

# OVERSEEC: Open-Vocabulary Costmap Generation from Satellite Images and Natural Language

Rwik Rana<sup>1</sup>, Jesse Quattrociochi<sup>2</sup>, Dongmyeong Lee<sup>1</sup>, Christian Ellis<sup>2</sup>,  
Amanda Adkins<sup>1</sup>, Adam Uccello<sup>2</sup>, Garrett Warnell<sup>2</sup>, Joydeep Biswas<sup>1</sup>

<sup>1</sup>The University of Texas at Austin

<sup>2</sup>DEVCOM Army Research Laboratory

**Abstract**—Aerial imagery provides essential global context for autonomous navigation, enabling route planning at scales inaccessible to onboard sensing. We address the problem of generating global costmaps for long-range planning directly from satellite imagery when entities and mission-specific traversal rules are expressed in natural language at test time. This setting is challenging since mission requirements vary, terrain entities may be unknown at deployment, and user prompts often encode compositional traversal logic. Existing approaches relying on fixed ontologies and static cost mappings cannot accommodate such flexibility. While foundation models excel at language interpretation and open-vocabulary perception, no single model can simultaneously parse nuanced mission directives, locate arbitrary entities in large-scale imagery, and synthesize them into an executable cost function for planners. We therefore propose OVERSEEC, a zero-shot modular framework that decomposes the problem into *Interpret–Locate–Synthesize*: (i) an LLM extracts entities and ranked preferences, (ii) an open-vocabulary segmentation pipeline identifies these entities from high-resolution imagery, and (iii) the LLM uses the user’s natural language preferences and masks to synthesize executable costmap code. Empirically, OVERSEEC handles novel entities, respects ranked and compositional preferences, and produces routes consistent with human-drawn trajectories across diverse regions, demonstrating robustness to distribution shifts. This shows that modular composition of foundation models enables open-vocabulary, preference-aligned costmap generation for scalable, mission-adaptive global planning.

## I. INTRODUCTION

Long-range route planning for autonomous ground vehicles (AGVs) in off-road environments requires converting high-resolution aerial imagery into a planner-ready costmap. However, costmaps built on a fixed set of classes and a predefined set of traversal costs fail to adapt to mission-specific preferences, which generally have arbitrary classes.

Robust solutions exist for on-road settings, which benefit from fixed map ontologies - predefined sets of classes like road, lane marking, and sidewalk. Whereas, off-road settings present two major challenges: (i) adapting to new ontological elements, such as unencountered terrain types; (ii) adhering to complex user traversal rules and preferences (e.g., “prefer grass unless it borders a building”) [1]–[4]. Traditional perception models are limited by fixed ontologies in training data and thus cannot recognize novel entities at test time. Furthermore, traditional costmap generation pipelines, which rely on fixed class-to-cost mappings, do not capture the compositional and spatial logic from user preferences.

To address these challenges, we introduce OVERSEEC, a modular, open-vocabulary, zero-shot costmap generation

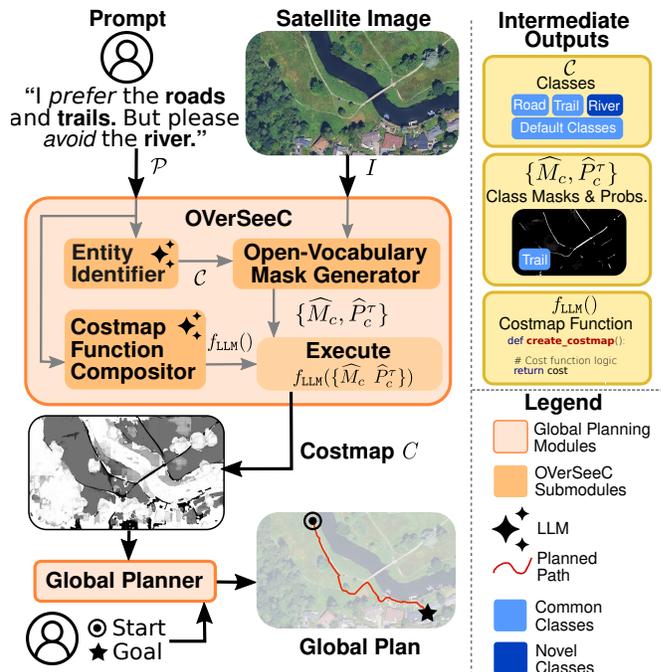


Fig. 1: **Overview of OVERSEEC**, which uses a satellite image  $I$  and a natural language prompt  $\mathcal{P}$  to generate a preference-aligned costmap  $C$  for global planning. The *Entity Identifier* and *Costmap Function Compositor* (Sec. IV-A, IV-C) use an LLM to extract relevant terrain classes  $\mathcal{C}$  and synthesize a cost function  $f_{LLM}(\cdot)$  respectively. The *Open-Vocabulary Mask Generator* (Sec. IV-B.1, IV-B.2) performs zero-shot semantic segmentation over  $I$ , yielding class masks  $\{\widehat{M}_c\}$  and thresholded probability maps  $\{\widehat{P}_c^T\}$ , where  $c \in \mathcal{C}$ . Finally  $f_{LLM}(\cdot)$  is executed to generate the final costmap  $C$ .

pipeline using aerial imagery and guided by natural language. This enables any standard global planner to compute a final route using the generated costmap. A single, end-to-end model is not suited to perform the distinct operations of interpreting compositional language and segmenting arbitrary entities from pixel data. Furthermore, the high resolution of satellite imagery prevents its direct processing by vision foundation models due to their fixed input-size constraints, demanding a specialized mechanism that can operate on the image at its native scale. OVERSEEC leverages a modular design to enable decomposition of the costmap generation problem into a logical sequence—*Interpret, Locate, and Synthesize*. Each stage is delegated to a specialized component: (i) a Large Language Model (LLM) for semantic entity interpretation; (ii) a perception pipeline to locate entities

within high-resolution imagery; and (iii) an LLM-driven code composer that generates a costmap function which maps the entities and the user’s compositional preferences to a costmap tailored to the mission.

Our work makes the following contributions:

- 1) We design a zero-shot perception pipeline for high-resolution satellite imagery that performs open-vocabulary segmentation while preserving native map resolution. This enables *locating* arbitrary, novel terrain classes at scale despite the fixed input-size constraints of segmentation models.
- 2) We demonstrate that a Large Language Model (LLM) can *interpret* entities and traversal rules from the user’s prompt, and *synthesize* executable costmap functions aligned to these preferences. This enables open-vocabulary, preference-aligned costmap generation directly from natural language.
- 3) We develop a GUI which enables rapid, zero-shot iteration: operators can modify entities or traversal preferences in natural language and obtain updated costmaps within minutes, without annotation, retraining, or dataset-specific supervision.
- 4) We propose the Ranked Regret Path Integral (RRPI), a metric for quantifying how well planned paths align with user preferences, enabling systematic evaluation of preference alignment.

## II. RELATED WORK

Generating costmaps from satellite imagery that adapt to novel terrain and natural language preferences requires integrating solutions from several domains: remote semantic scene understanding, preference interpretation, and flexible map representations. Prior work falls into three broad strategies. (i) Semantics-first fixed-ontology methods [5]–[9] assume a fixed ontology of terrain classes (e.g., road, water, building) and directly train a segmentation model to predict those classes, limiting adaptability to unseen entities or dynamic user preferences. (ii) Representation learning approaches [1]–[3] can learn complex functions but typically require extensive training data and offer limited interpretability. (iii) Modular open-vocabulary systems adapt to novel entities and instructions without retraining, and their staged design makes each step interpretable and debuggable, making them well-suited where labeled data is scarce. The modularity facilitates future component-wise upgrades. OVERSEEC aligns with this third strategy, using open-vocabulary VLMs [10]–[13] for text-prompted segmentation, prompt-based segmentation foundation models [14]–[17] for precise mask generation, and LLMs for code synthesis from natural language [18], [19].

Approaches such as Text2Seg [20] leverage CLIP-based [21] text embeddings for remote sensing segmentation with limited supervision. Our work extends these ideas by integrating open-vocabulary segmentation into a broader training-free costmap generation pipeline that also incorporates LLM-based preference composition, translating natural

language instructions directly into executable costmap logic, thereby creating an adaptable and interpretable framework.

## III. PROBLEM FORMULATION

Our objective is to synthesize a scalar-valued costmap  $C$  from a satellite image  $I$  based on a user’s natural language prompt  $\mathcal{P}$ , via a mapping function  $f$ , without requiring any task-specific training or extensive manual rule design. Formally, given  $I$  and  $\mathcal{P}$ ,

$$C = f(I, \mathcal{P}) \quad (1)$$

In this formulation,  $I \in \mathbb{R}^{H \times W \times 3}$  represents the input high-resolution RGB satellite image of dimensions  $H \times W$ .  $\mathcal{P}$  is the user’s natural language prompt, which can describe complex ontological and compositional preferences such as “go over the trail, but avoid the puddle”. The function  $f$  represents a mapping from  $I$  and  $\mathcal{P}$  as inputs to a scalar costmap  $C \in [0, 1]^{H \times W}$ , spatially aligned with  $I$ , where lower values indicate more desirable regions for traversal.

## IV. THE OVERSEEC ALGORITHM

We introduce OVERSEEC, a modular framework for open-vocabulary costmap generation. OVERSEEC is the realization of function  $f$  (Eq. 1) that maps a satellite image  $I$  and a natural language prompt  $\mathcal{P}$  to a final costmap  $C$ . This task demands: (i) *heterogeneous skills*—semantic parsing, open-vocabulary visual grounding at native resolution, and symbolic cost composition require different inductive biases; (ii) *zero-shot constraints*—the system would require task-specific supervision to handle unseen entities and compositional rules; and (iii) a *scale-aware perception* strategy to process high-resolution imagery.

While a number of frontier models touch on parts of this problem, neither a single model nor a trivial combination of them is sufficient. For instance, presenting a state-of-the-art VLM like ChatGPT-4o with the image and prompt fails because it fails to produce a quantitative, pixel-aligned cost grid required by a planner. Thus, we decompose  $f$  into a three-stage sequence *Interpret–Locate–Synthesize* (Fig. 1) :

- (a) **Entity Identification** (Sec. IV-A): natural language prompts are free-form, so a fixed label set of terrain/object classes cannot be assumed; the resulting class set is open-ontology. Each prompt  $\mathcal{P}$  must be parsed to construct the relevant class set  $\mathcal{C}$ . Thus, an LLM is used to generate  $\mathcal{C}$  which is then used for segmentation of  $I$ .
- (b) **Open-Vocabulary Mask Generation** (Sec. IV-B): Locating these arbitrary classes  $\mathcal{C}$  in  $I$  requires a *language-grounded* segmentation model that adapts to open-ontology prompts, followed by refinement to sharpen boundaries and preserve connectivity. This stage produces per-class refined binary masks  $\{\widehat{M}_c\}$  and probability maps  $\{\widehat{P}_c^r\}$  for each  $c \in \mathcal{C}$ , using tiled inference with blending to handle high-resolution  $I$ .
- (c) **Costmap Function Composition** (Sec. IV-C): Fixed mappings cannot capture conditional or compositional preferences. Because the cost function must adhere to

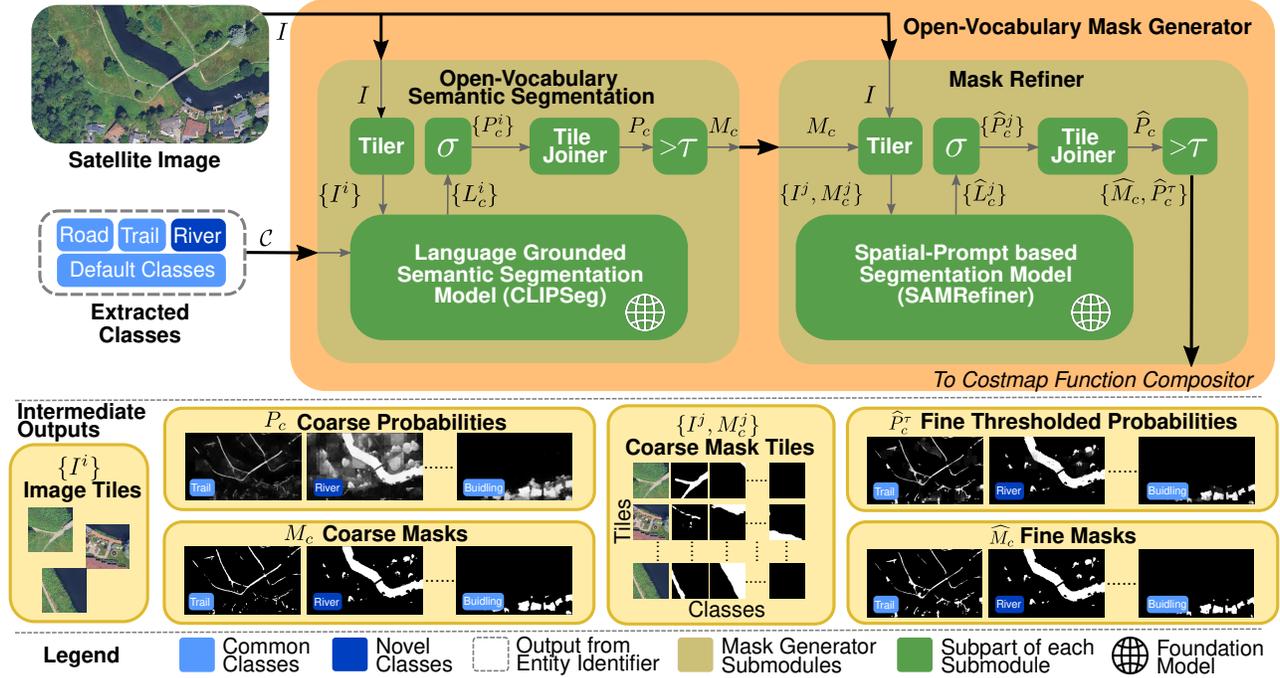


Fig. 2: **Open-Vocabulary Mask Generator** (Sec. IV-B). Given a satellite image  $I$  and extracted classes  $\mathcal{C}$ , the pipeline comprises two submodules: (i) *Open-Vocabulary Semantic Segmentation* (Sec. IV-B.1), which produces per-class probability maps  $P_c$  and coarse masks  $M_c$  for open-ontology classes; and (ii) *Mask Refinement* (Sec. IV-B.2), which refines them into fine probabilities  $\hat{P}_c$  and masks  $\hat{M}_c$ .

$\mathcal{P}$ , it must be generated on the fly for each prompt query. Thus an LLM synthesizes a costmap function  $f_{\text{LLM}}$  that encodes the prompt as executable logic over  $\{M_c\}$  and  $\{\hat{P}_c^\tau\}$ , with spatial predicates.

- (d) Finally, we execute  $f_{\text{LLM}}$  using  $\{\hat{M}_c\}$  and  $\{\hat{P}_c^\tau\}$  to yield a scalar costmap  $C \in [0, 1]^{H \times W}$ .

### A. Entity Identification

This first stage must parse the user’s natural language prompt ( $\mathcal{P}$ ) to identify all relevant semantic classes. Simpler methods like keyword extraction fail to handle the semantic complexity of natural language (e.g., synonyms or implicit intent), whereas an LLM can reliably identify novel classes outside any fixed ontology.

This stage also accounts for the geometric heterogeneity of these classes used for creating masks from segmentation models. Segmentation models output continuous per-pixel logit activations which are binarized to form class masks using a binarization threshold. However, activations for thin, linear structures are lower than for broad areal regions. A single high threshold fragments linear features while low values admit noise in areal masks. Consequently, classes must be grouped by geometry and assigned group-specific thresholds.

Thus, an LLM parses  $\mathcal{P}$  and identifies a list of classes  $\mathcal{C}$ , which are then merged with a default set  $\mathcal{C}_{\text{default}}$  for robust coverage. The LLM categorizes each class as either *linear* or *areal*. This distinction allows us to apply class-specific thresholds in the mask generation step (Sec. IV-B).

### B. Open-Vocabulary Mask Generation

Generating class-specific masks from high-resolution satellite imagery  $I$  presents three challenges: (i) obtaining per-class semantic segmentation from open-ontology classes; (ii) refining segmentation model outputs, which generally have soft edges, speckle, and broken connectivity; and (iii) adapting to high-resolution imagery and variable image dimensions, since segmentation models accept fixed, relatively small input sizes. For (i) and (ii) we use task-specific foundation models. For (iii), we tile the images into smaller chunks and feed them to the specific models.

1) *Open-Vocabulary Semantic Segmentation*: The first stage of our perception pipeline generates initial, coarse masks  $\{M_c\}$  for each terrain category  $c \in \mathcal{C}$  using a **language-grounded segmentation model (LGSM)**. This allows the system to identify arbitrary classes specified in the prompt at test time. To process the high-resolution satellite image  $I$  without losing detail through downsampling, the model operates on smaller, overlapping tiles  $\{I^i\}$ . For each tile, LGSM produces a per-pixel probability map  $P_c^i$  for every relevant class. These individual maps are then stitched together by averaging the predictions in the overlapping regions, resulting in a single, full-resolution probability maps  $\{P_c\}$  for the entire area. These maps are then binarized using the class-specific (*linear* or *areal*) thresholds to produce the coarse masks  $\{M_c\}$ .

2) *Mask Refinement*: These coarse masks  $\{M_c\}$  often suffer from imprecise boundaries and segmentation artifacts. The second stage of the pipeline therefore refines them. This process also operates on a tiled basis, providing a **spatial**

**prompt-based segmentation model (SPSM)** with both the original image tiles  $\{I^j\}$  and the corresponding coarse masks  $\{M_c^j\}$ . The coarse mask acts as a strong spatial prior, guiding the SPSM to produce a significantly sharper and more accurate probability map for each tile. These refined per-tile maps are joined to form the final probability maps  $\{\hat{P}_c\}$  and its corresponding binary masks  $\{\widehat{M}_c\}$ . This final mask is then used to gate the probability map, yielding the thresholded probability maps  $\{\hat{P}_c^\tau\}$  used in the final cost composition stage.

### C. Costmap Function Composition

Conventional costmap pipelines rely on a fixed class-to-cost lookup table, an approach that fails to capture conditional preferences, geometric preferences, and adapt to unseen object classes. While a simpler approach could use an LLM to generate weights for a fixed template, such a method cannot express the compositional or spatial logic inherent in user commands (e.g., “prefer grass unless it borders a building”). We therefore propose using an LLM to synthesize executable code on-the-fly in the form of a costmap generation function,  $f_{\text{LLM}}$ . This ensures generality and zero-shot compositionality. The LLM translates the user’s natural language prompt  $\mathcal{P}$  into an executable function using the following instructions:

- i. **Function signature:** Synthesize an executable function  $f_{\text{LLM}}: (\{\widehat{M}_c\}, \{\hat{P}_c^\tau\}) \rightarrow C \in [0, 1]^{H \times W}$ .
- ii. **Mask operators:** Predefined operators are used for basic pixel-wise mask transformations—AND, OR, NOT, and REMOVE (i.e.,  $\text{REMOVE}(A, B) = A \wedge \neg B$ ).
- iii. **Prompt analysis:** The prompt  $\mathcal{P}$  is analyzed to (i) assign class weights  $\{w_c\}$  (lower weight indicates higher preference); (ii) detect semantic hierarchies  $\mathcal{H}$  (e.g., “baseball field”  $\subset$  “grass”); and (iii) infer geometric cues  $\{\gamma_c\}$  that map spatial language (e.g., “near the road”) to mask transforms.
- iv. **Mask operations:** An unassigned mask  $U \in \{0, 1\}^{H \times W}$  is formed. Hierarchies  $\mathcal{H}$  are enforced by removing subset masks from parent masks to yield  $\{\widehat{M}_c^{\mathcal{H}}\}$ . Geometric cues  $\{\gamma_c\}$  are applied to transform  $\{\widehat{M}_c^{\mathcal{H}}\}$  to geometric-aware masks  $\{\widehat{M}_c^G\}$ .
- v. **Cost accumulation:** Per-class contributions are computed as  $w_c \cdot \widehat{M}_c^G \cdot \hat{P}_c^\tau$ . The contributions are summed pixel-wise to obtain  $\tilde{C}$ . Pixels in  $U$  are set to  $C_{\max} = \max \tilde{C}$ , yielding  $C_{\text{un-normalized}}$ .
- vi. **Normalization:**  $C_{\text{un-normalized}}$  is normalized to  $[0, 1]$  to yield final costmap  $C$ .

## V. IMPLEMENTATION DETAILS

### A. Entity Identification & Costmap Function Composition

We use `gemma-2-27b-it` [22] as the LLM for both the entity identification (Sec. IV-A) and cost synthesis (Sec. IV-C) stages. This model was chosen for its proficiency in the two capabilities required by our pipeline: (i) structured reasoning, where it accurately parses user prompts to identify and categorize entities, and (ii) robust code-generation, where

it synthesizes executable Python functions to implement the user’s compositional preferences.

### B. Open-Vocabulary Semantic Segmentation

For the initial coarse mask generation, we select CLIPSeg [10] as our LGSM. Its architecture is well-suited for this role, as its direct skip connections to the CLIP backbone provide strong localization for arbitrary text prompts, making it an effective open-vocabulary “first-pass” segmenter. As described in Sec. IV-B.1, we employ distinct thresholds for binarization. The values for linear features ( $\tau_L = 0.4$ ) and areal features ( $\tau_A = 0.8$ ) were chosen empirically to provide the best trade-off between preserving feature connectivity and minimizing noise across our validation datasets.

### C. Mask Refinement

For the second stage of our perception pipeline, we use SAMRefiner [17], a variant of the Segment Anything Model (SAM) [14], as our SPSM. This choice is deliberate: SAM is a class-agnostic, prompt-based segmenter, making it the ideal tool for refining a mask when given a strong spatial prior. In our pipeline, the coarse semantic mask from CLIPSeg provides this prior.

Given the tile mask  $M_c^{(i)}$  for tile  $I^{(i)}$  as input, SAMRefiner internally applies a multi-prompt strategy: (i) it heuristically selects foreground and background points, (ii) incorporates the coarse mask as a Gaussian-style prior, and (iii) uses context-aware elastic bounding boxes. These prompts are passed to SAM, which generates refined masks.

### D. Baselines

To benchmark against conventional semantic segmentation approaches, we use 2 fixed-ontology baselines: (i) SegFormer-B5 [8]; (ii) DINO-UNet, which combines a frozen ViT-DINO encoder [23] with a lightweight UNet decoder. We fine-tune SegFormer and train the DINO-UNet Decoder on a dataset  $\mathcal{D}_1$  curated from OpenStreetMap [24]. It consists of image patches of size  $512 \times 512$  and in total has 6000 images.

## VI. EXPERIMENTS AND RESULTS

Our evaluation is designed to answer three core research questions (RQs):

- 1) **Alignment and Comparative Performance:** How well do costmaps from OVERSEEC align with ground-truth semantic preferences, and how effectively do they guide planners to low-cost regions compared to state-of-the-art methods?
- 2) **Novel-Class Generalization:** Can the zero-shot pipeline accurately segment and assign traversal costs to terrain categories mentioned in natural language prompts but absent from the supervised training ontology?
- 3) **Robustness to Distribution Shift:** How well does the system maintain its segmentation accuracy and downstream planning performance with varying geographic regions or other visual domain shifts?

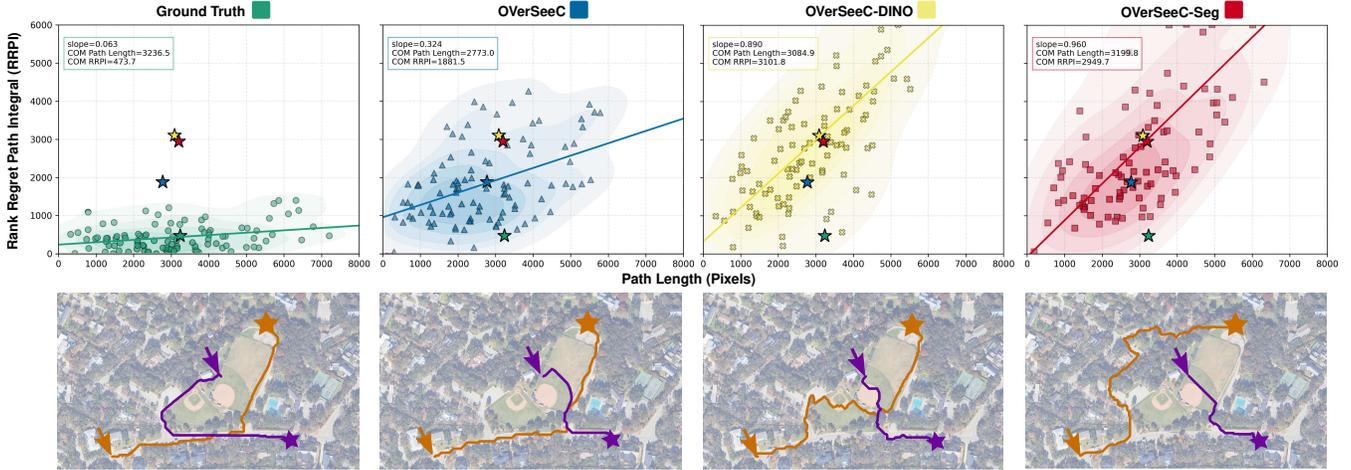


Fig. 3: **Planning results for the  $\mathcal{D}_2$ -OOD-OV scenario:** Comparison of costmap alignment using RRPI (Sec. VI-B.1) metric under the user preference: “Prefer the roads and trails, grass should be fine, try to avoid the baseball field as much as possible.” The class ranking used are: road: 1, trail: 1, grass: 2, baseball field : 3, tree: 4, building: 5. The top row shows RRPI vs. path length scatter plots with KDE contours; the colored pointers in these plots indicates the COM of the KDE, and the solid line represents a linear regression fit. A lower slope for this line is preferable, as it indicates that the RRPI score remains low even as path length increases. The bottom row shows a subset of these trajectories generated from Dijkstra’s algorithm overlaid on the map (start: arrow, goal: star).

### A. Experimental Setup

1) *Baselines:* In all the experiments henceforth, to compare against the baselines, we will replace the LGSM i.e., CLIPSeg with baseline fixed-ontology models i.e., SegFormer and DinoUNet, keeping all other components of OVERSEEC as is. We name them OVERSEEC-Seg and OVERSEEC-Dino respectively. We use Dijkstra’s algorithm to plan paths on the costmaps.

#### 2) Evaluation Environments:

We evaluate OVERSEEC and baseline methods on two datasets,  $\mathcal{D}_2$  and  $\mathcal{D}_3$ .  $\mathcal{D}_2$  is designed for quantitative comparison with baselines, assessing ontological and compositional preference adherence. Ground-truth (GT) semantic maps are created by manually annotating satellite images at 0.1 m/pixel resolution, and the dataset includes In-Distribution (ID), Out-of-Distribution (OOD), and OOD with Open-Vocabulary (OOD-OV) splits.  $\mathcal{D}_3$  provides a more challenging setting focused on compositional preferences and human intent alignment, also at 0.1 m/pixel resolution, and consists only of OOD and OOD-OV cases. For the human case study, maps  $\mathcal{D}_3$ -HE<sub>1</sub> and  $\mathcal{D}_3$ -HE<sub>2</sub> each use two distinct prompts, while  $\mathcal{D}_3$ -HE<sub>3</sub> and  $\mathcal{D}_3$ -HE<sub>4</sub> use three prompts each. Further details about both datasets are summarized in Table I.

### B. Evaluation Metrics

#### 1) Ranked Regret Path Integral (RRPI):

Quantifying the alignment of a generated costmap with user preferences is challenging, as defining an “ideal” cost function directly from natural language is often intractable. However, it is generally more straightforward to establish a preference-ordered ranking of terrain types based on a given natural language description. For instance, if a user states, “trails are good, grass is okay, avoid water,” we can assign ranks: trail (rank 1), grass (rank 2), water (rank 3), etc.

We introduce the **Ranked Regret Path Integral (RRPI)** score. Given a user preference  $\mathcal{P}$ , we first derive a rank

Map Name	Objective
<b>Dataset: <math>\mathcal{D}_2</math></b>	
$\mathcal{D}_2$ -ID <sub>1</sub>	ID setting with familiar regions and fixed ontology from baseline training.
$\mathcal{D}_2$ -ID <sub>2</sub>	ID setting with familiar regions and fixed ontology from baseline training.
$\mathcal{D}_2$ -OOD	OOD region with fixed ontology.
$\mathcal{D}_2$ -OOD-OV	OOD region with prompt mentioning ‘baseball field’ requiring OV generalization and understanding its relation to ‘grass’.
<b>Dataset: <math>\mathcal{D}_3</math></b>	
$\mathcal{D}_3$ -HE <sub>1</sub>	Recognizing novel class ‘electric tower’.
$\mathcal{D}_3$ -HE <sub>2</sub>	Differentiating ‘railway track’ from roads.
$\mathcal{D}_3$ -HE <sub>3</sub>	OOD region with prompt mentioning ‘sports fields’ requiring OV generalization and understanding its relation to ‘grass’.
$\mathcal{D}_3$ -HE <sub>4</sub>	Recognizing novel class ‘river’ and its relationship to ‘water’.

TABLE I: Evaluation settings across environment types.

**ID:** In-distribution (same domain as supervised training  $\mathcal{D}_1$ ). **OOD:** Out-of-distribution. **OV:** Open-vocabulary.

mapping  $R(c)$  for each relevant semantic class  $c$ , where  $R(c) \in \{1, 2, \dots, N_c\}$  and  $N_c$  is the number of distinct classes. A lower rank indicates a more preferred terrain type, with 1 being the lowest. The *rank regret* for traversing a pixel of class  $c$  is  $R(c) - 1$ . This value penalizes less preferred terrain; the most preferred has zero regret.

For any given path  $\tau = [(x_1, y_1), (x_2, y_2), \dots, (x_L, y_L)]$  of length  $L$  (i.e., the trajectory covers  $L$  pixels), through a semantic map  $S$  (where  $S(x_i, y_i)$  is the semantic class of the pixel at  $(x_i, y_i)$ ), the RRPI score is calculated as :

$$\text{RRPI}(\tau, S, R) = \sum_{(x,y) \in \tau} (R(S(x,y)) - 1)$$

where  $R(S(x,y))$  is the rank of semantic class of the pixel  $(x,y)$ . The RRPI score does not inherently account for path length. To provide a more nuanced evaluation, we analyze path characteristics across two dimensions: path length (in pixels) and the RRPI score ( Fig. 3). We evaluate planner

performance by sampling multiple start–goal pairs within the same map and preference prompt. Each planned trajectory yields a (distance, RRPI) pair, producing a set of scatter points that characterize how the planner performs under the costmap conditioned on the given image and prompt. We fit a Gaussian Kernel Density Estimate (KDE) to these scatter points. The Center of Mass (COM) of this KDE blob yields an aggregate (distance, RRPI) pair that represents the performance of the method in that specific environment for the given preference prompt. For both the distance and RRPI components of the COM, lower values are considered *better*.

We calculate the RRPI scores of 50 start and goal point pairs drawn uniformly at random over the whole image for each method within each map of dataset  $\mathcal{D}_2$ .

### 2) Segmentation Accuracy Metrics:

We report Intersection-over-Union (IoU) on the *stitched* maps, comparing each method’s per-class masks to hand-drawn ground-truth labels from dataset  $\mathcal{D}_2$ .

### 3) Human Case Study:

We conducted a human case study on maps from  $\mathcal{D}_3$  dataset, each with multiple prompts of varying complexity. Three annotators per prompt sketched start-to-goal trajectories that best satisfied the instructions, providing behavioral references. Alignment is quantified by an *mean* Hausdorff distance between a path  $\tau_{\text{sys}}$  generated by the system and the union of the three human paths  $\bigcup_i \tau_{h_i}$ ,

$$\text{HD}(\tau_{\text{sys}}) = \frac{1}{|\tau_{\text{sys}}|} \sum_{p \in \tau_{\text{sys}}} \min_{q \in \bigcup_i \tau_{h_i}} \|p - q\|_2,$$

then normalized by the map diagonal  $\sqrt{H^2 + W^2}$  for cross-map comparison. *Lower* values indicate *closer* adherence to the human-preferred paths.

## C. Evaluation

Thus, we address each RQ, presenting dedicated quantitative and qualitative evidence.

### 1) RQ1: Alignment and Comparative Performance:

We compare OVERSEEC to fixed-ontology baselines in  $\mathcal{D}_2$  in-distribution (ID) environments. Table II shows that OVERSEEC achieves competitive or better RRPI scores and path lengths, indicating strong comparative performance.

Table III quantifies alignment with human-drawn trajectories across four HE maps with multiple prompts. We report the mean Hausdorff distance averaged over prompts per map, and in every case OVERSEEC yields the lowest distance, showing closest agreement with human intent. This is reinforced by our prompt-sensitivity analysis (Table IV), which compares OVERSEEC’s trajectories against human-drawn references under different prompts on the same map. Low diagonal values indicate close alignment with the prompt, while larger off-diagonal distances show that different instructions lead to distinct paths rather than a generic solution.

Figure 4 further illustrates the generated costmaps adhere to geometric preferences. For example, when the prompt specifies “stay on the center of the road” (b), the lowest cost is along the centerline, whereas with “stay on the side of the

Map	Method	RRPI ↓	Path Length ↓
$\mathcal{D}_2\text{-ID}_1$	Ground Truth	889.2	3682
	OVERSEEC-DINO	2613.3	4749
	OVERSEEC-Seg	2424.6	4837
	OVERSEEC	<b>2379.9</b>	<b>4008</b>
$\mathcal{D}_2\text{-ID}_2$	Ground Truth	634.3	2452
	OVERSEEC-DINO	2214.2	4199
	OVERSEEC-Seg	<b>1923.8</b>	<b>3765</b>
	OVERSEEC	2118.2	3998

TABLE II: **RQ1** - Comparing performances of OVERSEEC and baselines using RRPI and total path lengths in ID environments. Best results among methods are in **bold**.

Map	OVERSEEC-DINO	OVERSEEC-Seg	OVERSEEC
$\mathcal{D}_3\text{-HE}_1$	0.258	0.089	<b>0.016</b>
$\mathcal{D}_3\text{-HE}_2$	0.027	0.073	<b>0.021</b>
$\mathcal{D}_3\text{-HE}_3$	0.072	0.069	<b>0.025</b>
$\mathcal{D}_3\text{-HE}_4$	0.090	0.051	<b>0.026</b>

TABLE III: **RQ1** — Prompt-level alignment with human-drawn trajectories. The table reports the average of mean Hausdorff distance between human-drawn and generated trajectories for across all prompts per HE map. Lower values indicate better alignment.

$\mathcal{D}_3\text{-HE}_1$	Human		$\mathcal{D}_3\text{-HE}_4$	Human			
	A	B		C	D		
OVERSEEC	A	<b>16.28</b>	172.51	OVERSEEC	C	<b>163.78</b>	892.97
	B	256.26	<b>54.92</b>		D	351.13	<b>151.92</b>

TABLE IV: **RQ1** — Prompt sensitivity between human and OVERSEEC (Hausdorff distance, px); low diagonal values show alignment with the intended prompt. Prompts: A = ‘avoid electric tower’, B = ‘can go under electric tower’, C = ‘avoid river’, D = ‘river is dry’.

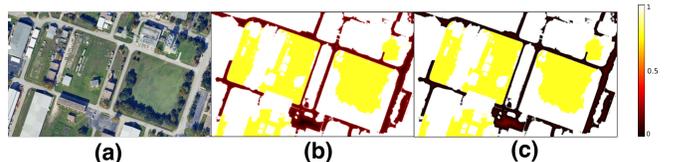


Fig. 4: **RQ1** — OVERSEEC’s alignment to geometric preferences. (a) is the satellite image, (b) costmap for the prompt “stay on the center of the road,” (c) costmap for the prompt “stay on the side of the road,” assigning low cost to the road edges. In both cases, the cost bar ranges from 0 (lowest cost) to 1 (highest).

road” (c), the lowest cost shifts to the edges. In both cases, roads as a whole remain low-cost regions, confirming that OVERSEEC can encode fine-grained geometric rules while preserving the broader semantics.

### 2) RQ2: Novel-Class Generalization:

The key capability of OVERSEEC is its ability to handle terrain classes that are unknown at deployment. We evaluate this in out-of-distribution, open-vocabulary (OOD-OV) settings, where prompts contain entities absent from the baselines’ training data. As shown in Table V, OVERSEEC achieves significantly lower RRPI than the supervised baselines. In the OOD-OV case, the baselines cannot reason about novel classes and therefore disregard the corresponding parts of the prompt, missing crucial preference information. OVER-

Map	Method	RRPI ↓	Path Length ↓
$\mathcal{D}_2$ -OOD	Ground Truth	249.1	3419
	OVERSEEC-DINO	4372.4	5583
	OVERSEEC-Seg	2044.4	<b>4085</b>
	OVERSEEC	<b>1573.8</b>	4351
$\mathcal{D}_2$ -OOD-OV	Ground Truth	473.7	3236.5
	OVERSEEC-DINO	3103.8	3084.9
	OVERSEEC-Seg	2949.7	3199.8
	OVERSEEC	<b>1881.5</b>	<b>2773</b>

TABLE V: **RQ2 and RQ3 — Novel-Class Generalization:** OVERSEEC outperforms baselines in OOD and OOD-OV. Qualitative analysis for OOD-OV can be found in Fig. 3

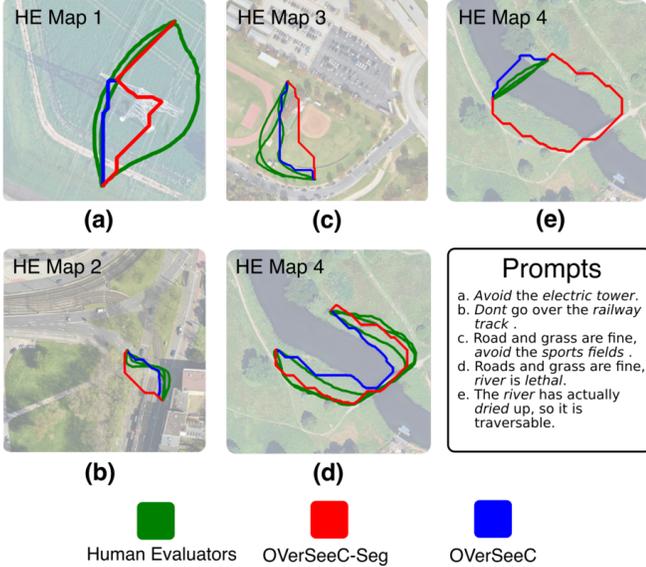


Fig. 5: **RQ2** — Qualitative results from the human case study experiments, all in open-vocabulary settings (see Table V). Each example corresponds to the scenarios in Table III, showing that OVERSEEC adapts to novel categories and contextual prompt semantics. Across these OV examples, the trajectories produced by OVERSEEC are qualitatively closest to the human-drawn references, demonstrating strong alignment with operator intent.

SEEC, however, parses these entities and integrates them into the costmap, producing trajectories that reflect user intent. Figure 5 illustrates this capability: OVERSEEC correctly avoids a novel electric tower, and adapts its path depending on whether a river is described as lethal or traversable, whereas baselines fail to distinguish. These results show that OVERSEEC generalizes to open-ontology entity sets.

### 3) **RQ3: Robustness to Distribution Shift:**

Finally, we evaluate how well OVERSEEC maintains performance when encountering geographic regions visually distinct from the training data of the supervised baselines. In these OOD settings (Table V), OVERSEEC again produces paths with lower regret, indicating its robustness to domain shift. In the OOD case, this improvement stems from the fact that fixed-ontology baselines fail to detect even known classes when appearance shifts across regions, weather, or lighting, leading to trajectories that ignore critical terrain. By contrast, OVERSEEC leverages a language-grounded foundation model (CLIPSeg) for segmentation, trained on

Method	Tree	Grass	Building	Water	Road/Trails
OVERSEEC-DINO <sup>-</sup>	0.298	0.084	0.224	0.335	0.281
OVERSEEC-DINO	0.346	0.105	0.212	0.210	0.359
OVERSEEC-Seg <sup>-</sup>	0.410	0.544	0.398	<b>0.802</b>	0.403
OVERSEEC-Seg	0.392	0.289	0.350	0.435	0.482
OVERSEEC <sup>-</sup>	<b>0.682</b>	<b>0.581</b>	0.640	0.697	0.543
OVERSEEC	0.623	0.517	<b>0.644</b>	0.665	<b>0.569</b>

TABLE VI: **RQ3 — Segmentation Robustness:** OVERSEEC maintains high segmentation quality (IoU) under distribution shifts. Method<sup>-</sup> denotes the pipeline without SAM refinement, highlighting its importance for linear features like roads.

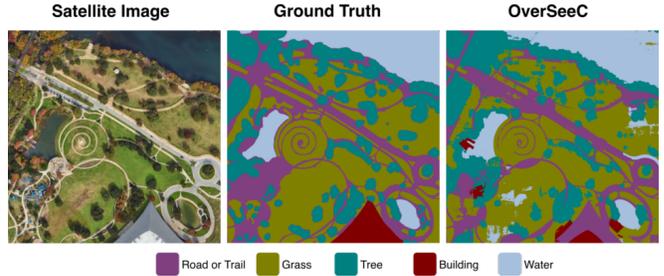


Fig. 6: **RQ3** — OVERSEEC’s segmentation output provides a reliable foundation for planning.

massive and diverse data, enabling it to consistently recognize entities, robust to such distributional shifts. This is supported by its strong underlying segmentation performance on the  $\mathcal{D}_2$  dataset (Table VI), achieving higher IoU scores than the baselines on all but one class, with the largest gains observed for linear features critical to navigation.

### D. Interactive GUI for Rapid Iteration

To demonstrate OverSeeC’s usability as an operator-facing tool, we develop a graphical user interface (GUI) (Fig. 7). The GUI showcases how operators can rapidly prototype and refine costmaps in a zero-shot manner by iterating with natural language instructions rather than retraining models. This design highlights OverSeeC’s modularity and interpretability: entity masks can be generated once and reused, preferences can be updated through re-prompting or direct edits, and costmaps can be quickly validated through standard planners. Together, the GUI emphasizes that mission-specific costmaps can be created, inspected, and corrected within minutes, enabling practical deployment and fast operator-in-the-loop adaptation.

## VII. LIMITATIONS AND FUTURE WORK

While OVERSEEC shows strong results in zero-shot, preference-aligned costmap generation, several aspects warrant refinement. First, tighter integration between segmentation and mask refinement—e.g., through shared embeddings or joint optimization—could improve efficiency and consistency. Second, reliance on the LLM for nuanced semantic hierarchies may miss complex relationships; graph-based reasoning [25] could strengthen handling of inter-related or multi-label classes. Finally, robustness to visual artifacts (e.g., shadows) and occlusions is limited, and incorporating contextual cues or generative inpainting may yield more coherent costmaps for planning.

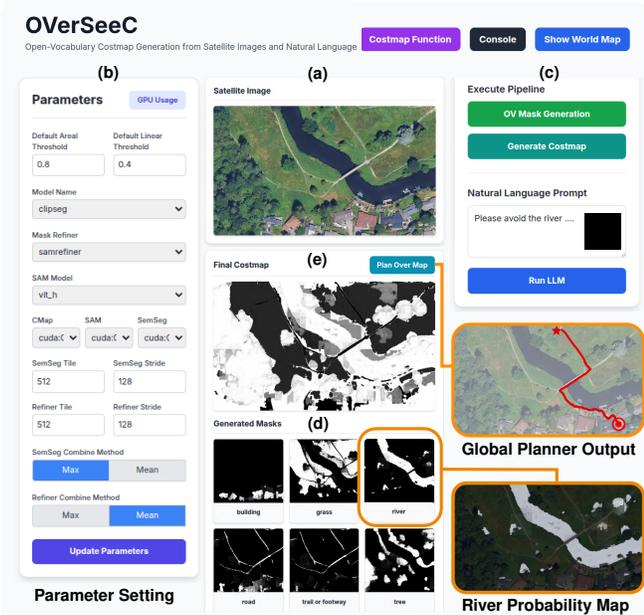


Fig. 7: **OverSeeC GUI** — A modular interface that enables rapid, zero-shot iteration. (a) Users upload a satellite image or extract imagery via the map tool. (b) Parameters can be adjusted and compute distributed across GPUs for fast inference. (c) Natural language prompts are processed by the LLM to generate classes of interest and a costmap function. (d) The open-vocabulary mask generator produces class-specific masks. (e) The finalized costmap is generated and used with planners such as A\* or Dijkstra.

## VIII. CONCLUSION

We present OVERSEEC, a modular and zero-shot architecture for generating costmaps from aerial imagery using natural language preferences, addressing the critical need for adaptability in off-road navigation without requiring fine-tuning. By leveraging language-grounded segmentation, mask refinement, and LLM-driven preference interpretation, OVERSEEC enables rapid adaptation to new classes and compositional instructions. Empirical evaluations demonstrate its high adaptability, successful generalization to novel terrains and preferences, and superior performance over baselines in challenging out-of-distribution and open-vocabulary scenarios. This work highlights the potential of combining large-scale pre-trained models in neuro-symbolic frameworks for creating adaptable, user-centric robotic navigation systems.

## REFERENCES

- [1] Kasi Viswanath, Felix Sanchez, Timothy Overbye, Jason M. Gregory, and Srikanth Saripalli. Trailblazer: Learning offroad costmaps for long range planning, 2025.
- [2] Luisa Mao, Garrett Warnell, Peter Stone, and Joydeep Biswas. pacer: Preference-conditioned all-terrain costmap generation. *IEEE Robotics and Automation Letters*, 10(5):4572–4579, 2025.
- [3] Kavan Singh Sikand, Sadeh Rabiee, Adam Uccello, Xuesu Xiao, Garrett Warnell, and Joydeep Biswas. Visual representation learning for preference-aware path planning. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 11303–11309, 2022.
- [4] Nan Wang, Xiang Li, Kanghua Zhang, Jixin Wang, and Dongxuan Xie. A survey on path planning for autonomous ground vehicles in unstructured environments. *Machines*, 12(1), 2024.

- [5] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.
- [6] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.
- [7] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
- [8] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Neural Information Processing Systems (NeurIPS)*, 2021.
- [9] Waleed Abdulla. Mask r-cnn for object detection and instance segmentation on keras and tensorflow. [https://github.com/matterport/Mask\\_RCNN](https://github.com/matterport/Mask_RCNN), 2017.
- [10] Timo Lüddecke and Alexander Ecker. Image segmentation using text and image prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7086–7096, June 2022.
- [11] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *European Conference on Computer Vision (ECCV)*, 2022.
- [12] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *International Conference on Learning Representations*, 2022.
- [13] Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, Nanyun Peng, Lijuan Wang, Yong Jae Lee, and Jianfeng Gao. Generalized decoding for pixel, image, and language, 2022.
- [14] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023.
- [15] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9796–9805, 2020.
- [16] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011.
- [17] Yuqi Lin, Hengjia Li, Wenqi Shao, Zheng Yang, Jun Zhao, Xiaofei He, Ping Luo, and Kaipeng Zhang. Samrefiner: Taming segment anything model for universal mask refinement. *ArXiv*, abs/2502.06756, 2025.
- [18] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9493–9500, 2023.
- [19] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, and Co. Authors. Palm-e: An embodied multimodal language model, 2023.
- [20] Jieliu Zhang, Zhongliang Zhou, Gengchen Mai, Mengxuan Hu, Zihan Guan, Sheng Li, and Lan Mu. Text2seg: Remote sensing image semantic segmentation via text-guided visual foundation models, 2024.
- [21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020, 2021.
- [22] Gemma Team and Co. Authors. Gemma: Open models based on gemini research and technology, 2024.
- [23] Maxime Oquab, Timothée Darcet, Moutakanni, and Co. Authors. Dinov2: Learning robust visual features without supervision, 2023.
- [24] OpenStreetMap contributors. Planet dump retrieved from <https://planet.osm.org>, 2017.
- [25] Liulei Li, Tianfei Zhou, Wenguan Wang, Jianwu Li, and Yi Yang. Deep hierarchical semantic segmentation. *arXiv preprint arXiv:2203.14335*, 2022.