2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

OTAS: Open-vocabulary Token Alignment for Outdoor Segmentation*

Simon Schwaiger[1,2], Stefan Thalhammer[2], Wilfried Wöber[2,3] and Gerald Steinbauer-Wagner[1]

***Abstract*— Understanding open-world semantics is critical for robotic planning and control, particularly in unstructured outdoor environments. Existing vision-language mapping approaches typically rely on object-centric segmentation priors, which often fail outdoors due to semantic ambiguities and indistinct class boundaries. We propose OTAS, an Open-vocabulary Token Alignment method for Outdoor Segmentation. OTAS addresses the limitations of open-vocabulary segmentation models by extracting semantic structure directly from the output tokens of pre-trained vision models. By clustering semantically similar structures across single and multiple views and grounding them in language, OTAS reconstructs a geometrically consistent feature field that supports open-vocabulary segmentation queries. Our method operates in a zero-shot manner, without scene-specific fine-tuning, and achieves real-time performance of up to ≈17 fps. On the Off-Road Freespace Detection dataset, OTAS yields a modest IoU improvement over fine-tuned and open-vocabulary 2D segmentation baselines. In 3D segmentation on TartanAir, it achieves up to a 151% relative IoU improvement compared to existing open-vocabulary mapping methods. Real-world reconstructions further demonstrate OTAS' applicability to robotic deployment. Code and a ROS 2 node are available at `https://otas-segmentation.github.io/`.**

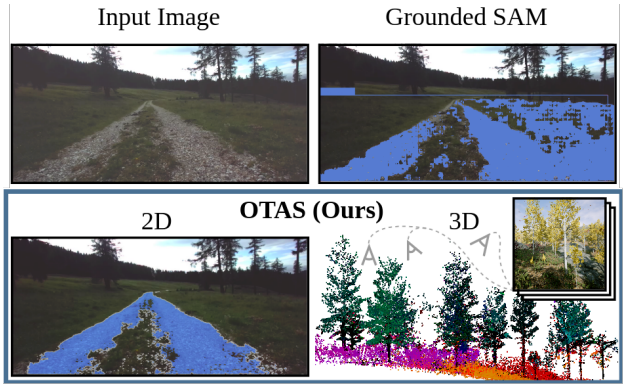

Fig. 1: **OTAS** is a training-free segmentation method that aligns tokens from vision and language foundation models for robotic outdoor tasks. It operates zero-shot on single (2D) or multi-view (3D) inputs and achieves real-time operation. For 2D, the prompt *"gravel road"* was used; 3D visualises *"trees"* in green, *"shrubbery"* in purple, *"grass"* in orange, and *"stone"* in red.

I. Introduction

Understanding the open world through semantics is a key challenge for robotics. Vision-Language Models (VLMs), that ground vision in language, have recently been shown to effectively provide semantics for mapping to facilitate task planning and navigation [1], [2]. However, open-vocabulary semantic mapping methods [3], [4], [5] rely on segmentation priors from general-purpose models to reason about the environment. These models are trained for object-centric knowledge retrieval, therefore, they are effective for segmenting structured settings with salient objects. However, segmentation fails in unstructured outdoor environments, such as forests or off-road paths (see Fig. 1). Unstructured, texture-rich classes relevant to outdoor robotics, such as roads and grass, are underrepresented in typical open-vocabulary image-text pair-based datasets and are often inconsistently labelled. Visual ambiguities and indistinct boundaries, such as overlaps between gravel and grass, further complicate the task for segmentation models, which leads to imprecise segmentation masks.

In order to obtain robust semantic segmentation in unstructured outdoor environments, we introduce **OTAS**, an **O**pen-vocabulary **T**oken **A**lignment method for Outdoor **S**egmentation. *Token alignment* refers to clustering self-supervised visual tokens into coarse semantic structures, then pooling co-located VLM tokens over these clusters for regularisation and language-grounding.

Instead of relying on language semantics for segmentation, we cluster tokens based on visual prototypes derived from self-supervised pre-trained vision models. Language-grounding is obtained through semantic and spatial alignment over token clusters, alleviating the need for linear probing or rendering. Optionally, multiple observations can be aligned to obtain a language-embedded reconstruction with geometric consistency. Hence, OTAS is not subject to the object-centric bias learned by general-purpose segmentation models, despite also performing zero-shot inference. The contributions of this study are:

- a *training-free token alignment* that fuses self-supervised visual tokens with language embeddings, regularising VLM features and improving non-object-class segmentation, without per-scene optimisation; and
- a *language-grounded 3D feature field* enabling real-

*This work was partly supported by the city of Vienna (MA23 – Economic Affairs, Labour and Statistics) through the project Stadt Wien Kompetenzteam für Drohnentechnik in der Fachhochschulausbildung (DrohnFH, MA23 project 35-02).

[1]Simon Schwaiger and Gerald Steinbauer-Wagner are with Graz University of Technology, Faculty of Computer Science and Biomedical Engineering, Institute of Software Engineering and Artificial Intelligence, 8010 Graz, Austria

[2]Simon Schwaiger, Stefan Thalhammer and Wilfried Wöber are with University of Applied Sciences Technikum Wien, Faculty of Industrial Engineering, Research Group Digital Manufacturing, Automation and Robotics, 1200 Vienna, Austria `schwaige@technikum-wien.at`

[3]Wilfried Wöber is with University of Natural Resources and Life Sciences, Department of Integrative Biology and Biodiversity Research, Institute for Integrative Nature Conservation Research, 1180 Vienna, Austria

TABLE I: **Comparison of Semantic Reconstruction Methods.** Assuming 10 fps as real-time (typical for low-dynamic settings like forests and agriculture) only OpenFusion and OTAS meet this threshold. Only LERF and OTAS use non-object-centric language maps. OTAS uniquely supports semantic segmentation in both 2D and 3D natively.

Method	Foundation Model	Real-Time	Zero-Shot	3D	2D	Representation	Not Object-Centric
LERF [6]	OpenCLIP [7], DINOv2 [8]	✗	✗	✓	✗	NeRF	✓
Feature Splatting[5]	CLIP [9], DINOv2 [8], SAM [10]	✗	✗	✓	✗	Gaussian Splatting	✗
ConceptGraphs [3]	OpenCLIP [7], SAM [10]	✗	✓	✓	✗	Points	✗
OpenFusion [4]	SEEM [11]	✓	✓	✓	✗	TSDF	✗
OTAS (ours)	CLIP [9], DINOv2 [8], SAM2 [12]	✓	✓	✓	✓	Points	✓

time mapping and open-vocabulary querying, built from aligned tokens, requiring no per-scene trained Multi-Layer Perceptrons (MLPs), and no differentiable rendering.

We demonstrate token alignment for 2D and 3D segmentation as well as semantic reconstruction tasks, where it achieves real-time inference on GPU. OTAS improves segmentation results on Off-Road Freespace Detection (ORFD) [13] and TartanAir [14]. Additional experiments on robot data demonstrate the advantage of OTAS for language-embedded reconstruction of unstructured outdoors in comparison to volumetric rendering with LERF [6] and Feature Splatting [5]. Finally, critical design decisions, including token alignment, clustering methods, number of token clusters, and backbone choice are ablated to motivate the recommended model configurations for robotic applications.

II. Related Work

VLMs ground vision in language by encoding a joint feature space, typically extracting one feature per image or patch [9], [7]. Many robotic tasks, however, require fine-grained spatial relationships. This motivates mapping VLM features to queryable semantic maps [3], [4], [6], [15], [16].

Early VLM-based navigation approaches detect objects, extract VLM features per instance, and ground them on 2D occupancy grids (e.g., VLMaps [1], VLFM [17]) by interpolating features spatially. They rely on general-purpose detection or segmentation models, which introduce an object-centric prior into feature extraction [18]. This paradigm has been extended to 3D. OpenFusion [4] fuses SEEM [11] features into a 3D semantic map through Simultaneous Localisation and Mapping (SLAM). Similarly, ConceptGraphs [3] uses SAM [10] masks and OpenCLIP [7] features, projected to 3D and fused via geometric and semantic similarity. While effective indoors, all retain object-centric biases from their segmentation models.

An alternative direction is to reconstruct language-grounded feature fields. Feature Splatting [5] retains object priors since it uses SAM for generating segmentation masks. LERF [6] avoids object priors by extracting multiscale OpenCLIP features, yielding dense, non-object-centric feature maps refined via neural rendering. Both rely on geometric consistency across views. Using neural scene representations, such as LERF or Feature Splatting, requires rendering, resulting in slow scene-specific training and making them neither zero-shot nor real-time capable. Similar to rendering-based methods, OpenScene [19] distils multi-view CLIP features into a sparse 3D network. However, this requires computationally intensive scene-specific training, making the method label-free but not training-free on new scenes.

Table I compares state-of-the-art semantic reconstruction methods for outdoor robot navigation relevance. Key requirements include real-time performance for robot control, zero-shot applicability to new environments, and avoidance of object-centric priors for accurate segmentation of non-salient objects. OTAS is the only method meeting all criteria, enabling training-free and fast robotic deployment.

III. Method

VLMs, such as Grounded SAM and SEEM are biased towards object-centric knowledge retrieval [20]. This becomes especially problematic in the unstructured environments of outdoor robotics, where the semantic classes of interest fall outside the a priori encoded object-centric knowledge. Examples of such classes are road, woods, and shrubbery, which, however, are highly relevant to mobile outdoor robotics.

Self-supervised pre-trained vision foundation models, such as DINOv2 [8], do not have this limitation, since they are not trained directly on segmentation tasks. Their training process results in an emergent semantic organisation of the feature space, where semantically similar classes are embedded adjacently. Hence, we disentangle the open-vocabulary semantic segmentation by using DINOv2 for coarse zero-shot semantic clustering, followed by natural language-grounding by pooling over CLIP's vision-language embeddings. Input views are embedded by the frozen vision and vision-language encoders, see Fig. 2 (a). Output tokens of the vision encoder are clustered to obtain semantic structures (b), and aligned with vision-language tokens to obtain language-grounding (c). The language-grounded semantic clusters are used as priors for optional zero-shot upscaling to pixel level (d) [12]. Optional spatial regularisation of steps b and c increases geometric consistency and allows multi-view reconstruction and segmentation (e).

A. Visual Feature Clustering

Given a monocular input image $I \in \mathbb{R}^{H \times W \times 3}$, our goal is to generate a semantic segmentation mask guided by both vision and language, with $\times$ denoting tensor dimensions (e.g., spatial height $\times$ width). The input image is first processed by a frozen vision encoder $\mathcal{E}_v$ to produce a

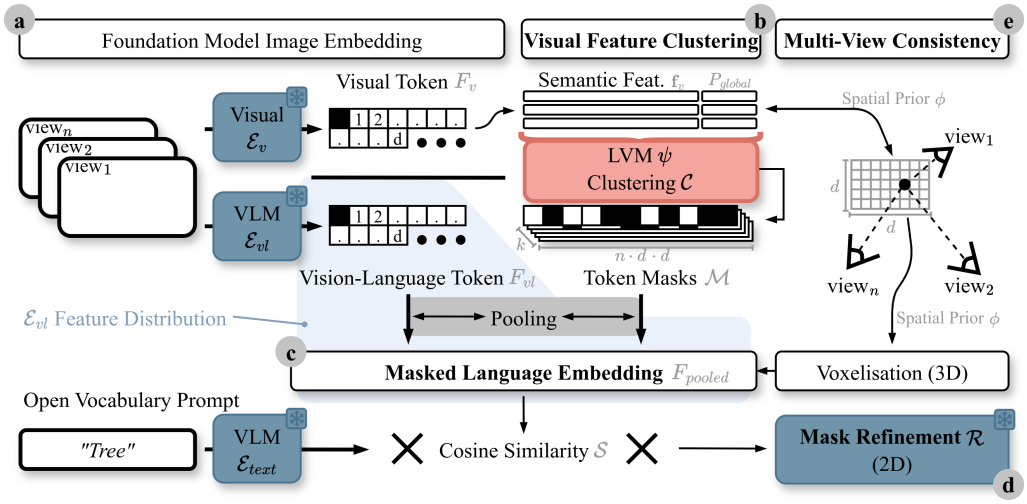

Fig. 2: **Method Overview.** a) OTAS encodes input views using frozen encoders. b) Patch tokens of the visual encoder are reduced and clustered to obtain semantic masks. c) The masks are pooled with normalised patch tokens of a vision-language encoder for natural language-grounding. d) A frozen mask refinement network projects semantic similarity to prompts to pixel-level. e) Clustering and pooling are optionally conditioned on environment geometry through projection.

coarse spatial feature map $F_v = \mathcal{E}_v(I) \in \mathbb{R}^{H' \times W' \times C_v}$. To align vision with language, F_v is interpolated to a shared feature dimension d using bilinear interpolation ($\mathcal{U}_{bl}$). The interpolated features are then flattened and L2 normalised, denoted by $\hat{\mathbf{f}}_v$. The flattened feature map is decorrelated and reduced in dimensionality using a Latent Variable Model (LVM) ψ, resulting in $\hat{\mathbf{f}}_{LVM} = \psi(\hat{\mathbf{f}}_v) \in \mathbb{R}^{d \cdot d \times C_r}$, where the reduced feature dimension C_r is a hyperparameter.

Subsequently, a clustering model $\mathcal{C}$ is applied to the flattened feature map $\hat{\mathbf{f}}_{LVM}$ to derive k clusters, that constitute mixtures of visual tokens, referred to as visual prototypes. The affiliation of each data point to a cluster is denoted by $\mathcal{C} = \{\mathcal{C}_1, ..., \mathcal{C}_{d \cdot d}\}, C_j \in \{1, ..., k\} \forall j$, representing the assignment of the latent representations $\hat{\mathbf{f}}_{LVM}$ to a visual prototype. The clusters are interpreted as a set of k binary masks $\mathcal{M} = \{\mathcal{M}_1, ..., \mathcal{M}_k\}$, where each mask $\mathcal{M}_i \in \{0, 1\}^{n \times d \times d}$ corresponds to the shared feature dimension d across n input images.

B. Masked Language Embedding

DINOv2 embeddings are not correlated with semantics such as language. An intuitive way to retrieve semantic categories is linear probing. This, however, requires annotated data in the target domain. Instead, we use a vision-language encoder $\mathcal{E}_{vl}$ to produce language-grounded tokens and align them with the visual tokens, resulting in $F_{vl} = \mathcal{E}_{vl}(I) \in \mathbb{R}^{H_{vl} \times W_{vl} \times C_{vl}}$. To extract dense patch-level features from the vision-language encoder, we use value features from the final attention layer rather than after global pooling, which preserves the vision-language association for dense prediction [21]. These tokens are subsequently interpolated to match d using nearest neighbour interpolation ($\mathcal{U}_{nn}$): $F_{vl}^{shared} = \mathcal{U}_{nn}(F_{vl}) \in \mathbb{R}^{d \times d \times C_{vl}}$. We adopt Masked Average Pooling (MAP) to address token alignment, following [5], who showed its regularising effect on VLMs. Unlike prior work, we apply MAP over coarse feature maps in the shared embedding space rather than at pixel level. MAP computes the mean language feature vector over each mask, conditioning VLM outputs on the masks' semantic structure. This pooling operation retains language-grounding in the feature distribution of $\mathcal{E}_{vl}$. Soft and overlapping language semantics are preserved by aggregating each region rather than collapsing features into a single semantic mode. Pooling is done per image, also in the case of multi-view inputs.

$$F_{pooled}(x, y) = \frac{1}{|M_c|} \sum_{(x,y) \in M_c} F_{vl}^{shared}(x, y) \quad (1)$$

Since each patch is only assigned to a single mask in $\mathcal{M}$, the resulting F_{pooled} is a feature map of shape $d \times d \times C_{vl}$. F_{pooled} represents a language-grounded image embedding, regularised by the token mask structure (see Fig. 3). Ultimately, pooled features are normalised using the L2 norm.

A frozen text encoder $\mathcal{E}_{text}$ maps text prompts to the vision-language feature dimension $F_{text} = \mathcal{E}_{text}(t) \in \mathbb{R}^{C_{vl}}$. Cosine similarity $\mathcal{S}$ between F_{text} and each feature in F_{pooled} produces a similarity map of shape $d \times d$. As done by [6], [5], $\mathcal{E}_{text}$ and the similarity computation are applied to a set of positive prompts t^+ and negative prompts t^-, indicating target and undesired concepts, respectively, resulting in the combined similarity map $\mathcal{S}_{combined}$.

$$\mathcal{S}_{combined} = \sum_{t \in t^+} \mathcal{S}(t, F_{pooled}) - \sum_{t \in t^-} \mathcal{S}(t, F_{pooled}) \quad (2)$$

C. Mask Refinement

We use the similarity map as a language-grounded prior to obtain a binary pixel-level segmentation mask M. Depending on the used encoders and interpolation to the shared feature resolution d, the similarity map resolution will be lower than the input image resolution. Typically, the similarity map is

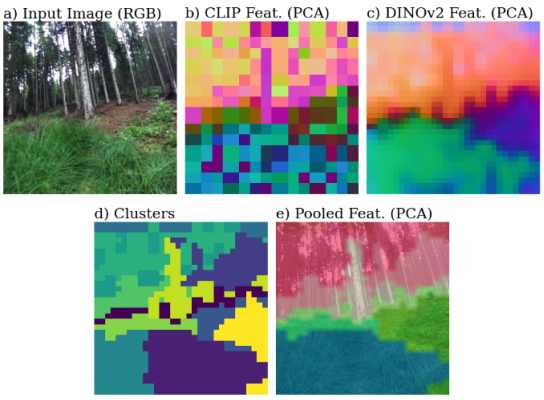

Fig. 3: **Feature comparison.** CLIP (b) [9] features include view-dependent noise that is detrimental to segmentation accuracy [5]. We achieve regularisation in non-object-centric environments by extracting visual prototypes from DINOv2 (c) [8], with k-Means clustering (d) and language-grounding via feature pooling (e).

at 1/7th or 1/14th of the input image resolution. In order to refine the coarse mask we employ a frozen mask refinement network $\mathcal{R}$ that takes the image I and the similarity map $\mathcal{S}_{combined}$ as input and outputs the final high-resolution segmentation mask.

$$M = \mathcal{R}(I, \mathcal{U}_{bl}(\mathcal{S}_{combined})) \in \{0,1\}^{H \times W} \tag{3}$$

D. Multi-View Consistency

To expand OTAS to the multi-view case, information is aggregated over multiple views using the depth map $D \in \mathbb{R}^{H \times W}$ and camera pose $T \in SE(3)$ associated with each frame. During image embedding, D is projected to 3D points $P \in \mathbb{R}^{N \times 3}$. Median depth $\tilde{D}$ is sampled in each grid of size $d \times d$ to align the 3D points with the vision and vision-language features. Using camera intrinsics K, 3D points P are projected to the image plane via $P = \pi(\tilde{D}, K) \in \mathbb{R}^{d \cdot d \times 3}$. A mapping ϕ tracks the relationship between 3D points P_u and patch indices (i, j). The points are transformed to a global coordinate frame using camera poses $\{T_1, ..., T_n\}$ to construct $P_{global} = \bigcup_{u=1}^{n} T_u P_u$.

Spatially Conditioned Clustering. Global point positions and relationship ϕ allow conditioning the visual feature clustering by concatenating semantic features F_v^{shared} with 3D coordinates P_{global}. This yields a combined feature map $F_{spatial} \in \mathbb{R}^{d \cdot d \times (C_v + 3)}$ that replaces F_v^{shared} as the LVM input, where each feature vector $F_{spatial}(i, j)$ contains both semantic and spatial information for corresponding points p.

Spatially Conditioned Pooling. After pooling the visual and vision-language features for each input view separately, each F_{pooled} is projected on the global point cloud P_{global} using the relationship ϕ, resulting in a spatial 3D feature volume $P_{semantic} \in \mathbb{R}^{d \cdot d \times (C_{vl} + 3)}$ where $P_{semantic} = \text{concat}(F_{pooled}(i, j), P_{global}(p)) \mid p \in P_{global}, (i, j) = \phi(p)$. The feature volume consists of keypoint position and language-grounded feature embedding pairs. Knowing the keypoint position, the feature volume is downsampled using a configurable voxel-size v. During downsampling, all pooled features in a voxel are linearly interpolated to further condition the language-embeddings with spatial context, where $\hat{P}_{semantic} = (\frac{1}{|V_k|} \sum_{(f,p) \in V_k} f)$ with $V_k = \{(f, p) \in P_{semantic} \mid \lfloor \frac{p}{v} \rfloor = k\}$. $\hat{P}_{semantic}$ describes a language-queryable 3D occupancy grid directly usable for robotic applications such as obstacle avoidance and goal-based navigation.

IV. Experiments

Datasets and Metrics. Monocular semantic segmentation is evaluated on the Off-Road Freespace Detection Dataset (ORFD) [13]. ORFD aims to identify traversable road types in the outdoors, such as gravel, dirt and sand. RELLIS-3D [22] is used in ablations as a stress test due to the high semantic overlap between annotated classes and fuzzy class boundaries. 3D feature reconstruction is evaluated on TartanAir [14], a large-scale, photorealistic synthetic dataset for visual SLAM and robot navigation.

Since TartanAir does not provide 3D ground truth labels, 3D labels are generated for all methods by projecting 2D labels onto the reconstructed point clouds via majority vote over each point's 5 nearest neighbours, following [23]. In order to evaluate unstructured outdoor segmentation, we evaluate segmenting vegetation, labels 152 and 109. Following previous work [24], Intersection over Union (IoU), F-score (Fsc), Precision (Pre), and Recall (Rec) are evaluated for all quantitative experiments. Practical applicability to robotic applications is demonstrated through qualitative real-world reconstruction in the alps [25] and runtime and memory footprint analysis in 2D and 3D.

Implementation Details. OTAS is provided in three configurations. All models use CLIP ViT-B-16 [9], [21] and DINOv2 ViT-S-14 with 4 registers [8], [26]. *OTAS Small* uses a shared feature dimension of $d = 16$ and SAM2.1 Hiera-T [12] for mask refinement. *OTAS Large* uses $d = 32$ and SAM2.1 Hiera-L. *OTAS Spatial* uses $d = 64$, a voxel-size of $v = 0.5m$, and no mask refinement, as segmentations are regularised geometrically. All models use GPU-accelerated Principal Component Analysis (PCA) for ψ and k-Means for $\mathcal{C}$. Evaluations are done on an Intel i7-12700 CPU and NVIDIA 4070 Ti Super GPU.

A. 3D Outdoor Segmentation

Semantic mapping is evaluated against Concept Graphs [3] and OpenFusion [4], since they create language-embedded 3D pointclouds, similarly to OTAS. Both methods serve as the state of the art for zero-shot semantic scene reconstruction in robotics, as they are not domain-specific and do not require a pretrained map prior (e.g., encoded in an MLP). Since both methods do not directly provide semantic labels, but rather language-grounded point clouds, we threshold using the same language queries as for OTAS.

TartanAir. Table II presents 3D segmentation results on outdoor scenes of TartanAir using the first annotated trajectory. OTAS improves all evaluated metrics over OpenFusion and ConceptGraphs on Amusement, Gascola, and

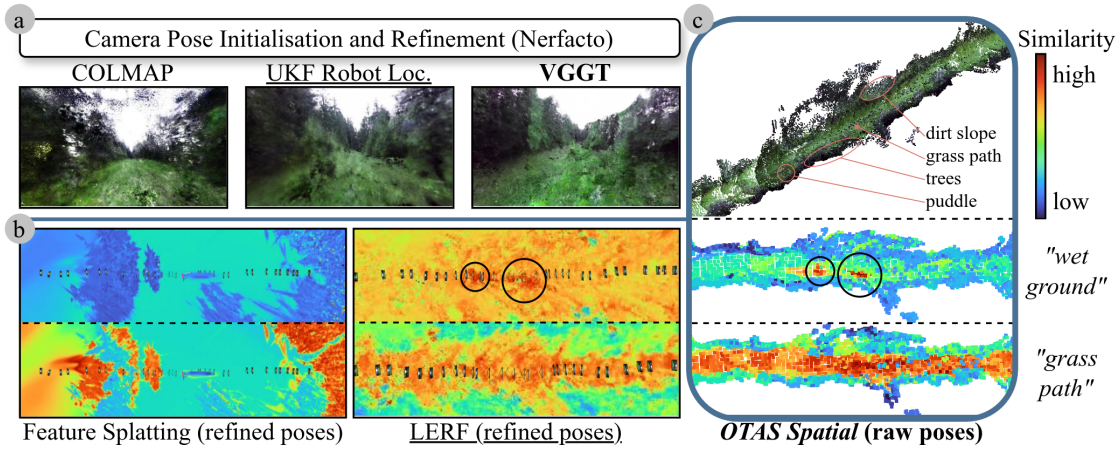

Fig. 4: **Alpine Ground Analysis.** Language-embedded reconstruction requires accurate camera poses. a) Reconstruction obtained using COLMAP, UKF Robot Localisation, and VGGT. All poses are refined using Nerfacto. b) Semantic similarity of Feature Splatting and LERF to prompts. c) Semantic reconstruction and prompt similarity of *OTAS Spatial*.

TABLE II: **3D Vegetation Segmentation on TartanAir.** All methods reconstruct a language-grounded point cloud given known camera poses. Time denotes total semantic reconstruction time excluding evaluation. We compare per point segmentation performance in identifying vegetation.

	IoU↑	Fsc↑	Pre↑	Rec↑	Time[s]↓
Amusement					
OpenFusion [4]	23.13	37.09	39.17	37.86	55
ConceptGraphs [3]	34.86	46.15	47.00	48.17	2201
OTAS Spatial (**Ours**)	**47.11**	**64.04**	**65.16**	**65.48**	**22**
Gascola					
OpenFusion [4]	10.24	18.37	18.23	20.36	52
ConceptGraphs [3]	30.68	38.03	30.68	50.00	333
OTAS Spatial (**Ours**)	**67.87**	**80.27**	**79.23**	**81.73**	**12**
Seasonsforest					
OpenFusion [4]	25.09	35.18	47.38	39.07	53
ConceptGraphs [3]	17.39	28.96	51.06	52.25	151
OTAS Spatial (**Ours**)	**43.63**	**57.23**	**57.09**	**57.42**	**10**
Seasonsforest Winter					
OpenFusion [4]	22.16	36.01	39.37	40.84	103
ConceptGraphs [3]	36.48	53.33	54.26	54.49	479
OTAS Spatial (**Ours**)	**39.61**	**55.13**	**56.22**	**55.33**	**18**

Seasonsforest. Especially in environments with barely any discrete objects, such as Gascola, the margin for improvement is huge, reaching up to 151% on IoU. The lower contrast reduces segmentation quality of object-centric open-vocabulary segmentation, highlighting the advantages of OTAS for outdoor robotics. We observe that ConceptGraphs performs closer in snowy scenes of Seasonsforest Winter. This is likely due to the high contrast between objects and the uniform snow, which enhances object boundaries and thus benefits object-centric methods.

B. 2D Outdoor Segmentation

ORFD. This section compares OTAS to the state of the art for fine-tuned and open-vocabulary 2D semantic segmentation. For open-vocabulary, we report Grounded SAM [31] and SEEM [11], since these are the models used by Concept Graphs [3] and OpenFusion [4] respectively. SAM- and SEEM-based methods currently define the state of the art in open-vocabulary segmentation, and are therefore natural baselines. While the original Grounded SAM-2 implementation relies on SAM2, we run it with the improved SAM2.1 Hiera-L segmentation head to provide a best-case scenario and fair comparison to our method.

TABLE III: **2D Semantic Segmentation on ORFD.** We include the current state of the art in fine-tuned off-road segmentation methods as well as other zero-shot segmentation methods that serve as the baseline for language-grounded semantic scene representations. The † indicates results optioned from the reimplementation by [27].

Fine-tuned Methods	IoU	Fsc	Pre	Rec	
OFF-Net [13]	82.30	90.30	86.60	94.30	
RTFNet† [28]	90.70	95.10	93.80	96.50	
RoadFormer [27]	92.51	96.11	95.08	**97.17**	
M2F2-Net [29]	93.10	96.40	97.30	95.50	
NAIFNet [30]	94.10	97.00	97.50	96.40	
Zero-Shot Methods	IoU	Fsc	Pre	Rec	fps
SEEM [11]	51.31	59.12	61.44	60.93	**15.0**
Grounded SAM [31]	90.49	94.13	95.12	93.32	1.8
Grounded SAM-2 [32]	93.32	96.38	97.73	95.38	3.8
OTAS Small (**Ours**)	91.72	95.59	96.93	94.58	11.2
OTAS Large (**Ours**)	**94.34**	**97.05**	**97.83**	96.39	5.1

Table III presents results on ORFD. OTAS achieves the highest IoU, F-score and precision among fine-tuned and zero-shot methods. OTAS reports the highest recall among zero-shot methods. Yet, the segmentation recall of the fine-tuned RoadFormer marginally improves over OTAS. Interestingly, this phenomenon can be observed for all zero-shot methods. They exhibit lower recall compared to fine-tuned methods. This is a consequence of the lack of dense supervision for specific classes and the necessity to generalise over broad, noisy semantics, whereas fine-tuned models directly optimise for segmenting the specific classes, including dataset characteristics like annotation errors and noise.

TABLE IV: **Influence of Model Size.** Comparison of accuracy, memory and fps of OTAS on ORFD. *No Token Alignment* ablates token alignment and directly prompts from CLIP similarity maps. Both OTAS versions without mask refinement significantly outperform directly prompting mask refinement from CLIP similarity maps (line 2) w.r.t. segmentation quality and throughput, validating our token alignment strategy.

Model	Mask Refinement	IoU (%)	Fsc (%)	Pre (%)	Rec (%)	GPU Mem. (GB)	fps (s^{-1})
No Token Alignment (GPU)	no	68.25	80.46	79.57	82.48	1.6	≈25
	yes	75.48	84.54	92.90	82.03	2.4	≈13
Small (GPU)	no	84.71	91.35	91.12	92.84	1.6	≈17
	yes	91.72	95.59	96.93	94.58	2.4	≈11
Small (CPU)	no	84.80	91.41	91.20	92.87	-	≈1.6
	yes	91.71	95.58	96.93	94.57	-	≈0.38
Large (GPU)	no	87.02	92.69	92.3	94.4	1.6	≈15
	yes	94.34	97.05	97.83	96.39	3.5	≈5

Token alignment, runtime and memory scaling as well as backbone choice are ablated in Sec. IV-D (see Table IV-VI and Fig. 5). These experiments demonstrate that OTAS maintains efficiency across varying cluster sizes and performs across different foundation models beyond the highlighted DINOv2/CLIP version.

C. Real-World Semantic Reconstruction

This section directly compares OTAS to LERF [6] and Feature Splatting [5] for semantic reconstruction in the foothills of the Alps. While neither zero-shot nor real-time due to their reliance on scene-specific training, they both represent the strongest existing baselines for language-embedded reconstruction. In particular, LERF's multiscale CLIP feature field avoids segmentation priors, making it non-object-centric and conceptually closest to OTAS. We therefore include it despite the runtime mismatch, as it illustrates the trade-off between accurate but computationally expensive differential rendering approaches and our training-free, real-time alternative. We use a ROS bagfile of RoboNav [25]. This allows for reproducible testing on real sensor data since the bagfile captures the full sensor and actuation context of the robot in representative environments.

Fig. 4 shows language-embedded reconstructions of a challenging forest scene featuring dense vegetation and different ground types, such as grass, dirt and puddles. LERF and Feature Splatting require highly accurate camera poses for reconstructing scenes with differential rendering. Usually, Structure from Motion, like COLMAP [33], is used for camera pose initialisation. However, due to the cluttered, highly-textured scene, COLMAP, UKF Robot Localisation [34], and VGGT [35] fail to provide poses with sufficient accuracy, see Fig. 4a. Hence, camera poses are initialised using VGGT, scaled using metric depth estimation [36], and refined using Nerfacto [37]. Even with pose refinement, Feature Splatting fails to properly reconstruct the ground. LERF correctly locates the grass-path itself and puddles (black circles) thanks to non-object-centric language-grounding, Fig. 4b. However, it is computationally intensive with ≈40 minutes for this scene. OTAS shows a geometrically accurate reconstruction with detailed language similarity at ≈1.3 seconds, Fig. 4c. All runtime reports exclude pose initialisation and pose refinement.

D. Ablations

Model Size and Inference Time (2D). We provide multiple model configurations for different compute capabilities. Table IV presents their speed-accuracy trade-off on GPU and CPU. Small and Large model configurations are outlined in Section IV. *No mask refinement* refers to normalising the similarity map $\mathcal{S}_{combined}$ to $[0, 1]$ and binary thresholding. No alignment with mask refinement represents $\mathcal{R}(I, \mathcal{U}_{bl}(\mathcal{S}(F_{text}, F_{vl})))$ and is equivalent to prompting SAM2.1 from CLIP similarity maps. OTAS Small no refinement (line 3) significantly improves IoU and results in improved throughput, validating our token alignment strategy for feature regularisation. Mask refinement further improves accuracy and adds ≈50% to runtime on GPU. *OTAS Small* runs at real-time (assuming 10 fps).

We also report few-shot segmentation speeds on ORFD on a Jetson Orin AGX, showing OTAS' applicability to low-power embedded robotics platforms. To reduce CPU-related overhead, ψ and $\mathcal{C}$ are pre-trained on-device using 10 training images. On Jetson, OTAS Small achieves ≈3.1 fps with mask refinement and ≈5 fps without mask refinement.

Foundation Model Choice. Foundation model dependence is evaluated on RELLIS-3D [22], a challenging off-road dataset with highly textured classes and semantically overlapping categories (e.g., "dirt," "mud," "puddle"). Unlike purpose-trained methods that reach ≈75% IoU on some classes [22], open-vocabulary zero-shot approaches underperform due to ambiguous class boundaries. We therefore use Rellis-3D as a close-to-real-world stress test across different vision backbones. Table V compares OTAS with

TABLE V: **Vision Backbones on RELLIS-3D.** Token alignment is evaluated across frozen backbones using raw class labels as prompts and no mask refinement. OTAS achieves higher IoU than Grounded SAM-2 with an overall significantly lower parameter count, highlighting OTAS' ability to regularise compact backbones into competitive open-vocabulary segmentation models without additional training.

Model Configuration	mIoU(%)↑	Param (M)↓
Grounded SAM-2	45.11	396
OTAS w. DINOv3 ViT-S/16 [38]	48.44	107
OTAS w. C-RADIOv3-B [39]	48.46	184
OTAS w. DINOv2 ViT-S/14 [8]	48.48	107

TABLE VI: **Dimensionality Reduction and Clustering Algorithms.** Score is the average of IoU, Fsc, Pre, and Rec of *OTAS Small* with mask refinement on ORFD.

Clustering	PCA	KPCA	PCA (GPU)	ICA
GMM	0.9381	0.9395	0.9390	0.9325
HDBSCAN	0.9394	0.9394	0.9394	0.9246
k-Means (GPU)	0.9427	0.9427	0.9424	0.9312
k-Means	0.9466	**0.9467**	**0.9467**	0.9363

Grounded SAM-2 and alternative foundation model choices for the visual encoder $\mathcal{E}_v$. Alternative foundation models are DINOv3 [38], a joint-embedding self-supervised model and successor to DINOv2, and AM-RADIO [39], which is achieved by distilling multiple foundation models into a single backbone. All experiments use raw class labels as prompts without tuning and a generic negative prompt of *thing*. Classes required for traversability assessment (i.e., dirt, water, asphalt, bush, mud, rubble) are evaluated with reported mean IoU (mIoU) over all classes equally weighted. To isolate the performance of token alignment, mask refinement is deactivated for OTAS results. Instead, similarity maps are thresholded at 0.8.

All three foundation models combined with token alignment outperform Grounded SAM-2 despite using fewer parameters, showing that OTAS lifts frozen vision-language features into a more discriminative representation without training or fine-tuning additional segmentation heads. However, performance when using the larger AM-RADIO (98M) backbone does not improve upon the significantly smaller DINO models (21M) when using the same CLIP image encoder for language-grounding (86M).

Reduction and Clustering Methods. Table VI examines the choice of LVM (ψ) and clustering model (C) in a factorial experiment, using PCA (CPU and GPU), KPCA and ICA for LVM and k-Means (CPU and GPU), Gaussian Mixture Model (GMM) and HDBSCAN for clustering. This comparison shows that k-Means clustering leads to the cleanest segmentation results, with PCA and Kernel-PCA being equally suitable for dimensionality reduction. Density-based clustering (HDBSCAN) is a viable alternative if setting the number of clusters as a hyperparameter is not possible.

Number of Clusters and Components. The top of Fig. 5 shows an ablation of PCA components (C_r) and k-Means clusters (k) on a 20% split of ORFD. Results are truncated for length from a grid search over $k = [4; 20]$, $C_r = [4; 64]$ with marginal score difference between the best (0.94) and the worst-performing (0.92) combination. Positive prompts are *gravel, road, dirt* and negative prompts are *sky, grass, forest*. The denoted score is an average of the IoU, F-score, precision and recall. $C_r = 4$ and $k = 4$ score the highest.

Model Scalability (3D). Bottom of Fig. 5 shows the time requirements for reconstruction with *OTAS Spatial* on TartanAir Seasonsforest Winter in blue, and the memory usage in orange. Both time and space complexity are comparably low to the state of the art and scale approximately linearly over the measured input view range. At 10 views, both time and memory usage are marginally above 1 second and Gigabyte respectively. Using 250 views takes 14.78 seconds and requires 11.26 Gigabytes of GPU memory, resulting in an average throughput of ≈17 fps.

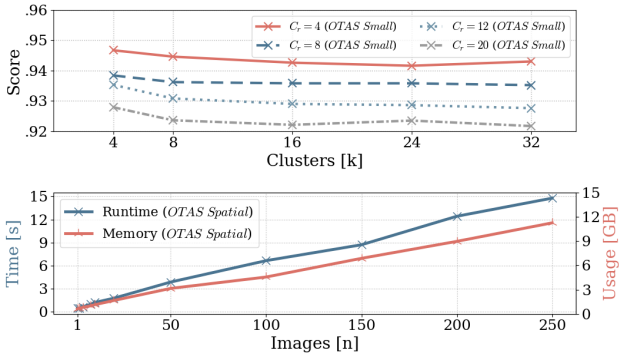

Fig. 5: **Clusters, Components, Runtime and Memory.** Top presents the number of k-Means clusters (k) and number of components (C_r). Bottom shows runtime and memory usage.

V. Conclusion

This work addressed open-vocabulary segmentation in unstructured outdoor environments. We introduce OTAS, an open-vocabulary segmentation method that aligns semantic tokens across single and multiple views to reconstruct a geometrically consistent feature field. It aligns the output tokens of a pre-trained vision model to a language embedding by clustering semantically similar tokens through unsupervised learning and pooling. OTAS is zero-shot, does not require scene-specific fine-tuning, and runs at up to ≈17 fps. Results show a minor improvement over open-vocabulary and fine-tuned baselines on the ORFD dataset, a significant improvement over the state of the art on TartanAir, and robust applicability to real-world robotic tasks. Scaling, runtime and backbone ablations confirm that OTAS is both efficient and backbone-agnostic, addressing concerns about model dependence and deployment trade-offs. Future work will investigate employing our semantic maps for outdoor navigation, e.g., through costmap modification [16] or with learned policies [40].

References

[1] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 10 608–10 615.

[2] B. Chen, F. Xia, B. Ichter, K. Rao, K. Gopalakrishnan, M. S. Ryoo, A. Stone, and D. Kappler, "Open-vocabulary queryable scene representations for real world planning," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 11 509–11 522.

[3] Q. Gu, A. Kuwajerwala, S. Morin, K. M. Jatavallabhula, B. Sen, A. Agarwal, C. Rivera, W. Paul, K. Ellis, R. Chellappa *et al.*, "Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 5021–5028.

[4] K. Yamazaki, T. Hanyu, K. Vo, T. Pham, M. Tran, G. Doretto, A. Nguyen, and N. Le, "Open-fusion: Real-time open-vocabulary 3d mapping and queryable scene representation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 9411–9417.

[5] R.-Z. Qiu, G. Yang, W. Zeng, and X. Wang, "Language-driven physics-based scene synthesis and editing via feature splatting," in *European Conference on Computer Vision (ECCV)*, 2024, pp. 368–383.

[6] J. Kerr, C. M. Kim, K. Goldberg, A. Kanazawa, and M. Tancik, "Lerf: Language embedded radiance fields," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 19 672–19 682.

[7] M. Cherti, R. Beaumont, R. Wightman, M. Wortsman, G. Ilharco, C. Gordon, C. Schuhmann, L. Schmidt, and J. Jitsev, "Reproducible scaling laws for contrastive language-image learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 2818–2829.

[8] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*," 2023.

[9] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 2021, pp. 8748–8763.

[10] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollar, and R. Girshick, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 4015–4026.

[11] X. Zou, J. Yang, H. Zhang, F. Li, L. Li, J. Wang, L. Wang, J. Gao, and Y. J. Lee, "Segment everything everywhere all at once," in *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36. Curran Associates, Inc., 2023, pp. 19 769–19 782.

[12] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer, "Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*," 2024.

[13] C. Min, W. Jiang, D. Zhao, J. Xu, L. Xiao, Y. Nie, and B. Dai, "Orfd: A dataset and benchmark for off-road freespace detection," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 2532–2538.

[14] W. Wang, D. Zhu, X. Wang, Y. Hu, Y. Qiu, C. Wang, Y. Hu, A. Kapoor, and S. Scherer, "Tartanair: A dataset to push the limits of visual slam," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 4909–4916.

[15] N. M. M. Shafiullah, C. Paxton, L. Pinto, S. Chintala, and A. Szlam, "Clip-fields: Weakly supervised semantic fields for robotic memory. *arXiv preprint arXiv:2210.05663*," 2022.

[16] R.-Z. Qiu, Y. Hu, Y. Song, G. Yang, Y. Fu, J. Ye, J. Mu, R. Yang, N. Atanasov, S. Scherer, and X. Wang, "Learning generalizable feature fields for mobile manipulation," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025, pp. 20 952–20 959.

[17] N. Yokoyama, S. Ha, D. Batra, J. Wang, and B. Bucher, "Vlfm: Vision-language frontier maps for zero-shot semantic navigation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 42–48.

[18] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*. PMLR, 2023, pp. 19 730–19 742.

[19] S. Peng, K. Genova, C. ". Jiang, A. Tagliasacchi, M. Pollefeys, and T. Funkhouser, "Openscene: 3d scene understanding with open vocabularies," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 815–824.

[20] Y. Zhang, N. Konz, K. Kramer, and M. A. Mazurowski, "Quantifying the limits of segmentation foundation models: Modeling challenges in segmenting tree-like and low-contrast objects. *arXiv preprint arXiv:2412.04243*," 2025.

[21] C. Zhou, C. C. Loy, and B. Dai, "Extract free dense labels from clip," in *Computer Vision – ECCV 2022*, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds. Cham: Springer Nature Switzerland, 2022, pp. 696–712.

[22] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "Rellis-3d dataset: Data, benchmarks and analysis," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 1110–1116.

[23] O. Alama, A. Bhattacharya, H. He, S. Kim, Y. Qiu, W. Wang, C. Ho, N. Keetha, and S. Scherer, "Rayfronts: Open-set semantic ray frontiers for online scene understanding and exploration," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025, pp. 5930–5937.

[24] C. Min, S. Si, X. Wang, H. Xue, W. Jiang, Y. Liu, J. Wang, Q. Zhu, Q. Zhu, L. Luo, F. Kong, J. Miao, X. Cai, S. An, W. Li, J. Mei, T. Sun, H. Zhai, Q. Liu, F. Zhao, L. Chen, S. Wang, E. Shang, L. Shang, K. Zhao, F. Li, H. Fu, L. Jin, J. Zhao, F. Mao, Z. Xiao, C. Li, B. Dai, D. Zhao, L. Xiao, Y. Nie, Y. Hu, and X. Li, "Autonomous driving in unstructured environments: How far have we come?, radiological, and nuclear disaster response. *arXiv preprint arXiv:2410.07701*," 2024.

[25] M. Eder, R. Prinz, F. Schöggl, and G. Steinbauer-Wagner, "Traversability analysis for off-road environments using locomotion experiments and earth observation data," *Robotics and Autonomous Systems*, vol. 168, p. 104494, 2023.

[26] T. Darcet, M. Oquab, J. Mairal, and P. Bojanowski, "Vision transformers need registers. *arXiv preprint arXiv:2309.16588*," 2023.

[27] J. Li, Y. Zhang, P. Yun, G. Zhou, Q. Chen, and R. Fan, "Roadformer: Duplex transformer for rgb-normal semantic road scene parsing," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 7, pp. 5163–5172, 2024.

[28] Y. Sun, W. Zuo, and M. Liu, "Rtfnet: Rgb-thermal fusion network for semantic segmentation of urban scenes," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2576–2583, 2019.

[29] H. Ye, J. Mei, and Y. Hu, "M2f2-net: Multi-modal feature fusion for unstructured off-road freespace detection," in *2023 IEEE Intelligent Vehicles Symposium (IV)*, 2023, pp. 1–7.

[30] Y. Lv, Z. Liu, G. Li, and X. Chang, "Noise-aware intermediary fusion network for off-road freespace detection," *IEEE Transactions on Intelligent Vehicles*, pp. 1–11, 2024.

[31] T. Ren, S. Liu, A. Zeng, J. Lin, K. Li, H. Cao, J. Chen, X. Huang, Y. Chen, F. Yan, Z. Zeng, H. Zhang, F. Li, J. Yang, H. Li, Q. Jiang, and L. Zhang, "Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*," 2024.

[32] IDEA-Research, "Grounded-sam-2: Ground and track anything in videos with grounding dino, florence-2, and sam 2," https://github.com/IDEA-Research/Grounded-SAM-2, 2025, accessed: 2025-04-30.

[33] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4104–4113.

[34] T. Moore and D. Stouch, "A generalized extended kalman filter implementation for the robot operating system," in *Intelligent Autonomous Systems 13: Proceedings of the 13th International Conference IAS-13*. Springer, 2016, pp. 335–348.

[35] J. Wang, M. Chen, N. Karaev, A. Vedaldi, C. Rupprecht, and D. Novotny, "Vggt: Visual geometry grounded transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 5294–5306.

[36] S. F. Bhat, R. Birkl, D. Wofk, P. Wonka, and M. Müller, "Zoedepth: Zero-shot transfer by combining relative and metric depth. . *arXiv preprint arXiv:2302.12288*," 2023.

[37] M. Tancik, E. Weber, E. Ng, R. Li, B. Yi, J. Kerr, T. Wang, A. Kristoffersen, J. Austin, K. Salahi, A. Ahuja, D. McAllister, and A. Kanazawa, "Nerfstudio: A modular framework for neural radiance field development," in *ACM SIGGRAPH 2023 Conference Proceedings*, ser. SIGGRAPH '23, 2023.

[38] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa, F. Massa, D. Haziza, L. Wehrstedt, J. Wang, T. Darcet, T. Moutakanni, L. Sentana, C. Roberts, A. Vedaldi, J. Tolan, J. Brandt, C. Couprie, J. Mairal, H. Jégou, P. Labatut, and P. Bojanowski, "DINOv3. *arXiv preprint arXiv:2508.10104*," 2025.

[39] G. Heinrich, M. Ranzinger, H. Yin, Y. Lu, J. Kautz, A. Tao, B. Catanzaro, and P. Molchanov, "Radiov2.5: Improved baselines for agglomerative vision foundation models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 22 487–22 497.

[40] P. Maheshwari, W. Wang, S. Triest, M. Sivaprakasam, S. Aich, J. G. R. III, J. M. Gregory, and S. Scherer, "Piaug – physics informed augmentation for learning vehicle dynamics for off-road navigation. *arXiv preprint arXiv:2311.00815*," 2023.