IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

PAg-NeRF: Towards fast and efficient end-to-end panoptic 3D representations for agricultural robotics

Claus Smitt†, Michael Halstead†, Patrick Zimmer†, Thomas Läbe‡, Esra Guclu†, Cyrill Stachniss‡, Chris McCool†

***Abstract*—Precise scene understanding is key for most robot monitoring and intervention tasks in agriculture. In this work we present PAg-NeRF which is a novel NeRF-based system that enables 3D panoptic scene understanding. Our representation is trained using an image sequence with noisy robot odometry poses and automatic panoptic predictions with inconsistent IDs between frames. Despite this noisy input, our system is able to output scene geometry, photo-realistic renders and 3D consistent panoptic representations with consistent instance IDs. We evaluate this novel system in a very challenging horticultural scenario and in doing so demonstrate an end-to-end trainable system that can make use of noisy robot poses rather than precise poses that have to be pre-calculated. Compared to a baseline approach the peak signal to noise ratio is improved from 21.34dB to 23.37dB while the panoptic quality improves from 56.65% to 70.08%. Furthermore, our approach is faster and can be tuned to improve inference time by more than a factor of 2 while being memory efficient with approximately 12 times fewer parameters. Code, data and interactive results are available at `https://claussmitt.com/pagnerf`**

I. Introduction

In recent years the agricultural sector has rapidly incorporated multiple robotic systems to perform monitoring and intervention tasks [1]–[7]. This is due to emerging needs of a more efficient and sustainable production, driven by factors such as climate change, scarcity of skilled labour, customer requirements, and increasing production costs. The successful adoption of robotic systems in agriculture is largely due to recent advancements in vision based deep learning (DL) [5], [6], [8]. In particular, the ability to perform vision based semantic and spatial reasoning in the robot's environment.

In horticulture, detecting [5], measuring size [4], estimating ripeness [6] and counting fruit [9] are some key monitoring tasks that provide growers detailed information to make better decisions, improving sustainability, and increasing production efficiency. Current state-of-the-art vision systems for detection and ripeness estimation use DL to perform instance-based semantic segmentation [4]. A set of recent work show that incorporating 3D information can greatly improve fruit counting and size estimation [4], [8]–[10].

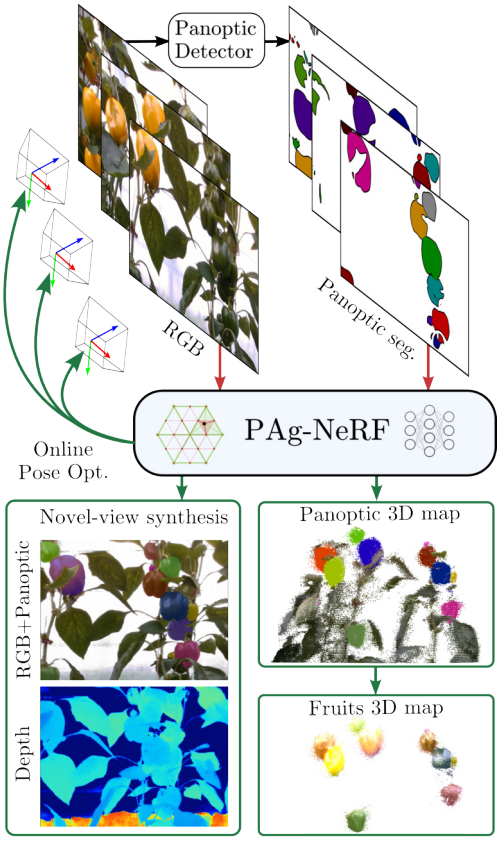

Fig. 1: PAg-NeRF is a fast and efficient model that renders novel-view photo-realistic images and ID consistent panoptic 3D maps from images, panoptic detections and noisy poses.

Recently, neural radiance fields (NeRF) have shown great potential to implicitly represent 3D information from just posed images. NeRF has been used to predict multiple properties of 3D scenes such as geometry [11], photo-realistic appearance [12], and more recently semantics [13] that are consistent across novel views. Furthermore, recent contributions have considerably improved their performance and memory efficiency [11], [14], [15], turning them into a promising representation for robotics applications.

In this work, we present a novel system **PAg-NeRF** (Fig.1) which performs online pose optimization [16] using state-of-the-art accelerated NeRFs [11] to produce 3D consistent panoptic representations. These representations enable us to resolve identity associations across image sequences using

Manuscript received: August 31, 2023; Revised: October 18, 2023; Accepted: November 22, 2023.

This paper was recommended for publication by Editor Hyungpil Moon upon evaluation of the Associate Editor and Reviewers comments.

Authors † are with Institute of Agriculture, Agriculture Robotics & engineering and ‡ are with the Institute of Photogrametry of the University of Bonn, Germany [csmitt, patrick.zimmer, michael.halstead, egueclue, cmccool]@uni-bonn.de, laebe@ipb.uni-bonn.de, cyrill.stachniss@igg.uni-bonn.de

This work was partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) FOR 5351 – 459376902 (AID4Crops) and under Germany's Excellence Strategy - EXC 2070 – 390732324 (Phenorob).

Digital Object Identifier (DOI): see top of this page.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

only the implicit geometry of NeRF. We deploy this novel system in a very challenging horticultural scenario and in doing so make the following contributions, we:

- propose a novel panoptic delta grid architecture that outperforms previous state-of-the-art [17] and is ∼2x faster with ∼12x fewer parameters;
- propose a modified instance assignment loss targeted for planar side-facing camera motions which improves panoptic quality (PQ) by 1.56 points;
- and present an end-to-end trainable NeRF-based panoptic 3D representation targeted for agriculture.

II. Related Work

A diverse range of techniques have been employed in agriculture to provide robotic systems with geometric and semantic scene understanding. More recently, NeRF approaches have also been used to enhance monitoring systems in the agriculture sector. Below we provide a brief review of the relevant monitoring systems and recent advancements in fast and efficient NeRFs that allowed the development of this work.

A. Crop Monitoring

Crop monitoring is an important component of any agricultural robotic platform, from arable farmland to glasshouses. Without accurate monitoring, these platforms would be unable to accurately perform intervention activities such as weeding [1], harvesting [3], or yield estimation [4].

In recent years DL, in particular deep neural networks (DNNs), has dominated state-of-the-art techniques for agricultural monitoring. Sa *et al.* [5] showed how DNN object detectors could be fine-tuned for accurate sweet pepper detection. This was extended in [6] to include subclass-based (fruit ripeness) classification, integrated into a fruit tracking approach to count fruit. Turning grape detection into a three-class problem (background, grapes, edges) Zabawa *et al.* [7] improved the detection of grapes in an orchard by creating better distinction between the individual grapes. Once again using a top-down approach to instance-based semantic segmentation, [4] showed that crop monitoring could use a similar approach in both arable farmland and glasshouses. In general, the majority of these approaches are still-image based and do not consider spatial-temporal information. Smitt *et al.* [8] showed how integrating robot trajectory and 3D scene information into DNNs could improve state-of-the-art results. More recently [10] tracked fruit instance detections and mapped them into a multi-resolution occupancy grid. Finally, fruit 3D models predicted with a CNN were registered to the grid, generating a panoptic 3D map of the crops.

B. Panoptic Segmentation

Panoptic segmentation jointly solves the tasks of semantic and instance segmentation and combines them in a single prediction. A semantic label is predicted for each pixel of "stuff" segments, as well as an instance ID for "things" segments, yielding detailed and effective scene understanding [18]. In early works, semantic and instance outputs of CNNs were combined to obtain panoptic predictions [19], [20].

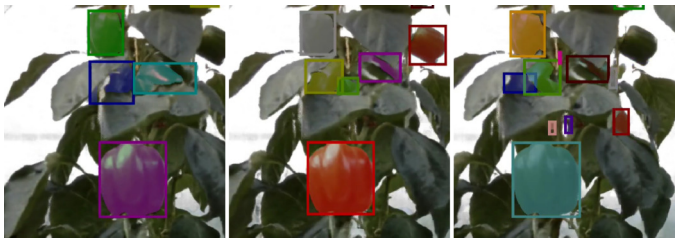

Fig. 2: Still-image panoptic detector producing inconsistent instance IDs and false positives in a sequence of frames

More recently, vision-based transformers have gained popularity with their strong performance [21] such as Mask2Former [22]. This model was proposed as a transformer-based universal segmentation model, which uses masked-attention to extract localized features. Another recent method by Jain *et al.* [23] applies task-guided queries to obtain mask predictions, achieving state-of-the-art performance.

These techniques are an obvious option for crop monitoring purposes. However, they are still-image detectors and do not produce consistent instance ID predictions between frames (as depicted in Fig. 2). The flickering instance ids can be tackled separately by solving instance tracking [4], [9], [10]. By contrast, in this work we use frame-based instance predictions to train a NeRF 3D scene representation which then implicitly ensures there is inter-frame ID consistency.

C. Neural Randiance Fields

Since their original introduction [12], NeRFs have become a very popular method to implicitly represent 3D scenes from posed views. Some applications include photo-realistic novel-view synthesis [12], localization and mapping [16], detailed 3D reconstruction [11] and semantic 3D mapping among others [17], [24], [25]. However, the original NeRF architecture is very memory and compute intensive since it is based on deep multi layer perceptrons (MLPs) and uses offline structure from motion (SFM) to compute camera poses.

1) Learning efficient geometric encodings: Seeking to improve training and inference speed of NeRFs [14] bookkeep an occupancy grid to sample only occupied space, maintaining however a large MLP network. Sun et al. [26] employed 3D grids with learnable features at their vertices and a very shallow MLP decoder on top. This allowed the gradients to only be propagated through the interpolated features when queried by ray-tracing. Müller *et al.* [15] embedded sparse hash tables in multi-resolution grids, with less parameters than keys. Hash collisions were resolved through the training process, yielding implicitly sparse grids to learn fast and memory efficient neural graphic models.

More recently, Rosu *et al.* [11] tackle fast estimation of high-detailed sign distance functions (SDF). They employed permutoedral hash-grids, instead of cubic ones, since they interpolate less points per queried 3D coordinate resulting in faster training and inference. This method also incorporates hash-grids making it memory efficient as well.

2) 3D semantic NeRFs: The continuous differentiable nature of NeRFs allows encoding of any continuous properties into 3D space. Recently Zhi *et al.* [13] presented an approach

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

to encode semantics into an MLP based NeRF with noisy 2D predictions of synthetic environments as input. This yielded cleaner and 3D consistent novel-view semantic predictions.

In the field of horticulture monitoring, Kelly *et al* [27] trained a MLP NeRF model with strawberries and sweet pepper images captured by field robots. Semantic detections were used to refine the render quality of fruits, and inspired by [16] they jointly optimize camera poses, using odometry as initial guess, avoiding the need for offline SFM. Later works encoded instance prediction from 2D images into NeRFs in the context of autonomous cars [24], [25], handling inter-frame ID inconsistencies with a separate 3D tracking phase. Very recently, a tracking free panoptic NeRF model was introduced [17], which matches instance ID detections and model outputs by solving a linear assignment problem optimally. This model uses a TensoRF [28] feature encoding grid and is evaluated in indoor environments.

All these panoptic approaches require off-line pose optimization to produce accurate results. To the best knowledge of the authors, there are no panoptic NeRF end-to-end models that jointly optimize poses and the 3D representation.

III. Proposed Approach

Our method generates implicit 3D representations of static environments with color, depth and panoptic segmentation modalities. We achieve this by training a NeRF model from color frames with automatic (per-frame) panoptic detections and noisy robot poses. In addition to a regular color grid, we propose a novel delta grid for panoptic decoding. The resultant model can then be used to perform novel-view synthesis with consistent label IDs for the instances present in a scene. We target our approach to challenging agricultural scenarios.

We represent 3D scenes as volumetric panoptic radiance fields. These map 3D points $\mathbf{p} \in \mathbb{R}^3$ and view directions $\mathbf{d} \in S^2$ to volumetric fields with density $\hat{\sigma} \in \mathbb{R}_{[0,\infty]}$, color $\hat{\mathbf{c}} \in \mathbb{R}^3_{[0,1]}$ and distributions of $\hat{\mathbf{s}}$ over D semantic classes as well as unique instance ID of objects $\hat{\mathbf{k}}$ over N instances. We approximate this continuous representation with an NN $\mathfrak{F}_\Theta : (\mathbf{p}, \mathbf{d}) \rightarrow (\hat{\sigma}, \hat{\mathbf{c}}, \hat{\mathbf{s}}, \hat{\mathbf{k}})$ by optimizing parameters Θ.

As shown in Fig. 3, in order to render color images, each point $\mathbf{p}$ sampled along each pixel ray is first encoded by a color grid into a feature vector $\mathbf{g}_c$. Then these are decoded into density features and the last element is interpreted as the scene volumetric density $\hat{\sigma}$. Since color is view dependent, we encode the view direction $\mathbf{d}$ and concatenate it to the density features before decoding them into the final color prediction $\hat{\mathbf{c}}$.

On the other hand, panoptic features $\mathbf{g}_p$ are obtained by correcting the appearance features with $\Delta\mathbf{g}_p$. Semantic $\hat{\mathbf{s}}$ and instance id $\hat{\mathbf{k}}$ magnitudes are then directly decoded by shallow NNs as they are view-independent magnitudes. Finally, we use the estimated $\hat{\sigma}$ to perform volumetric rendering of all magnitudes by ray-marching all sampled rays.

A. Learnable grid encodings

With the aim of reproducing fine-grain detail while maintaining low running time and memory footprint, we choose to use 3D multi-resolution permutoedral hash-encodings [11]. This encoding partitions 3D space into tetrahedral lattices and the queried grid features are interpolated between 4 values (instead of 8 for cubic grids) making them faster. Furthermore, we book-keep a 3D octree to further accelerate inference by ray-sampling coordinates in high-density voxels only. Similar to [11], we encode each queried ray sample by interpolating feature vectors at each grid resolution and concatenating them. Then, we decode them with shallow NNs to obtain the estimated scene properties as shown in Fig. 3.

1) Delta grid architecture: As shown by [17], despite the underlying scene geometry being the same, features required for panoptic decoding $(\hat{\mathbf{s}}, \hat{\mathbf{k}})$ might be slightly different than the ones needed for appearance decoding $(\hat{\mathbf{c}}, \hat{\sigma})$. Thus, similar to other works [11], [28], we choose to have a separate grid encoder for specific outputs, in our case for panoptic quantities. Our panoptic grid architecture (Fig. 3) leverages the similarity between modalities by computing panoptic grid features as $\mathbf{g}_p = \mathbf{g}_c + \Delta\mathbf{g}_p$. Where $\Delta\mathbf{g}_p$ is a feature vector output of the panoptic grid G_p. Thanks to the implicit sparseness of hash-grids, we are able to reduce the capacity G_p w.r.t. G_c, to only have valid values where corrections are needed to have a good panoptic representation. In addition, we avoid propagating gradients from the panoptic to the color branch to ensure G_p only learns corrections on top of G_c. See Sec. V-C for the corresponding ablations exploring this design choice.

B. Volumetric rendering

To predict a pixel color from a given camera with center of projection at $\mathbf{o} \in \mathbb{R}^3$ and direction to the pixel $\mathbf{d} \in S^2$, NeRF [12] leverages volumetric rendering [29], integrating the values of field $\mathfrak{F}$ along a ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$.

1) Appearance rendering: The observed color C depends on viewing direction $\mathbf{d}$, due to illumination and translucence phenomena and so the predicted color $\hat{C}$ can be computed as:

$$\hat{C}(\mathbf{r}, \mathbf{d}) = \int_{t_i}^{t_f} T(t)\hat{\sigma}(\mathbf{r}(t))\hat{\mathbf{c}}(\mathbf{r}(t), \mathbf{d}))dt \ , \tag{1}$$

$$T(t) = \exp\left(\int_{t_i}^{t} -\hat{\sigma}(\mathbf{r}(m))dm\right) \ , \tag{2}$$

where $T(t)$ represents the transmittance probability at t.

2) Panoptic rendering: Our model combines semantic and instance predictions at each rendered pixel to produce panoptic predictions. Unlike color, panoptic fields are independent of view direction, thus semantic and instance predictions can be expressed as continuous functions conditioned only on 3D points $\mathfrak{S}(\mathbf{p})$ and $\mathfrak{I}(\mathbf{p})$. In order to treat samples along rays as samples of a distribution, similar to [17], we apply softmax before ray integration. Using $\hat{X}$ as a notation proxy for panoptic predicted quantities and $\hat{\mathbf{x}}_{sm} = softmax(\hat{\mathbf{x}})$, $\hat{X}$ can be rendered with the following equation:

$$\hat{X}(\mathbf{r}) = \int_{t_i}^{t_f} T(t)\hat{\sigma}(\mathbf{r}(t))\hat{\mathbf{x}}_{sm}(\mathbf{r}(t))dt \ . \tag{3}$$

C. Training losses

1) Color loss: For $\mathcal{L}_{color}$ we minimize the average photometric loss $\|C_\mathbf{r} - \hat{C}_\mathbf{r}\|^2$ over a batch of rays $\mathbf{r} \in \mathcal{R}$, randomly sampled rays across camera frames.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

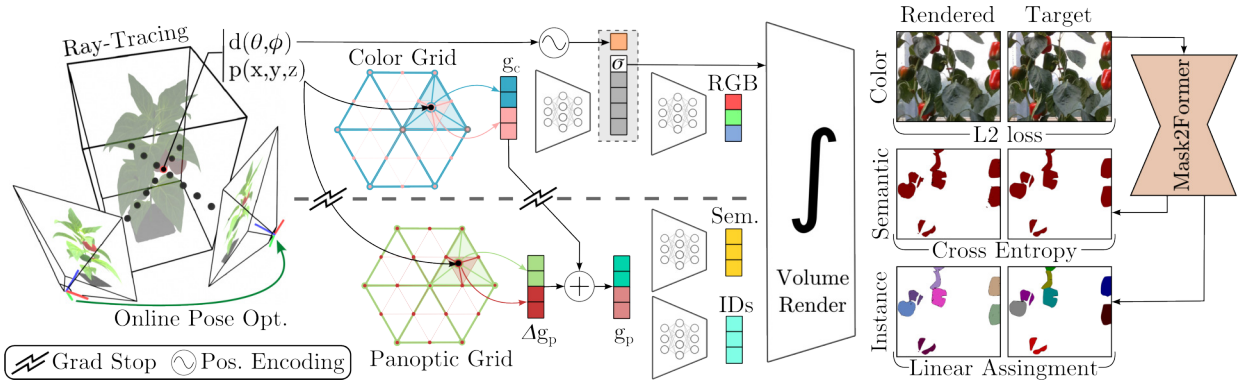

Fig. 3: Our 3D neural representation uses 2 separate grids to represent color and panoptic quantities. It learns the panoptic representation from automatic detections and the camera poses are jointly optimized with only the color loss.

2) Semantic loss: To predict semantic labels from frame-wise detections of a still-image segmentation model, as proposed by [13], we compute the cross entropy loss between the rendered semantic multi-variate distribution $\hat{S}_\mathbf{r}$ over D classes and the detected semantic class $S_\mathbf{r}$ for a batch of rays $\mathcal{R}$,

$$\mathcal{L}_{sem} = \frac{1}{|\mathcal{R}|} \sum_{\mathbf{r}\in\mathcal{R}} w_\mathbf{r} S_\mathbf{r} \log \hat{S}_\mathbf{r} \ , \tag{4}$$

where $w_\mathbf{r}$ is the confidence of the semantic detector [17].

3) Instance ID Linear assignment: To train our model on frame-wise panoptic segmentation predictions, we need to tackle the inter-frame inconsistency of "things" IDs. Similar to [17], for the i-th frame we perform an optimal linear assignment of sampled rays from the n-th predicted thing mask ID K^i_n to the m-th most similar rendered one $\hat{K}^i_m$. The assignment cost can be expressed as:

$$\mathcal{C}^i_{nm} = \frac{-1}{|\mathcal{R}^i_n|} \sum_{\mathbf{r}\in\mathcal{R}^i_n} \hat{K}^i_{\mathbf{r}m} \ , \tag{5}$$

We employ the Hungarian algorithm [30] to optimally solve the assignment, obtaining frame-wise pseudo-label vectors K'^i. This is then used to compute a frame-wise cross entropy loss to train our instance ID head

$$\mathcal{L}_{things} = \frac{1}{|\mathcal{I}|} \sum_{i\in\mathcal{I}} \frac{1}{|\mathcal{R}_i|} \sum_{\mathbf{r}\in\mathcal{R}_i} w_\mathbf{r} K'^i_\mathbf{r} \log \hat{K}^i_\mathbf{r} \ , \tag{6}$$

where $\mathcal{I}$ is a batch of training frames. We also minimize the following loss for all rays $\mathcal{R}_s$ corresponding to detected stuff classes over all images for their predicted ID to be 0.

$$\mathcal{L}_{stuff} = \frac{1}{|\mathcal{R}_s|} \sum_{\mathbf{r}\in\mathcal{R}_s} w_\mathbf{r} \mathbf{e}_1 \log \hat{K}^i_\mathbf{r} \ , \tag{7}$$

where $\mathbf{e}_1$ is the first canonical versor. This yields the following instance ID loss $\mathcal{L}_{id} = \mathcal{L}_{things} + \mathcal{L}_{stuff}$.

4) Repeated ID rejection: In the glasshouse scenario, target fruits are arranged in a quasi-planar fashion and the camera moves parallel to them [9]. This means that each fruit is always measured at the same distance from the image plane. Thus, targets at opposite ends of the camera frustum will always appear in several frames on their own, but only in a few frames together. This does not encourage the optimal assignment to assign different IDs to targets at the edge of images and can lead to multiple objects having the same ID. This can lead to poor panoptic detection results, see Panoptic Lifting in Fig. 6. This issue also arises in domains such as warehouse monitoring, factory inspection, etc. However, the linear assignment loss has only been employed in target-centric or free camera motions [17], where this phenomenon does not occur.

We address this issue with a simple sliding window of assignable IDs, linearly dependent of each target's 3D position along the robot trajectory. This gets incorporated to the optimal ID association by setting the assignment costs (eq. 5) of predicted IDs outside of the window to a prohibitively high value. Additionally, we design the window such that targets at a spatial distance close to the frustum length (e.g. at opposite ends of the camera frustrum) won't have overlapping assignable IDs. Finally, the fruit's position is computed by unprojecting their mask pixels with our model's predicted depth.

5) Post-processing and total loss: Inspired by [17], we apply a very weak segment consistency loss $\mathcal{L}_K$ to instances instead of semantic segments. A single erosion dilation stage with a 3x3 kernel is applied as post-processing step of the panoptic output. Finally, our total loss can be written as:

$$\mathcal{L} = \lambda_c \mathcal{L}_{color} + \lambda_s \mathcal{L}_{sem} + \lambda_{id} \mathcal{L}_{id} + \lambda_{reg} \mathcal{L}_K \ . \tag{8}$$

D. Camera extrinsics optimization

As we aim to represent 3D scenes from images with noisy robot odometry poses, these need to be refined to ensure proper multi-view consistency. Most NeRF approaches perform offline bundle adjustment on the data as a pre-processing step. This is reasonable when camera extrinisics are unknown. In our case we have good initial guesses from odometry which allows us to perform online optimization within our training process, similar to [16], [27]. To achieve this we add the camera pose parameters to the optimizer and propagate gradients through the color branch and the ray-tracing operation. Thus, rays $\mathbf{r}$ in eq. 1 are made dependent on their corresponding camera extrinsics $E \in$SE3

$$\mathbf{r}(t, E) = \mathbf{o}(E) + t\mathbf{d}(E) = \mathbf{t}_E + tR_E\mathbf{d}_c \ , \tag{9}$$

where $\mathbf{t}_E$ and R_E are the camera translation and rotation in the world coordinate frame respectively, and $\mathbf{d}_c$ is the direction of

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

a pixel in the camera coordinate frame. During optimization we use the 6DOF representation proposed by [31] which is suitable for NN optimization. We do not employ a coarse-to-fine modulation approach [16] as our robot odometry provides a close enough initial estimate of the poses.

IV. Experimental Setup

We evaluate our models in a very challenging agricultural dataset of commercial glasshouse sweet pepper cropping (BUP20) that we introduced previously in [9]. The dataset was captured by a robot driving at $0.2m/s$ in a glasshouse parallel to the crops and consists of 10 sequences with wheel odometry and RGB-D images recorded using Intel RealSense D435i of 6 crop rows. It presents two sweet pepper cultivar *Mavera* (yellow) and *Allrounder* (red), each cultivar matured from green, to mixed, to their primary color. Data was captured twice, two months apart, to show different growth stages. Moreover, the dataset presents a rich data variety including different illumination conditions, noisy odometry, high level of fruit occlusions, green on green detection scenarios and camera shake. Furthermore, the dataset comes with panoptic semantic segmentation labels. For this work, we also pre-computed panoptic predictions using an instance segmentation model [22] trained on this domain. The dataset has sparse non-overlapping instance segmentation annotations, and we generate short sequences around the labeled frames for training and validation. In order to assess panoptic segmentation performance in the plant row closest to the robot we filter out masks with depth larger than 1.5m, ignoring masks from further crop rows, similar to [9].

A. Implementation details

PyTorch is used to implement our models and we leverage the Kaolin-Wisp framework [32] which provides rendering infrastructure, state-of-the-art NeRF building blocks and allows for rapid-ptotyping. For the permutohedral grids, we integrate the implementation provided in [11] to Kaolin-Wisp. Both our color and panoptic grids have 21 LODs linearly spanning from $1m$ to $0.0001m$, vertex features of dimension 2 and a maximum capacity of 2^{18}. We encode ray directions with a regular positional encoder for color prediction. Our density and color decoders are single layer NNs of width 64, with ReLU and sigmoid activations respectively. Our density feature vector has width 16. The semantic and instance decoders are shallow and narrow NNs of 2 and 3 hidden layers respectively and both of width 64. Such small NNs are effective enough as most of the scene representation is already encoded in the feature grids.

1) Training scheme: Since we only have sparse panoptic labeled frames in each video sequence, we train our scene representations in windows of frames around each labeled frame. Every second frame in the window is used for training, and validation appearance metrics are computed in the remaining frames. For panoptic metrics, we only compute them for the middle labeled frame. Finally, we average all the windowed results to obtain our final performance metrics.

We train each scene window for 800 epochs, sampling 4096 rays per image with 512 samples along each for the first 200 epochs. After that we prune the occupancy grid every 200 epochs and change from ray-tracing to voxel-tracing, taking 2 samples at each of the first few occupied voxels along the rays. This way we refine the scene geometry close to the surface of objects [11], [12]. For the first 600 epochs we only train the color head, later adding the panoptic head for the remaining ones. We use Adam [33] as the training optimizer with a momentum of 0.9 and a fixed global learning rate of 0.01. Grid encodings have a learning rate of 1.0 in order for them to converge faster. For all extrinsic parameters we set their learning rate to 0.0001. From an early parameter search, we set the loss weights to $\lambda_{color} = 1,\ \lambda_{sem} = 0.1,\ \lambda_{id} = 10$ and $\lambda_{reg} = 0.1$ for all our experiments. We train with a batch size of 6 with images at full resolution (1280x720). All models for each validation sample were trained on a single A6000 GPU.

2) Validation extrinsics optimization: Since we jointly optimize camera extrinsics along with the scene representation, validation extrinsics can get slightly miss-aligned to the scene. Thus, every 10 epochs we optimize the camera extrinsics of all validation frames while freezing all grid and decoder parameters. This way the validation extrinsics get registered to the scene representation allowing for a direct comparison.

B. Evaluated models

In order to evaluate the novel-view rendering quality of the evaluated models, we employ peak signal to noise ratio (PSNR). The panoptic quality mentric (PQ) is used to measure the instance and semantic outputs. Semantic quality alone is also presented using the intersection over union (IoU).

We compare PAg-NeRF with similar state-of-the-art approaches that also employ different radiance fields at their core. Performance compared to the still-image panoptic segmentation model used to generate the training pseudo-labels is also provided; note that the pseudo-labels do not provide consistent IDs across frames.

Mask2Former [22]: the instance-based semantic segmentation model trained fully-supervised on BUP20.

Semantic NeRF [13]: the first approach incorporating a semantic head to an NN based NeRF model. We train this models using poses obtained through SFM. Results for this model are presented using only render quality (PSNR) and segmentation (IoU).

Panoptic Lifting [17]: is a state-of-the-art panoptic 3D scene representation, leveraging [28] as its feature grid. This models is trained with pre-calculated poses and panoptic predictions from Mask2Former.

PAg-NeRF(L): our large panoptic 3D scene representation model using the proposed panoptic grid architecture. All variants are trained with online optimized poses and panoptic predictions from Mask2Former. Three versions of this model are evaluated: a baseline one, another adding repeated ID rejection (sec. III-C4), and a third one using Mask2Former confidence to re-weight all panoptic losses.

PAg-NeRF(S): A faster version of our model with reduced number of parameters (see sec. V-C) that still outperforms the state-of-the-art.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

V. Results

We start by presenting three quantitative experiments. First we compare our best model with other semantic novel-view synthesis methods, and the NN still-image detector used for training. Second, we assess the effect of performing online pose optimization compared to other trajectory sources. Third, to better understand the benefits and limitations of our panoptic grid architecture we perform an ablation study. Finally, we showcase qualitative results of novel-view panoptic predictions of our model compared to other systems. When we present inference time, this is based on how long each model takes to render all outputs per image at full resolution.

A. Overall performance

In Tab. I we present the results of our model, Mask2Former, and other relevant NeRF models (SemanticNerf and PanopticLifting). It can be seen that the best results are obtained for our model when using uncertainty predictions, the proposed repeated ID rejection loss and pose optimization. In particular, our large version of PAg-NeRF outperforms panoptic lifting by 2dB in render quality and an absolute improvement of 13.43 and 6.37 for panoptic and semantics respectively. Moreover, our system is 1.34 times faster at training and 1.54 times at inference, despite having 3.4 times more parameters than Panoptic Lifting. This is due to the fast interpolation of permuto grids and shallow NN decoders. It can also be seen that our modified repeated ID rejections loss improves PQ by 1.56 while the use of uncertainty loss re-weighting gives an improvement of another 1.69.

It can also be seen that our large model outperforms semantic NeRF by an absolute IoU margin of 4.84. Furthermore, as our model is grid-based, we are 4.5 and 11 times faster at training and inference respectively. We attribute this performance gap to our online pose optimization scheme (see sec. V-B) and the dedicated panoptic delta feature grid to improve the panoptic representation (see sec. V-C).

It is also worth noting that PAg-NeRF achieves these results with very shallow NN decoders with only a few hidden layers of width 64, whereas both panoptic lifting and semantic NeRF use more and wider layers of width 256.

B. Pose estimation ablation

Our approach provides an end-to-end system that relies only on initial estimates of pose. To evaluate the impact of this approach we compare against offline optimized poses with a bundle-adjusted software, and odometry only in terms of render and panoptic quality. This is because the dataset we use lacks a precise ground-truth trajectory.

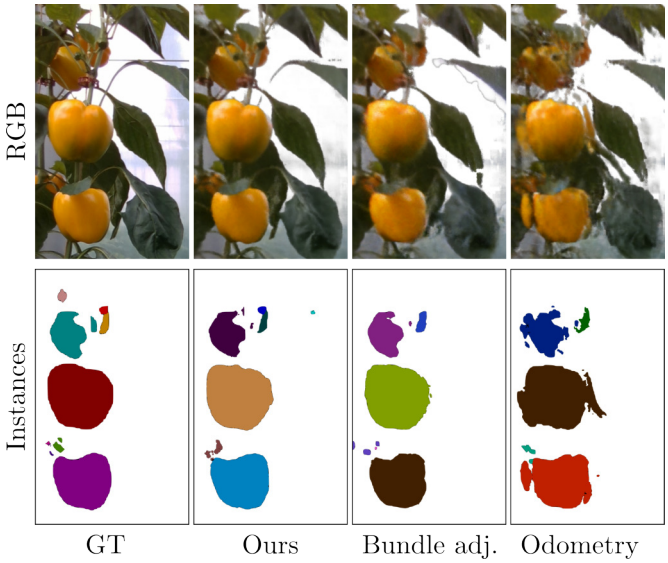

Fig. 4: Render comparison for different camera pose sources.

In Tab. II it can be seen that our online pose optimization method produces the best performance. Compared to bundle-adjusted poses we improve the absolute performance by 1.45dB, 8.54 and 5.21 for render quality, panoptic and semantic metrics respectively.

Qualitative results in Fig. 4 highlight the advantages of our system (Ours) over offline optimized poses with a bundle-adjusted software (Bundle adj.) and odometry only (Odometry). It can be seen qualitatively that our system (Ours) is able to reproduce fine details of the scene and well estimate instance mask predictions. In comparison, poses estimated using bundle adjustment (Bundle adj.) can generate good results, however, the resultant renders are noisier with noticeable artifacts for the plant leaves. Finally, it can be seen that using noisy odometry (Odometry) leads to incorrect estimates of the scene geometry presenting smeared panoptic predictions and degraded performance.

TABLE II: Performance for different camera extrinsic sources.

	PSNR[dB]	PQ [%]	IoU [%]
Robot odometry	20.27	60.15	79.80
Bundle Adjustment	21.92	61.54	77.44
Online optimization (Ours)	**23.37**	**70.08**	**82.65**

C. Speed and efficiency ablation study

Seeking to better understand how to tune our model's speed and efficiency, we progressively vary its grid parameters and obtain PAg-NeRF(S), a small yet competitive version of our model. In Fig. 5 we show the render and panoptic performance of our model on 3 successive parameter ablations, where we choose a suitable configuration from one experiment and apply this to the subsequent experiment. For each experiment, we

TABLE I: Comparison of PAg-NeRF with other relevant panoptic and semantic NeRF methods trained on the BUP20 dataset.

	ID Rej.	Uncert.	PSNR [dB] ↑	PQ [%] ↑	IoU [%] ↑	Training [min/seq] ↓	Inference[s/img] ↓	#Params. ↓
Mask2Former [22]	-	-	-	71.16	80.52	-	-	-
SemanticNerf [13]	✗	✗	19.01	-	77.81	123.2	61.3	0.63M
PanopticLifting [17]	✗	✓	21.34	56.65	76.28	36.5	8.6	7.42M
PAg-NeRF(L)	✗	✗	23.24	66.83	81.41	26.3	5.6	25.21M
	✓	✗	23.34	68.39	81.45	26.6		
	✓	✓	**23.37**	**70.08**	**82.65**	27.2		
PAg-NeRF(S) (sec. V-C)	✓	✓	21.37	66.10	79.86	**16.7**	**3.9**	**0.62M**

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

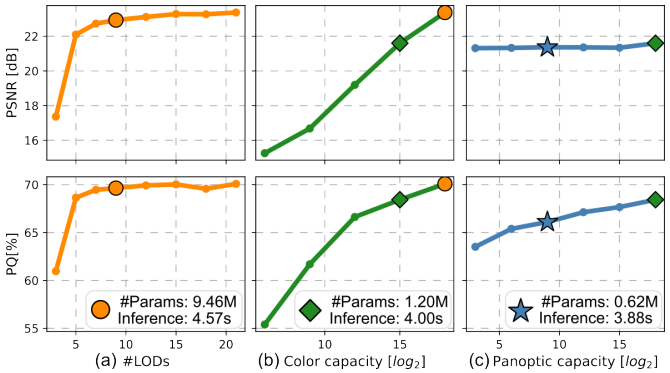

Fig. 5: Ablation study of feature grid parameters.

present the number of parameters and inference time per image to show how we can progressively improve both.

First, in Fig. 5.a, we take PAg-NeRF(L) (see Tab. I) and reduce the LODs of both grids simultaneously. We choose to set the number of LODs to 9 as it still maintains good performance while reducing the number of parameters and inference time by a factor of 2.65 and 1.2 respectively.

Second, we start from a model with 9 LODs and progressively reduce the color grid capacity. As can be seen in Fig. 5.b, this has a large impact on performance and so we choose to keep a high value of (2^{15}) for the color capacity. Despite retaining a relatively high number, we still reduce the number of parameters and improve the inference speed.

Third, the effect of reducing the panoptic delta grid capacity is shown in Fig. 5.c. As expected, this change has no impact on render quality, since the panoptic branch is detached from the color one. However, it does have a modest impact on PQ performance. We chose a panoptic capacity value of 2^9 to further reduce parameter count, trading off some performance.

These optimizations lead to our efficient PAg-NeRF(S) model. This model has approximately 39 times fewer parameters than PAg-NeRF(L) and is 1.5 times faster at inference time. Furthermore, it still has competitive performance beating the baseline methods (Tab.I).

D. Qualitative results

In Fig. 6 we compare the output of PAg-NeRF to Panoptic Lifting, Semantic NeRF and the Mask2Former detector in a cluttered scene. In this figure we concentrate on the render (RGB), instance and semantic segmentation quality against the ground truth (GT) frames. Further qualitative results for multiple frames and for 3D panoptic maps are available in the supplementary video and the project website **`http://claussmitt.com/pagnerf`**.

In terms of render quality, PAg-NeRF is able to reproduce very fine details of the fruits and leaf textures as well as high frequency edges and thin structures that get smoothed out by Panoptic Lifting. Moreover, Panoptic Lifting fails to reproduce fine details of masks and misses several fruits. In the case of Semantic NeRF, it can be seen that some of the fruit and leaves get blended together completely blurring their edges.

The panoptic mask quality (both instance and semantics) for PAg-NeRF is heavily influenced by the detection system, which in this case is Mask2Former. In Fig. 6 Mask2Former merges the large yellow pepper with the one behind it, and our model reproduces the same mistake. On the other hand, even though Mask2Former does not detect the pepper at the bottom left of the image, because it is detected in other images PAg-NeRF is able to recover from this error through its implicit modelling of 3D scene geometry.

Additionally, thanks to our repeated ID rejection loss, our model is able to properly distinguish between instances at the far ends of the frame, whereas Panoptic Lifting merges

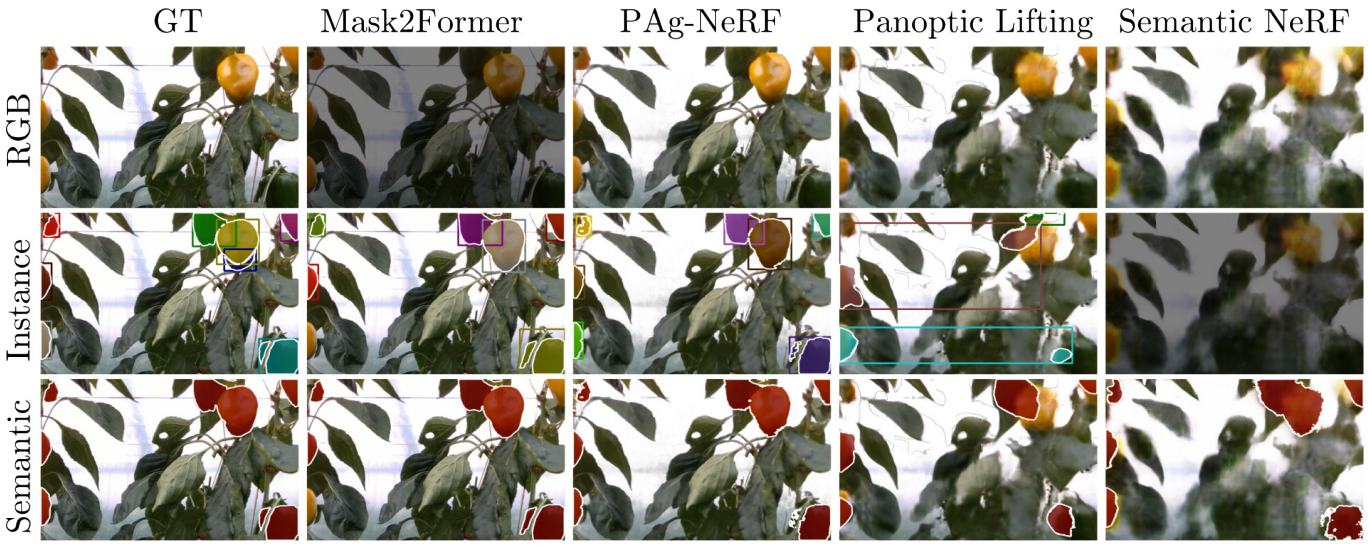

Fig. 6: PAg-NeRF results compared with Panoptic Lifting, Semantic NeRF, the input detections from Mask2Former and the ground truth (GT). We compare the results for the RGB images (first row), Instance results (second row), and Semantic results (third row). For the Instances, each mask color represents the ID assigned to them along with their bounding boxes. The long Instance bounding boxes of Panoptic Lifting are example failure cases of the plain linear assignment loss (Sec.III-C4). Our model renders fine details of the scene, which can be seen in the RGB image, and also produce sharp panoptic predictions with unique IDs for fruits close to image edges, which can be seen in the Instance and Semantic images.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

them into a single detection with long bounding boxes since it uses a plain linear assignment loss. Overall, PAg-NeRF is able to achieve accurate scene reconstruction, panoptic segmentation, and sequential ID assignment while being considerably quicker and more memory efficient than Panoptic Lifting.

VI. Summary & Future Work

We have presented a novel end-to-end 3D panoptic implicit representation that we validated in a challenging agricultural scenario. Our architecture is able to distinguish individual fruit instances, being trained only from RGB images with noisy robot poses and still-image panoptic segmentation detections with inconsistent fruit IDs. By leveraging online pose optimization, our modified instance ID linear assignment loss and hash-permutoedral grid encodings we are able to beat a state-of-the-art 3D panoptic NeRF approach. Moreover, our panoptic delta-grid architecture allowed us to trade off performance and efficiency, producing fast and efficient models that outperform the state-of-the-art. As future work, we plan to improve the 3D geometry of our representation and map entire cropping chambers to produce global phenotypic metrics suitable for crop phenotyping.

References

[1] A. Ahmadi, M. Halstead, and C. McCool, "Towards autonomous visual navigation in arable fields," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 6585–6592.

[2] A. You, C. Grimm, A. Silwal, and J. R. Davidson, "Semantics-guided skeletonization of upright fruiting offshoot trees for robotic pruning," *Computers and Electronics in Agriculture*, vol. 192, p. 106622, 2022.

[3] C. Lehnert, C. McCool, I. Sa, and T. Perez, "Performance improvements of a sweet pepper harvesting robot in protected cropping environments," *Journal of Field Robotics*, vol. 37, pp. 1197–1223, 2020.

[4] M. Halstead, A. Ahmadi, C. Smitt, O. Schmittmann, and C. McCool, "Crop agnostic monitoring driven by deep learning," *Frontiers in plant science*, vol. 12, 2021.

[5] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "Deepfruits: A fruit detection system using deep neural networks," *Sensors*, vol. 16, no. 8, p. 1222, 2016.

[6] M. Halstead, C. McCool, S. Denman, T. Perez, and C. Fookes, "Fruit quantity and ripeness estimation using a robotic vision system," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2995–3002, 2018.

[7] L. Zabawa, A. Kicherer, L. Klingbeil, A. Milioto, R. Topfer, H. Kuhlmann, and R. Roscher, "Detection of single grapevine berries in images using fully convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[8] C. Smitt, M. Halstead, A. Ahmadi, and C. McCool, "Explicitly incorporating spatial information to recurrent networks for agriculture," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 017–10 024, 2022.

[9] C. Smitt, M. Halstead, T. Zaenker, M. Bennewitz, and C. McCool, "Pathobot: A robot for glasshouse crop phenotyping and intervention," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 2324–2330.

[10] Y. Pan, F. Magistri, T. Läbe, E. Marks, C. Smitt, C. McCool, J. Behley, and C. Stachniss, "Panoptic Mapping with Fruit Completion and Pose Estimation for Horticultural Robots," 2023.

[11] R. A. Rosu and S. Behnke, "Permutosdf: Fast multi-view reconstruction with implicit surfaces using permutohedral lattices," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 8466–8475.

[12] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[13] S. Zhi, T. Laidlow, S. Leutenegger, and A. J. Davison, "In-place scene labelling and understanding with implicit scene representation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 838–15 847.

[14] L. Liu, J. Gu, K. Zaw Lin, T.-S. Chua, and C. Theobalt, "Neural sparse voxel fields," *Advances in Neural Information Processing Systems*, vol. 33, pp. 15 651–15 663, 2020.

[15] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics (ToG)*, vol. 41, no. 4, pp. 1–15, 2022.

[16] C.-H. Lin, W.-C. Ma, A. Torralba, and S. Lucey, "Barf: Bundle-adjusting neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5741–5751.

[17] Y. Siddiqui, L. Porzi, S. R. Bulò, N. Müller, M. Nießner, A. Dai, and P. Kontschieder, "Panoptic lifting for 3d scene understanding with neural fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9043–9052.

[18] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, "Panoptic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9404–9413.

[19] A. Kirillov, R. Girshick, K. He, and P. Dollár, "Panoptic feature pyramid networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6399–6408.

[20] B. Cheng, M. D. Collins, Y. Zhu, T. Liu, T. S. Huang, H. Adam, and L.-C. Chen, "Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 12 475–12 485.

[21] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, "Transformers in vision: A survey," *ACM computing surveys (CSUR)*, vol. 54, no. 10s, pp. 1–41, 2022.

[22] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1290–1299.

[23] J. Jain, J. Li, M. T. Chiu, A. Hassani, N. Orlov, and H. Shi, "Oneformer: One transformer to rule universal image segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 2989–2998.

[24] A. Kundu, K. Genova, X. Yin, A. Fathi, C. Pantofaru, L. J. Guibas, A. Tagliasacchi, F. Dellaert, and T. Funkhouser, "Panoptic neural fields: A semantic object-aware neural scene representation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 871–12 881.

[25] X. Fu, S. Zhang, T. Chen, Y. Lu, L. Zhu, X. Zhou, A. Geiger, and Y. Liao, "Panoptic nerf: 3d-to-2d label transfer for panoptic urban scene segmentation," in *2022 International Conference on 3D Vision (3DV)*. IEEE, 2022, pp. 1–11.

[26] C. Sun, M. Sun, and H.-T. Chen, "Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5459–5469.

[27] S. Kelly, A. Riccardi, E. Marks, F. Magistri, T. Guadagnino, M. Chli, and C. Stachniss, "Target-aware implicit mapping for agricultural crop inspection," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9608–9614.

[28] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "Tensorf: Tensorial radiance fields," in *European Conference on Computer Vision*. Springer, 2022, pp. 333–350.

[29] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger, "Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3504–3515.

[30] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[31] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5745–5753.

[32] T. Takikawa, O. Perel, C. F. Tsang, C. Loop, J. Litalien, J. Tremblay, S. Fidler, and M. Shugrina, "Kaolin wisp: A pytorch library and engine for neural fields research," https://github.com/NVIDIAGameWorks/kaolin-wisp, 2022.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.