations (ICLR)*, 2021.

[5] J. Lee, D. Jung, S. Lee, J. Park, J. Shin, U. Hwang, and S. Yoon, "Entropy is not enough for test-time adaptation: From the perspective of disentangled factors," in *Proc. of the Intl. Conf. on Learning Representations (ICLR)*, 2024.

[6] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa, *et al.*, "DINOv3," *arXiv preprint arXiv:2508.10104*, 2025.

[7] J. F. Reid, *The development of computer vision algorithms for agricultural vehicle guidance*. Texas A&M University, 1987.

[8] A. English, P. Ross, D. Ball, and P. Corke, "Vision based guidance for robot navigation in agriculture," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2014, pp. 1693–1698.

[9] A. Ahmadi, L. Nardi, N. Chebrolu, and C. Stachniss, "Visual servoing-based navigation for monitoring row-crop fields," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2020, pp. 4920–4926.

[10] R. De Silva, G. Cielniak, G. Wang, and J. Gao, "Deep learning-based crop row detection for infield navigation of agri-robots," *Journal of Field Robotics (JFR)*, vol. 41, no. 7, pp. 2299–2321, 2024.

[11] G. Kahn, P. Abbeel, and S. Levine, "BADGR: An autonomous self-supervised learning-based navigation system," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1312–1319, 2021.

[12] M. V. Gasparino, A. N. Sivakumar, Y. Liu, A. E. Velasquez, V. A. Higuti, J. Rogers, H. Tran, and G. Chowdhary, "WayFAST: Navigation with predictive traversability in the field," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 651–10 658, 2022.

[13] T. Lee, J. Tremblay, V. Blukis, B. Wen, B.-U. Lee, I. Shin, S. Birchfield, I. S. Kweon, and K.-J. Yoon, "TTA-COPE: Test-time adaptation for category-level object pose estimation," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 21 285–21 295.

[14] J. Knights, S. Hausler, S. Sridharan, C. Fookes, and P. Moghadam, "GeoAdapt: Self-supervised test-time adaptation in lidar place recognition using geometric priors," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 915–922, 2023.

[15] H. Blum, F. Milano, R. Zurbrügg, R. Siegwart, C. Cadena, and A. Gawel, "Self-improving semantic perception for indoor localisation," in *Proc. of the Conf. on Robot Learning*. PMLR, 2022, pp. 1211–1222.

[16] H. Yang, W. Dong, L. Carlone, and V. Koltun, "Self-supervised geometric perception," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 14 350–14 361.

[17] Y. Yuan, Q. Shen, S. Wang, X. Yang, and X. Wang, "Test3R: Learning to reconstruct 3D at test time," in *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[18] D. Shah, A. Sridhar, A. Bhorkar, N. Hirose, and S. Levine, "GNM: A general navigation model to drive any robot," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 7226–7233.

[19] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. P. Foster, P. R. Sanketi, Q. Vuong, T. Kollar, B. Burchfiel, R. Tedrake, D. Sadigh, S. Levine, P. Liang, and C. Finn, "Openvla: An open-source vision-language-action model," in *Proc. of the Conf. on Robot Learning*. PMLR, 2025, pp. 2679–2713.

[20] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "DINOv2: Learning robust visual features without supervision," *Transactions on Machine Learning Research (TMLR)*, 2024.

[21] J. Frey, M. Mattamala, N. Chebrolu, C. Cadena, M. Fallon, and M. Hutter, "Fast traversability estimation for wild visual navigation," in *Proc. of the Robotics: Science and Systems (RSS)*, vol. 19. Robotics: Science and Systems, 2023.

[22] S. Ægidius, D. Hadjivelichkov, J. Jiao, J. Embley-Riches, and D. Kanoulas, "Watch your STEPP: Semantic traversability estimation using pose projected features," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2025, pp. 2376–2382.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. of Intl. Conf. on Learning Representations (ICLR)*, Y. Bengio and Y. LeCun, Eds., 2015.

[24] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "MixMatch: A holistic approach to semi-supervised learning," *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.

[25] J. Cuaran, A. E. Baquero Velasquez, M. Valverde Gasparino, N. K. Uppalapati, A. N. Sivakumar, J. Wasserman, M. Huzaifa, S. Adve, and G. Chowdhary, "Under-canopy dataset for advancing simultaneous localization and mapping in agricultural robotics," *The International Journal of Robotics Research*, vol. 43, no. 6, pp. 739–749, 2024.