

# SemP-NBV: Semantic-Aware Predictive Next-Best-View for Autonomous Plant 3D Reconstruction

Xingjian Li<sup>1,2,3</sup>   Weilong He<sup>1,3,4</sup>   Jeremy Park<sup>1,5</sup>   S. Chris Reberg-Horton<sup>1,6</sup>  
 Steven Mirsky<sup>7</sup>   Edgar Lobaton<sup>1,2</sup>   Lirong Xiang<sup>\*1,3,4</sup>

**Abstract**—Three-dimensional (3D) Plant Phenotyping enables comprehensive trait analysis for evaluating plant growth in precision agriculture. Current phenotyping frameworks are low-throughput due to frequent manual intervention and inefficiencies in handling large volumes of repetitive data. Existing view planners for phenotyping are limited to static sampling methods or narrow focus on specific plant organs, restricting their utility in capturing the full complexity of plant structures. To address these limitations, we propose SemP-NBV, a novel semantic-aware predictive next-best-view approach for sample-efficient 3D plant phenotyping. Evaluated in a photorealistic simulator with 12 plant categories, SemP-NBV achieves 15.3% more observed points than an even-space sampler at 8 images on average, while matching the reconstruction quality at 20 even-space images. We demonstrate that existing state-of-the-art predictive planners designed for artificial structures struggle with zero performance increase compared to even-space sampler for complex plant structures. Furthermore, our approach generates semantic information during reconstruction, reducing the need for post hoc semantic labeling, and streamlining the 3D phenotyping workflow. The project is available at <https://github.com/ARLabXiang/SemP-NBV>.

## I. INTRODUCTION

Precision agriculture involves precise crop management and autonomous operations in data collection, analysis, and decision-making. The sensor and data analysis of individual plants is crucial in plant phenotyping, including the assessment of morphological (plant height, width, leaf length, etc.) and physiological (e.g., canopy temperature) plant traits related to growth, yield, and quality. Traditionally, top-view 2D imaging systems were favored for phenotyping due to high throughput in data collection, with depth sensors for 3D information of plant height and biomass (Fig. 1). However, these systems cannot fully capture plant 3D structures for accurate 3D trait analysis due to occlusions in plant canopies.

Recent advances in 3D sensing via laser scanner, multi-view stereo (MVS), and structure-from-motion (SfM) enable a detailed and accurate reconstruction of plant structure with details for individual leaves and stems that are typically not visible from 2D systems [2]. These developments also led to 3D point cloud datasets with plant organ semantics and temporal point cloud registration [3], [4]. However, there is either no autonomy or a limited autonomy workspace

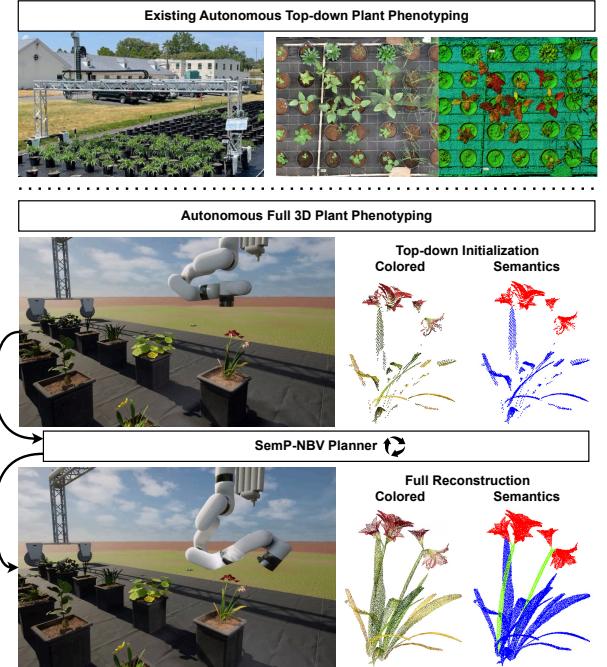


Fig. 1: Autonomous phenotyping frameworks and phenotype samples. Top: Real-world top-down phenotyping and depth visualization [1]. Bottom: Full 3D phenotyping with RGB and semantic point clouds in simulation.

for these 3D workflows. The current frameworks perform 3D reconstruction first and manual semantic labeling after, further slowing down the 3D phenotyping process. Our autonomous framework assigns 3D semantic information through 2D segmentation images to reduce manual labeling efforts.

Traditional autonomous 3D phenotyping frameworks generally follow a predetermined path and sample at many locations to ensure full reconstruction [2], [4]. This strategy is sample inefficient, which causes additional file storage and processing time, which should be minimized for higher throughput in 3D phenotyping. Next-best-view (NBV) planning improves sampling efficiency by prioritizing viewpoints with the most information gain. In recent years, NBV planners [5], [6] in artificial objects have incorporated deep learning, outperforming traditional Octomap and ray-cast strategies. However, these NBV planners show limited performance in 3D phenotyping as they are designed without considering the unique challenges in agricultural scenarios.

\* Corresponding author: lxiang@cornell.edu

<sup>1</sup> Plant Science Initiative, North Carolina State University <sup>2</sup>Department of Electrical and Computer Engineering, NCSU, <sup>3</sup>Department of Biological and Environmental Engineering, Cornell, <sup>4</sup>Department of Biological and Agricultural Engineering, NCSU, <sup>5</sup>Department of Computer Science, NCSU, <sup>6</sup>Department of Crop and Soil Sciences, NCSU, <sup>7</sup>USDA-ARS

Reconstructing 3D plants presents challenges that have not been specifically addressed by previous approaches. These challenges include: (1) homogeneous texture among plant leaves; (2) similar coloration between plant stems, leaves, and other organs; (3) difficulty in reconstructing thin structures, such as stems and petioles; (4) severe occlusion due to dense plant canopies; and (5) potential motion blur caused by wind conditions and robot vibrations. To mitigate these challenges, we introduce a semantic predictive (SemP) NBV planner, an advanced predictive planner inspired by Pred-NBV [5] that is applicable to plant structures while also operating on the semantics of plant organs. The framework can be extended to 3D reconstruction in other domains where similar challenges are present, such as industrial inspection and forestry monitoring.

Our work makes the following novel contributions:

- We present an advanced predictive next-best-view approach for applicable potted plant reconstruction with diverse features, where the existing approach would fail without gradient descent.
- We developed a semantic-aware predictive next-best-view (SemP-NBV) pipeline for further performance gain in plant reconstruction on top of the predictive-only approach.
- We developed a realistic field simulation sandbox for simulating the first large autonomous 3D phenotyping robotic systems involving a variety of plants.

## II. RELATED WORK

### A. 3D Plant Phenotyping

Most 3D plant phenotyping systems are stationary where individual plant pots are transferred to an imaging station for scanning. Laser scanning and MVS are commonly adopted reconstruction approaches in existing 3D plant reconstruction [2]. For example, Pheno-4D [7] mounted a laser scan system on a measuring arm for 3D phenotyping of early tomato and maize plants. [8] developed a mobile MVS imaging system for field image acquisition of maize plants, however, this system weighs over 120 kg and involves extensive human labor for plant transplantation and system reassembly. In addition to hardware restrictions, the aforementioned phenotyping frameworks also perform reconstruction before segmentation, requiring manual labeling or a variety of 3D point cloud segmentation algorithms [2] only suited for specific plants.

Neural Radiance Field and 3D Gaussian Splatting were adopted in 3D plant reconstruction [9]–[11] with the benefit of low-cost sensor hardware via SfM and the additional photorealistic reconstructions. However, the details of the plant structure are noisy and inconsistent in outdoor environments compared to a laser scanning system [9], [10]. Despite photorealistic appearances, the imprecise depth derived from simple SfM systems may limit the analysis of plant traits compared to a well-defined laser or MVS system.

### B. Next-Best-View

Next-best-view (NBV) planning attempts to maximize the information gain in 3D reconstruction. This planner



Fig. 2: Top-down view of plant assets. Top two rows: Test set models, ordered from largest point cloud (top-left) to least (bottom-right). Bottom two rows: 12 out of 76 training set models. In order are Boxwood (B), Bigleaf Hydrangea (BH), Indian Cress (IC), Amaryllis (A), Common Fern (CF), Coltsfoot (C), Funnel Weed (FW), Boston Fern (BF), Paddle Plant (PP), Bog Marshcress (BM), Gazania (G), and Taro (T).

generates the next viewpoint based on current observations. Existing NBV frameworks study man-made objects for efficient inspection tasks applicable to vehicles or buildings. Early work involves selecting the frontier of the observed voxel grid by casting rays from view candidates [12], [13]. Deep learning approaches utilized point cloud completion networks by predicting the complete point clouds using partial point clouds [5], [14]. A more generalized NBV approach utilizes reinforcement learning to remove the sampling of candidate views [6].

Plants have significant structure differences and increased geometry complexity compared to artificial structures studied in general NBV research. Few research studies studied NBV for full plant reconstruction, where [15] utilized voxel clustering and raycasting to determine missing regions for NBV for a SfM setup involving a robot arm and a turntable. This technique also relied on a static set of view candidates and 30 predetermined images for initialization, which is inefficient compared to stereo. Other NBV applications in agriculture focused on fruit or plant organ mapping instead of full reconstruction. Semantic NBV [16] used semantic voxels for multiclass mapping of tomato, peduncle, and petiole. NBV-SC [17] proposed shape completion methods for sweet pepper fruits by fitting partial peppers with superellipsoids to estimate the missing point clouds. Deep reinforcement learning [18] was also explored for faster sweet pepper mapping compared to the frontier voxel method [19]. In general, existing agricultural NBV relies on traditional and more limited methods, involving static viewpoint constraints and only to specific plant or plant organs. SemP-NBV addresses these limitations by providing a more generalized NBV planner for plant phenotyping and optimizing for full plant reconstructions while maintaining plant organ semantics across a diverse set of plant models.

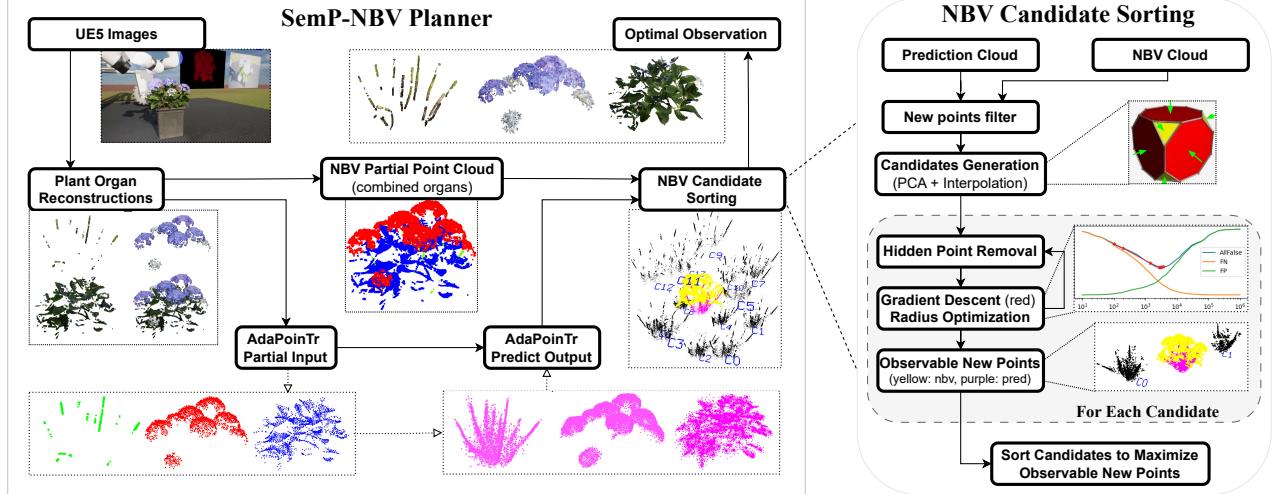


Fig. 3: SemP-NBV flowchart with corresponding visualizations. Note: The yellow point cloud represents the NBV partial point cloud. The green arrows in Candidates Generation illustrate the candidate set  $C$ .

### III. METHODS

#### A. Simulation

We constructed a simulation environment (Fig. 1) in Unreal Engine 5 with Benchbot, a customized mobile gantry platform using an Amiga robot, with an xArm6 robotic manipulator attached for local workspace adjustments. We simulate a stereo camera setup attached to the xArm6 end effector. The cameras are modeled with Raspberry Pi v3 Wide camera parameters with a baseline of 3.81 cm. A short baseline was used to capture nearby plants. Through simulation, we can obtain the depth and semantic ground truths using the framework detailed in our prior work [20]. The entire robot system operates on a black tarp ground with a grid of potted plants in an outdoor scenario. The Benchbot gantry in the simulation and the real-world has a width of 6.6 m across the base, allowing 7 potted plants per row underneath with a 60 cm gap in between.

For plant assets, we selected 12 plant categories with 88 realistic 3D models from Quixel Megascans [21] encompassing a variety of plant structures as shown in Fig. 2. During scene generation, these plant models are randomly sampled and placed with random rotations and positions above the pot grid. The framework will automatically assign panoptic labels to these plants per instances and semantics (stem, leaf, and flower) for each plant category.

The simulation is controllable through ROS Integration [22]. We used ROS2 Humble to control the base and joint positions of the two robots in UE5 to collect photorealistic synthetic images and ground truths for simulation. This pipeline also controls other simulation parameters, such as camera lights, grid dimensions, and time-of-day.

#### B. Synthetic Dataset

Using this simulation framework, we generated the synthetic point cloud dataset to train the AdaPoinTr [23] point cloud completion model. To emulate realistic observations

from camera sensors, we generated the partial point clouds by controlling the robot camera in a randomized pattern over a 40 cm radius semisphere around the plant. The camera orientations are also randomized to aim at a random position in a 10-cm cube around the plant. The complete point cloud will be a concatenation of the partial point clouds.

Specifically, we sampled across 96 seeded environments, each with a  $6 \times 7$  grid with a 60 cm gap between pots, where half of the environments had a single semantic for the entire plant and the other half were further segmented into the three plant organs. We then processed this dataset into a suitable format for training the AdaPoinTr network. For complete point clouds, we applied voxel downsampling, farthest point downsampling, and linear interpolation upsampling to standardize them to 16,384 points. Partial point clouds were downsampled to 2,000 points if they exceeded 2,000 points, and we discarded any with less than 200 points. All point clouds were then translated to a random position within a 10 cm box centered at the origin. The final dataset consisted of 15,349 pairs, each containing a complete point cloud with eight corresponding partial point clouds.

#### C. NBV Planner

Our approach draws inspiration from Pred-NBV [5], leveraging a point completion network combined with hidden point removal (HPR) [24] to efficiently estimate the information gain for each view candidate. We employed AdaPoinTr [23] to predict the complete point cloud  $P_p$  from partial point cloud  $P_i$ . The predicted new points  $P_n$  were obtained by filtering points that were closer to the input  $P_i$  than the predicted points  $P_p$ :

$$P_n = \{p \in P_p \mid \text{Nearest}(p, P_p) < \text{Nearest}(p, P_i)\}$$

Instead of the layered concentric circles parallel to the ground proposed in Pred-NBV, we applied principal component analysis (PCA) to the predicted new points  $P_n$ , generating

six viewpoint candidates from the three eigenvectors to form orthogonal view candidates. This approach also maximizes the variance of the 2D projection of  $P_n$  at these viewpoints, relaxing the constraints in view candidate sampling. We further expanded the set of candidates to 14 by interpolating the eigenvectors, accounting for potential occlusions caused by plant structures. These candidates are positioned 40 cm away from the  $P_p$  center.

The view candidates were then sorted to optimal order based on the estimated number of observable expected new points  $P_o$ . Similar to Pred-NBV, we applied the HPR operator to estimate the new observable points  $P_o$  from each candidate. However, we also introduced gradient descent to optimize the radius parameter of HPR to minimize the number of false estimates. The false estimation is defined by the intersections (false positives) and unobserved points (false negatives) of the two view candidates along the axis at different radii, as illustrated in Fig. 3. While this increases planning time, our experiments demonstrate that optimizing the radius parameter is critical for effective view selection. The optimal view candidates  $C^*$  were sent to the robot trajectory planner to find the feasible optimal candidate under the constraint of the workspace. This optimal view candidate will be the next-best-view for the reconstruction process.

In the context of SemP-NBV in Fig. 3, we apply the above steps to each plant organ segment  $P_{ij}$ , with  $j$  denoting the plant stem, flower, and leaf segment. After processing each segment, we choose the best  $C^{*j}$  based on proportion of new points denoted as  $\arg \max_j (P_{nj}/P_{pj})$ . This strategy prioritizes the least reconstructed segment.

## IV. EXPERIMENTS

### A. AdaPoinTr Training

We trained the AdaPoinTr model on multiple variations of our synthetic dataset with a 70-15-15 train-val-test split. We experiment with 5 dataset configurations across the whole plant prediction, segmented plant prediction, and both combined. Specifically, the dataset contained 7,516 whole plant pairs, 1,949 stem pairs, 4,940 leaf pairs, and 944 flower pairs. We used the PCN pretrained model and trained 600 epochs using default hyperparameters. We select the best model on the validation set evaluated at 5 epoch intervals.

### B. Stereo Matching Training

In order to obtain point clouds in the real world with quality comparable to those from simulation environments and thus enable the deployment of our NBV algorithm, we generated a synthetic stereo dataset corresponding to 15,349 pairs of point clouds for training the state-of-the-art stereo matching algorithm Selective-IGEV [25] from their SceneFlow pretrained model. This dataset comprises 15,349 disparity pairs, which we randomly divided into 90% (13,814 pairs) for training and 10% (1,535 pairs) for testing. The training protocol consisted of 6K iterations with a batch size of 40, employing a one cycle learning rate schedule ( $3 \times 10^{-4}$ ) and a weight decay ( $1 \times 10^{-5}$ ). Quantitative evaluation on the test set was performed using the Bad 0.5,

Bad 2.0, Bad 4.0, and Bad 5.0 metrics, as well as the End-Point Error (EPE) [25].

### C. NBV Experiments

The only prior work that addressed full plant reconstruction with a view planner was designed for an SfM system, where 30 images were required for initialization alone [15], significantly more than our planner designed for a stereo plant 3D phenotyping, we evaluate our NBV pipeline against the baseline approach of Pred-NBV, even-space sampling, and their variations. The testing environment uses 12 unseen plant models from the same categories. The base NBV configuration, denoted as WholeNBV, uses 8 image captures per plant without plant organ segmentation and applies a 10-step gradient descent optimization. We evaluate the reconstruction quality on two levels of detail using 2 mm and 5 mm voxel resolutions. All simulations were conducted on a workstation with a Ryzen Threadripper PRO 5945WX, RTX A6000 GPU, and 64GB of RAM. The UE5 simulation ran on Windows 11, with the ROS and planners in WSL2 Ubuntu 22.04.

1) *Baseline comparison:* For the even-space sampler baseline, we use a semi-spherical grid defined by radius  $r = 40$  cm, polar angle  $\frac{\pi}{20} \leq \phi \leq \frac{9\pi}{20}$ , and azimuthal angle  $0 \leq \theta \leq 2\pi$ . The grid comprises 5 rows in  $\phi$  and 16 columns in  $\theta$ . Using this configuration, we sampled 525 plants at every 11 steps to capture in a circular pattern with height variations. We also compared different step configurations by sampling every  $n \in \{2, 4, 6, \dots, 38\}$  step, which varied both the location and the number of images per plant (ranging from 3 to 40 images). Note the stepping iterates through  $\phi$  first and then  $\theta$ , e.g.  $n = 2$  produces a checkerboard pattern.

2) *Gradient Descent for HPR:* Gradient descent optimization for the radius parameter in HPR is a time-consuming process for our planner. We evaluated the importance of this process by experimenting with the reconstruction quality using one-step radius parameters and a 10-step gradient descent (WholeNBV). The constant radius parameters are 100, 1,000, and 10,000, chosen based on Fig. 3 for a mix of false negatives and false positives. Since WholeNBV without gradient descent is similar to Pred-NBV, we refer to these baseline variations as Pred-NBV-r100, 1,000, and 10,000. For additional reference, radius 100 was used in the original Pred-NBV implementation [5].

3) *SemP-NBV:* In this experiment, the SemP-NBV planner computes each plant organ and selects the NBV with the highest information gain based on the predicted new point cloud  $P_n$  and current reconstructions  $P_i$  of different plant organs. We also experiment by forcing specific plant parts denoted as StemNBV, FlowerNBV, and LeafNBV. In these experiments, the planner will revert to SemP-NBV if no corresponding plant part is found. For example, only four plants have flowers, the remaining plants will fall back to SemP-NBV under FlowerNBV.

TABLE I: AdaPoinTr test set results.

Class	Combined Training			Individual Training		
	F-Score	CD- $l_1$	CD- $l_2$	F-Score	CD- $l_1$	CD- $l_2$
Whole	0.965	2.568	0.041	0.964	2.632	0.042
Leaf	0.975	2.243	0.033	0.973	2.286	0.034
Stem	0.983	1.379	0.018	0.985	1.297	0.016
Flower	0.987	1.492	0.021	0.985	1.600	0.026

TABLE II: Stereo matching evaluation on different train set.

Train set	Bad 0.5	Bad 2.0	Bad 4.0	Bad 5.0	EPE (px)
KITTI 2012	64.91	16.88	14.53	13.79	10.71
KITTI 2015	64.61	16.54	14.21	13.42	8.35
Middlebury	<b>50.58</b>	<b>8.77</b>	<b>6.16</b>	<b>5.29</b>	<b>2.48</b>
Scene Flow	51.28	8.86	6.77	6.05	3.37
ETH3D	52.59	9.59	6.28	5.44	<b>2.34</b>
Synthetic (Ours)	<b>17.47</b>	<b>6.84</b>	<b>4.82</b>	<b>4.13</b>	<b>1.10</b>

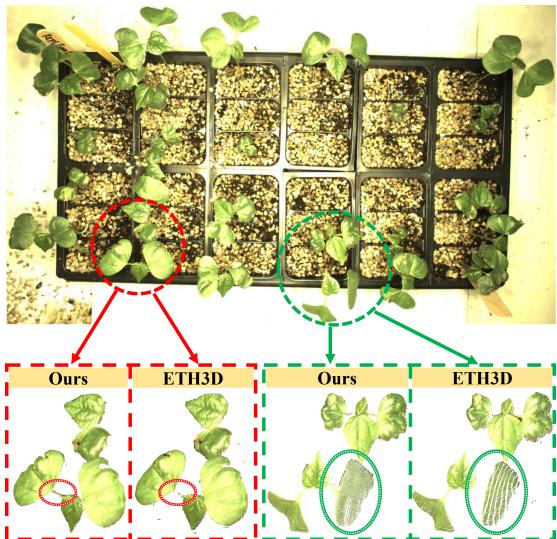


Fig. 4: Real plant 3D point cloud generation. Comparison of point clouds generated by the stereo matching model trained on our synthetic dataset (green circle) and the model trained on the ETH3D dataset (red circle). Our method is capable of reconstructing complete, delicate stems (left) and generating smoother point clouds (right).

## V. RESULTS

### A. AdaPoinTr

Table I shows AdaPoinTr results on the synthetic point cloud test set. Overall, combining the classes to train one model performed similarly compared to individually trained models. In particular, the stem and flowers have better predictions in F-Score and chamfer distances (CD) compared to the leaf or whole plant predictions. Indicating the model performs better in more structured geometries. We only utilized the combined training model in our NBV experiments.

### B. Stereo Matching Evaluation

Table II presents the evaluation results of the Selective-IGEV model trained on our synthetic dataset alongside those trained on public datasets including ETH3D, Middlebury, Scene Flow, KITTI 2015, and KITTI 2012. Compared to

TABLE III: Average observed points for all plants by even space sampling and WholeNBV.

Plant Abbr.	Full Name	EvenSpace Points	WholeNBV Points	Improvement
A	Amaryllis	45280.8	<b>48509.3</b>	7.13%
B	Boxwood	108210.7	<b>136680.9</b>	26.31%
BF	Boston Fern	17026.1	<b>18194.1</b>	6.86%
BH	Bigleaf Hydrangea	99965.5	<b>110301.9</b>	10.34%
BM	Bog Marshcress	11395.7	<b>12764.3</b>	12.01%
C	Coltsfoot	19846.5	<b>21166.3</b>	6.65%
CF	Common Fern	43193.0	<b>48976.5</b>	13.39%
FW	Funnel Weed	17240.5	<b>18493.9</b>	7.27%
G	Gazania	9916.3	<b>10604.5</b>	6.94%
IC	Indian Cress	58413.2	<b>67344.6</b>	15.29%
PP	Paddle Plant	16269.5	<b>18384.5</b>	13.00%
T	Taro	7598.0	<b>8954.2</b>	17.85%

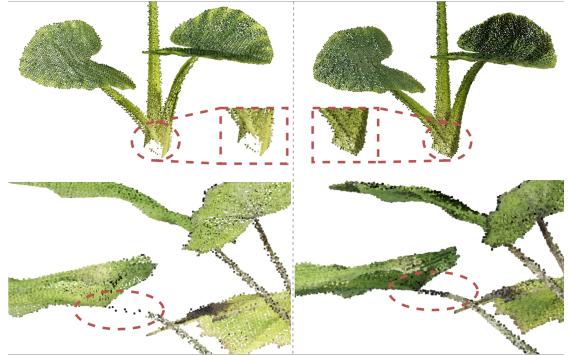


Fig. 5: Qualitative reconstruction comparisons between even space sampling (left) and WholeNBV (right).

training on public datasets, our dataset shows significant improvements in disparity estimation for plant structures and environmental backgrounds. This enables the stereo matching model trained on our synthetic dataset to generate real plant 3D point clouds of superior quality (Fig. 4). The point clouds produced by our approach contain more detail and exhibit enhanced smoothness, which is particularly crucial for the effective deployment of NBV algorithms in real-world agricultural applications.

### C. NBV Evaluation

Overall, the planning time for WholeNBV varies depending on the input plant point cloud sizes. Generally, the planner ranges between 0.5 and 4 seconds from AdaPoinTr prediction to sorted view candidates.

1) *Baseline*: Fig. 6 shows the complete plant reconstruction performance evaluated at two voxel resolutions: 2 mm (small features) and 5 mm (large features). Our planner achieves comparable coverage to 14 even-space samples for large features (4.3% improvement) and 20 even-space samples for small features (11.4% improvement) using only 8 images. Across all plant categories, the WholeNBV planner outperforms even-space sampling in both feature scales. We also show that our planner is more efficient in observing new points, surpassing even-space sampling by 19.3% on average across iterations 2–8. This efficiency metric may be higher due to per iteration normalization rather than the cumulative reconstruction. Notably, WholeNBV consistently observes

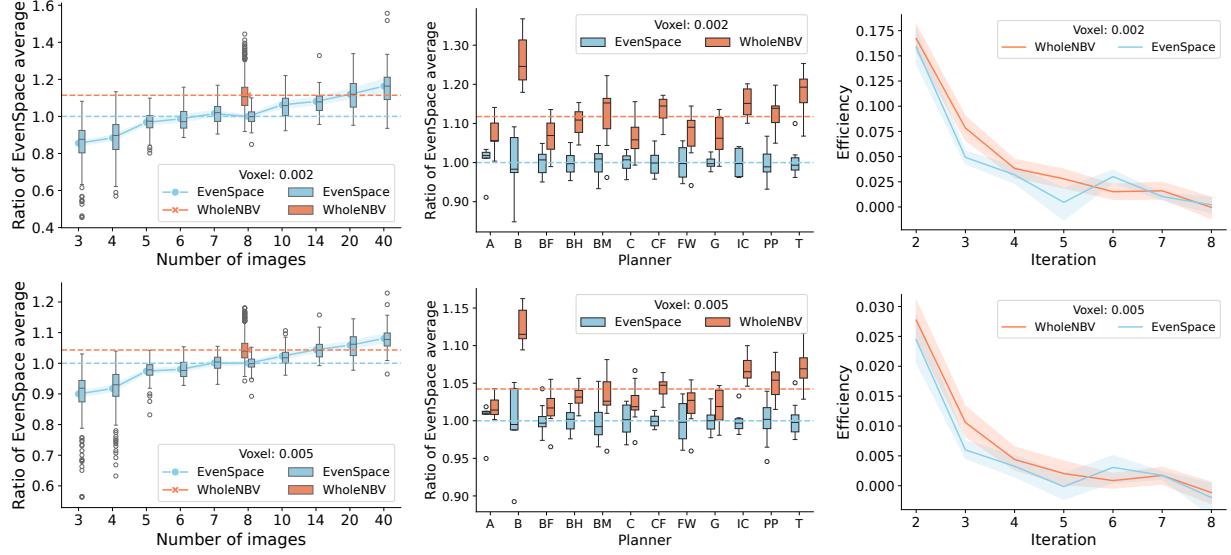


Fig. 6: Quantitative comparisons to baseline for the whole plant reconstruction. Top row: Smaller plant features at 2 mm voxels. Bottom row: Larger plant features at 5 mm voxels. Left: Reconstructions compared to the 8-image even-space sampling. The blue curve indicates the average and shaded area for 95% confidence intervals. Horizontal lines indicate the average for the NBV and 8-image even-space planner. Middle: Reconstruction for individual plant categories for 8-image even-space and NBV planner. Right: Efficiency is the ratio of observed plant pixels being new points for each image iteration, starting at the second iteration. Note: The efficiency metric may be negative due to voxelization combining observed points.

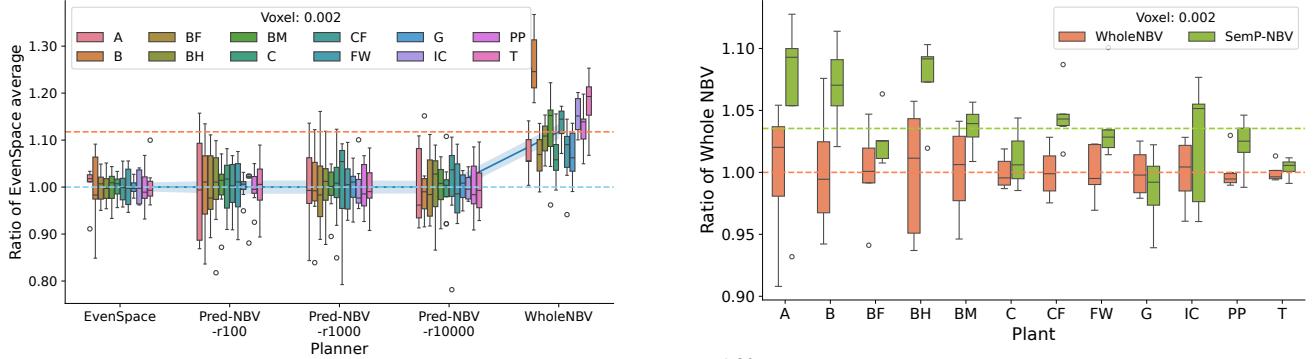


Fig. 7: Reconstruction quality with and without gradient descent for optimizing radius ( $r$ ) parameter in HPR at two voxel resolutions. Pred-NBV [5] at three constant radius parameters, the original implementation adopted  $r = 100$ .

more new points until iteration 6. Given the time cost in robot movements and image captures, our planner demonstrates both sample and time efficiency compared to a predetermined even-space sampling. From Table III, WholeNBV achieves a 26.31% improvement for the Boxwood (B) plant, likely attributable to its intricate internal structure. Qualitative improvements in Fig. 5 highlight significant gaps (several centimeters) from the even-space sampler, which are effectively reconstructed by the WholeNBV planner.

2) *Gradient Descent for HPR*: Fig. 7 shows the impact of performing gradient descent on the WholeNBV variation. We observe that the variations of Pred-NBV planner performed only negligibly better than even-space at all three constant

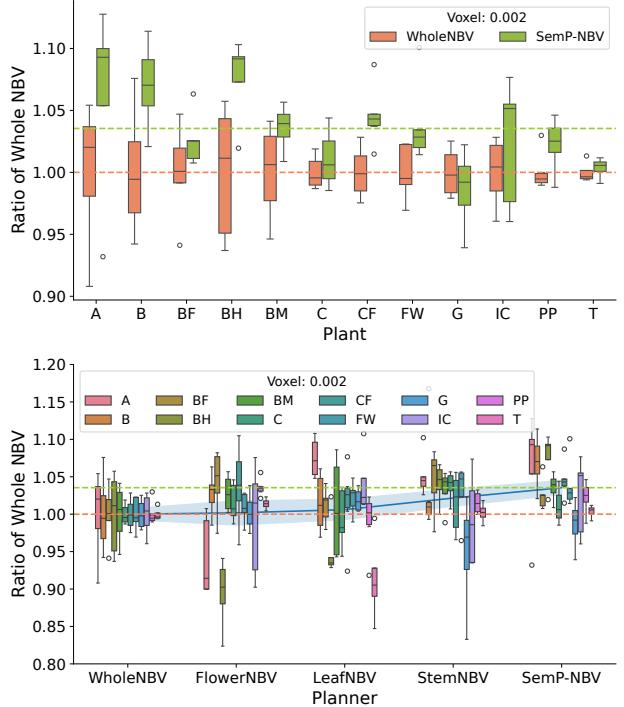


Fig. 8: Plant organ NBVs. Horizontal lines indicate the average performance of the WholeNBV and SemP-NBV.

radius parameters for HPR. This indicates that HPR parameter optimization is critical for the refinement of Pred-NBV for complex plant structures.

3) *SemP-NBV*: Fig. 8 shows the impact of semantics-aware planning in our framework. We observe that SemP-NBV further enhanced the WholeNBV planner by 3.5% on average in small feature reconstructions and even-space by 15.3%. However, no performance gains were observed in Coltsfoot (C), Gazania (G), and Taro (T) plants compared to WholeNBV, likely due to their simpler structure with sparse leaves and stems. When enforcing specific semantics in the FlowerNBV planner, we noticed the overall reconstruction performance decreases for three out of four applicable plants (A, BH, G, and IC). We attribute this to the limited spatial coverage of flowers in these plants, resulting in insufficient variety in view candidates for other plant organs.

4) *System Time*: Using the default joint limits in the xArm6 MoveIt configuration (maximum angular velocity of 2.14 rad/s), the average time to complete eight WholeNBV samples per plant is 145.7 seconds, and eight SemP-NBV take 177.1 seconds. In comparison, eight even-space samples require 100.2 seconds per plant, while 20 even-spaced samples take 234.2 seconds. The planner can be time-consuming for the same number of views, but it still demonstrates a reduction of 57.1 to 88.5 seconds compared to a 20 views even-space planner.

## VI. CONCLUSION

We proposed SemP-NBV, a novel semantic-aware predictive next-best-view approach for 3D plant phenotyping. A photorealistic UE5 simulation containing 12 different plant categories was also constructed to train and evaluate the planner. Our method enhances prior state-of-the-art predictive NBV planners by introducing PCA view candidate generation and gradient descent optimized HPR operator. Experimental results demonstrated that 8 images sampled by our planner achieve comparable coverage to 20 evenly spaced images, improving image efficiency by 19% and capturing 11.4% more observed points. Furthermore, we validated that our contributions are necessary for selecting the best view candidates in complex plant structures compared to the Pred-NBV planner. By incorporating plant organ information, the full SemP-NBV further increased reconstruction quality by 3.5% to an overall 15.3% compared to the baselines.

## REFERENCES

- [1] “Agricultural Image Repository.” [Online]. Available: <https://www.precisionustainableag.org/agimagerepo>
- [2] N. Harandi, B. Vandenberghe, J. Vankerschaver, S. Depuydt, and A. Van Messel, “How to make sense of 3D representations for plant phenotyping: a compendium of processing and analysis techniques,” *Plant Methods*, vol. 19, no. 1, p. 60, June 2023.
- [3] K. Mertoğlu, Y. Salk, S. K. Sarıkaya, K. Turgut, Y. Evrenesoğlu, H. Çevikalp, N. Gerek, H. Dutagaci, and D. Rousseau, “PLANesT-3D: A new annotated dataset for segmentation of 3D plant point clouds,” July 2024, arXiv:2407.21150 [cs].
- [4] S. Wu, W. Wen, Y. Wang, J. Fan, C. Wang, W. Gou, and X. Guo, “MVS-Pheno: A Portable and Low-Cost Phenotyping Platform for Maize Shoots Using Multiview Stereo 3D Reconstruction,” *Plant Phemonics*, vol. 2020, 2020.
- [5] H. Dhami, V. D. Sharma, and P. Tokekhar, “Pred-nbv: Prediction-guided next-best-view planning for 3d object reconstruction,” in 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2023, pp. 7149–7154.
- [6] X. Chen, Q. Li, T. Wang, T. Xue, and J. Pang, “Gennbv: Generalizable next-best-view policy for active 3d reconstruction,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16436–16445.
- [7] D. Schunck, F. Magistri, R. A. Rosu, A. Cornelissen, N. Chebrolu, S. Paulus, J. Léon, S. Behnke, C. Stachniss, H. Kuhlmann, and L. Klingbeil, “Pheno4D: A spatio-temporal dataset of maize and tomato plant point clouds for phenotyping and advanced plant analysis,” *PLOS ONE*, vol. 16, no. 8, pp. 1–18, Aug. 2021.
- [8] S. Wu, W. Wen, Y. Wang, J. Fan, C. Wang, W. Gou, and X. Guo, “Mvs-pheno: A portable and low-cost phenotyping platform for maize shoots using multiview stereo 3d reconstruction,” *Plant Phemonics*, vol. 2020, 2020.
- [9] M. A. Arshad, T. Jubery, J. Afful, A. Jignasu, A. Balu, B. Ganapathysubramanian, S. Sarkar, and A. Krishnamurthy, “Evaluating Neural Radiance Fields (NeRFs) for 3D Plant Geometry Reconstruction in Field Conditions,” Aug. 2024, arXiv:2402.10344 [cs] version: 3.
- [10] G. Tian, C. Chen, and H. Huang, “Comparative analysis of novel view synthesis and photogrammetry for 3d forest stand reconstruction and extraction of individual tree parameters,” 2024, arXiv:2410.05772 [cs].
- [11] T. Ojo, T. La, A. Morton, and I. Stavness, “Splanting: 3d plant capture with gaussian splatting,” in *SIGGRAPH Asia 2024 Technical Communications*, ser. SA ’24. New York, NY, USA: Association for Computing Machinery, 2024.
- [12] J. Daudelin and M. Campbell, “An adaptable, probabilistic, next-best view algorithm for reconstruction of unknown 3-d objects,” *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1540–1547, 2017.
- [13] S. Isler, R. Sabzevari, J. Delmerico, and D. Scaramuzza, “An information gain formulation for active volumetric 3d reconstruction,” in 2016 IEEE International Conference on Robotics and Automation (ICRA), 2016, pp. 3477–3484.
- [14] R. Zeng, W. Zhao, and Y.-J. Liu, “Pc-nbv: A point cloud based deep network for efficient next best view planning,” in 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020, pp. 7050–7057.
- [15] J. A. Gibbs, M. P. Pound, A. P. French, D. M. Wells, E. H. Murchie, and T. P. Pridmore, “Active vision and surface reconstruction for 3d plant shoot modelling,” *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, vol. 17, no. 6, p. 1907–1917, Dec. 2020.
- [16] A. K. Burusa, J. Scholten, X. Wang, D. Rapado-Rincón, E. J. v. Henten, and G. Kootstra, “Semantics-aware next-best-view planning for efficient search and detection of task-relevant plant parts,” *Biosystems Engineering*, vol. 248, pp. 1–14, 2024.
- [17] R. Menon, T. Zaenker, N. Dengler, and M. Bennewitz, “Nbv-sc: Next best view planning based on shape completion for fruit mapping and reconstruction,” in 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2023, pp. 4197–4203.
- [18] X. Zeng, T. Zaenker, and M. Bennewitz, “Deep reinforcement learning for next-best-view planning in agricultural applications,” in 2022 International Conference on Robotics and Automation (ICRA), 2022, pp. 2323–2329.
- [19] T. Zaenker, C. Smitt, C. McCool, and M. Bennewitz, “Viewpoint planning for fruit size and position estimation,” in 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2021, pp. 3271–3277.
- [20] X. Li, J. Park, C. Reberg-Horton, S. Mirsky, E. Lobaton, and L. Xiang, “Photorealistic arm robot simulation for 3d plant reconstruction and automatic annotation using unreal engine 5,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2024, pp. 5480–5488.
- [21] Quixel, “Quixel Megascans,” <https://quixel.com/megascans>, 2025, accessed: 2025-06-24.
- [22] P. Mania and M. Beetz, “A framework for self-training perceptual agents in simulated photorealistic environments,” in *International Conference on Robotics and Automation (ICRA)*, 2019.
- [23] X. Yu, Y. Rao, Z. Wang, J. Lu, and J. Zhou, “Adapointr: Diverse point cloud completion with adaptive geometry-aware transformers,” 2023, arXiv:2301.04545 [cs].
- [24] S. Katz, A. Tal, and R. Basri, “Direct visibility of point sets,” *ACM Trans. Graph.*, vol. 26, no. 3, p. 24–es, July 2007.
- [25] X. Wang, G. Xu, H. Jia, and X. Yang, “Selective-Stereo: Adaptive Frequency Information Selection for Stereo Matching,” in 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Seattle, WA, USA: IEEE, June 2024, pp. 19701–19710.