

Article

Unified UAV Open-Vocabulary Semantic Segmentation: Benchmark Construction and LLM-Guided Text–Visual Enhancement

Kun Wang, Wei Li, Xiaopeng Liu *, Duping Huang and Jiale Yang

Air Force Engineering University, Xi'an 710043, China; fed_wang01@163.com (K.W.); wei_li109@163.com (W.L.); dup_huang@163.com (D.H.)

* Correspondence: afeulxp@163.com

Abstract

Open-vocabulary semantic segmentation (OVSS) of unmanned aerial vehicle (UAV) imagery aims to recognize arbitrary text-specified categories in aerial scenes, but existing OVSS models often suffer from UAV-domain shifts. To provide a reproducible testbed, this paper constructs a unified UAV OVSS benchmark by reorganizing multiple UAV segmentation datasets into cross-dataset transfer settings with explicit category harmonization and seen/unseen vocabulary analysis. Based on this benchmark, we propose UAV-OVSeg, a Cost Aggregation-style dense matching framework enhanced in two complementary directions: an LLM-guided Category Expansion Module that converts raw category names into structured UAV-aware descriptions, and a DINO-enhanced Geometric Feature Fusion Module that injects local structure into dense visual–text matching. Under SynDrone training, UAV-OVSeg achieves 65.5% mean intersection over union (mIoU) and 78.0% mean accuracy (mACC), improving upon the CAT-Seg baseline by 2.6 mIoU and 3.2 mACC. Under Aeroscapes training, it achieves 64.5% mIoU and 76.8% mACC, improving upon CAT-Seg by 2.6 mIoU and 3.1 mACC. Additional analyses of LLM variants, object scales, prompt sensitivity, boundary quality, and computational cost further verify the effectiveness and reproducibility of the proposed UAV-oriented text–visual enhancement strategy.

Keywords: unmanned aerial vehicle; open-vocabulary semantic segmentation; remote sensing; geospatial artificial intelligence; vision–language model; large language model; benchmark construction; text–visual enhancement



Academic Editors: Yang Zhan, Xiuyuan Zhang, Ailong Ma, Qiqi Zhu and Xiaoyan Lu

Received: 28 May 2026

Revised: 9 July 2026

Accepted: 10 July 2026

Published: 20 July 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Semantic segmentation of unmanned aerial vehicle (UAV) imagery aims to assign a semantic label to each pixel in high-resolution aerial scenes. As an important form of remote sensing observation, UAV imagery provides flexible acquisition, fine spatial details, and wide-area geospatial perception, and has been widely used in urban planning, disaster monitoring, traffic surveillance, precision agriculture, environmental assessment, and infrastructure inspection [1,2]. Accurate pixel-level interpretation transforms raw UAV observations into structured geospatial semantic maps, which are essential for downstream decision-making and intelligent Earth observation systems.

Despite the progress of deep learning-based UAV segmentation, most existing methods still follow a closed-set recognition paradigm, where the model can only predict categories predefined during training [3–7]. This setting is restrictive for real-world remote sensing and geospatial applications, because UAV platforms often observe open, dynamic, and long-tailed environments. For example, a UAV monitoring system may encounter temporary

shelters, construction machinery, damaged buildings, rare vehicles, flooded roads, or newly appearing land-cover types that are absent from the original training set. Once novel categories appear, closed-set models usually require additional pixel-level annotations and expensive retraining, which limits their scalability and practical deployment.

Open-vocabulary semantic segmentation (OVSS) provides a promising solution by segmenting arbitrary categories according to textual category descriptions [8–12]. With vision–language models (VLMs), such as contrastive language–image pre-training (CLIP) [13], visual features and textual descriptions can be mapped into a shared embedding space, allowing unseen categories to be recognized through text prompts. This capability is particularly attractive for geospatial artificial intelligence: by changing the textual category set, an OVSS model can potentially recognize new objects, land-cover types, and scene elements without collecting additional dense annotations.

However, UAV-oriented OVSS remains insufficiently studied in remote sensing scenarios. Existing OVSS benchmarks mainly focus on natural images, while UAV imagery presents substantially different imaging characteristics. UAV images are captured from bird’s-eye or oblique viewpoints, where objects are often small, densely distributed, arbitrarily oriented, partially occluded, and visually similar to surrounding backgrounds [1,2,14,15]. Moreover, UAV datasets differ in altitude, camera angle, spatial resolution, scene type, geographic distribution, and annotation taxonomy. As a result, it is difficult to evaluate whether an OVSS model truly generalizes across diverse geospatial domains.

To fill this gap, we first construct a **Unified UAV Open-Vocabulary Semantic Segmentation Benchmark**. As illustrated in Figure 1, the goal is to move from conventional closed-set UAV segmentation toward open-vocabulary geospatial scene understanding with stronger generalization. The benchmark reorganizes multiple representative UAV semantic segmentation datasets into a unified evaluation protocol and emphasizes cross-dataset transfer [1,2,16]. Specifically, the model is trained on one UAV source dataset and evaluated on multiple target UAV datasets with different label spaces and visual distributions. This setting better reflects practical remote sensing deployment, where the testing environment is rarely identical to the training domain.

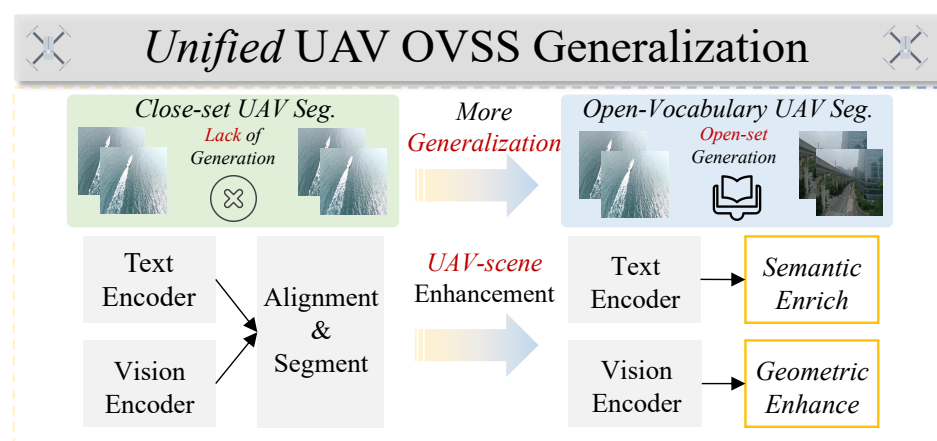


Figure 1. Motivation for unified UAV open-vocabulary semantic segmentation. Conventional closed-set UAV segmentation lacks generalization to unseen categories and domains. We construct a unified UAV OVSS benchmark and further improve UAV-based geospatial scene understanding from two complementary perspectives: LLM-guided semantic enrichment for the text branch and DINO-based geometric enhancement for the visual branch.

Figure 1 is intended to clarify the logic of the whole paper rather than only illustrate the task background. The left part shows why closed-set UAV segmentation is insufficient when target categories change across missions and domains. The middle part indicates that

open-vocabulary segmentation introduces text encoders for category-level generalization, but still lacks UAV-specific adaptation. The right part summarizes our solution: semantic enrichment strengthens the text branch, geometric enhancement strengthens the visual branch, and the two branches jointly support unified UAV OVSS generalization.

Beyond benchmark construction, directly applying existing OVSS models to UAV remote sensing imagery remains challenging from both the text and visual sides. On the text side, most OVSS methods use raw category names or simple prompts, such as “*a photo of a [class]*” [11,12,17]. Such prompts are often inadequate for geospatial scenes. A category in aerial imagery is not only defined by its name but also by its aerial appearance, geometric structure, spatial context, and relation with surrounding objects. For instance, *road* usually appears as an elongated continuous surface connected with vehicles and buildings; *building* often appears as a rectangular or polygonal roof region; and *vehicle* may occupy only a few pixels on roads or parking areas. Without these UAV-aware geospatial descriptions, textual prototypes can be ambiguous and weakly discriminative.

On the visual side, CLIP-like VLMs are mainly optimized with image-level contrastive learning [13,18]. Although they provide strong language alignment, they do not explicitly learn dense geometric structures. Therefore, their activation maps are often coarse and may produce blurred boundaries or incomplete small-object responses. This limitation is particularly severe for UAV imagery, where thin roads, small vehicles, roofs, vegetation boundaries, and shadows require fine-grained spatial reasoning. In contrast, DINO features contain stronger local structure and geometric cues [19], but lack explicit open-vocabulary language alignment. This motivates us to combine CLIP’s semantic openness with DINO’s geometric perception.

Based on these observations, we propose **UAV-OVSeg**, a UAV-oriented OVSS framework built upon a CAT-Seg-style cost aggregation paradigm [12]. Rather than introducing a complicated new architecture, we retain the dense visual–text matching mechanism of CAT-Seg and enhance it in two clear directions. First, an **LLM-guided Category Expansion Module** expands raw UAV category names into structured textual descriptions, including related categories, visual attributes, geometric properties, and geospatial context. The enriched prompts are encoded by the CLIP text encoder to form more informative textual prototypes. Second, a **DINO-enhanced Geometric Feature Fusion Module** extracts fine-grained geometric features from self-distillation with no labels (DINO) [19] and fuses them with CLIP visual features and category-aware cost maps, thereby improving dense localization while preserving open-vocabulary recognition.

Rather than treating LLM prompting and DINO features as isolated add-ons, UAV-OVSeg couples them with a UAV-specific cross-dataset benchmark: the text branch addresses category-name ambiguity under aerial viewpoints, while the visual branch compensates for the coarse spatial localization of CLIP in small-object UAV scenes. This design clarifies the methodological difference from directly applying CAT-Seg to UAV imagery.

The main contributions of this paper are summarized as follows:

- We construct a **Unified UAV Open-Vocabulary Semantic Segmentation Benchmark** by reorganizing multiple UAV semantic segmentation datasets into a unified cross-dataset protocol with explicit category harmonization, source–target transfer settings, and seen/unseen vocabulary analysis.
- We propose **UAV-OVSeg**, a UAV-oriented open-vocabulary segmentation framework that improves a CAT-Seg-style dense matching baseline through two integrated components: UAV-aware LLM-based textual enrichment and category-guided DINO geometric feature fusion. The framework is designed to preserve CLIP-based open-vocabulary alignment while strengthening UAV-specific semantic descriptions and local geometric localization.

- We provide reproducibility-oriented analyses and release plans, including LLM comparison, prompt sensitivity, object-scale evaluation, feature-fusion ablations, computational cost, prompt templates, category mappings, evaluation scripts, and model implementation.

2. Related Work

2.1. UAV Semantic Segmentation

UAV semantic segmentation parses aerial scenes at the pixel level and provides structured scene understanding for downstream applications [1,2]. Compared with natural images, UAV images have unique characteristics, including large-scale variation, arbitrary object orientation, dense object distribution, low foreground-background contrast, and complex scene layouts [20]. Early semantic segmentation methods mainly rely on convolutional encoder-decoder networks, multi-scale context aggregation, and skip connections [3–6]. More recent methods introduce attention mechanisms and Transformer-based architectures to capture long-range dependencies and improve global context modeling [7,21–27].

Although these methods achieve promising results, most of them are designed for closed-set segmentation. Their label space is fixed during training and cannot be flexibly expanded during inference. This limitation is unsuitable for open-world UAV applications, where novel categories and unseen scene elements frequently appear. In contrast, open-vocabulary learning enables models to segment categories specified by textual descriptions. However, UAV OVSS remains underexplored, especially under cross-dataset evaluation settings where the source and target UAV domains differ in viewpoint, resolution, scene type, and label taxonomy [14,15,28,29].

2.2. Open-Vocabulary Semantic Segmentation

Open-vocabulary semantic segmentation aims to segment arbitrary categories beyond a fixed training label set [8–10]. This task is largely enabled by vision–language models that align images and texts in a shared semantic embedding space [13,18]. Existing OVSS methods can be roughly divided into two categories. Two-stage methods first generate class-agnostic mask proposals and then classify each proposal using text embeddings [30,31]. These methods benefit from explicit region-level segmentation but depend heavily on proposal quality. One-stage methods directly compute dense visual–text similarities and predict segmentation masks in an end-to-end manner [11,12,17,32]. They are more efficient but often suffer from coarse localization because VLMs are not originally trained for dense prediction.

Most existing OVSS methods are developed and evaluated on natural image benchmarks such as ADE20K, Pascal VOC, and COCO-Stuff. Their applicability to UAV imagery is limited. In UAV scenes, objects can be extremely small, textures are often repetitive, and semantic boundaries may be difficult to distinguish from shadows or background clutter. Recent remote sensing OVSS studies have begun to explore open-vocabulary recognition in aerial and satellite imagery [33–38], and foundation-model-based remote sensing segmentation has also attracted increasing attention [39], but UAV-oriented benchmark construction and UAV-specific text–visual enhancement are still insufficiently studied. Therefore, this paper focuses on UAV OVSS by first constructing a unified UAV OVSS benchmark and then improving a strong dense visual–text matching baseline from the text and visual perspectives.

2.3. Differences from Existing OVSS Frameworks

The proposed work differs from existing OVSS frameworks in both problem setting and technical design. CAT-Seg [12] provides an effective dense visual–text cost aggregation

baseline, but it is mainly evaluated on natural-image benchmarks and does not explicitly model UAV-specific prompt ambiguity or small-object geometric degradation. SAN [17] and OVSeg [31] improve open-vocabulary segmentation through side-adapter learning or mask-level classification, while our method focuses on cross-dataset UAV transfer where the source and target label spaces and imaging distributions differ. Recent remote-sensing OVSS methods, such as SegEarth-style and AerialCLIP-style approaches [28,29,34,38], improve aerial vision–language alignment, but they usually do not jointly provide a unified multi-dataset UAV benchmark, category harmonization protocol, LLM prompt reproducibility, and category-guided DINO-CLIP fusion. Therefore, the novelty of UAV-OVSeg lies in formulating UAV OVSS as a reproducible cross-dataset benchmark and in designing a benchmark-driven text–visual enhancement strategy for UAV scenes.

2.4. Vision–Language Models and DINO Features

Large-scale vision–language models, such as CLIP [13], provide strong zero-shot recognition ability by learning image-text alignment from web-scale data. Their text encoder can map arbitrary class names or descriptions into semantic prototypes, while their image encoder can produce visual features comparable with text embeddings. This property makes CLIP a fundamental component in many OVSS methods [11,12,17].

However, CLIP is mainly optimized by image-level contrastive learning and is not specifically designed to capture dense spatial structures. When applied to pixel-level segmentation, CLIP features may produce incomplete object activations and imprecise boundaries. DINO features provide a complementary capability [19]. Since DINO is trained with self-supervised visual objectives, it captures rich local structures, object layouts, and geometric patterns. These properties are useful for UAV imagery, where roads, roofs, vehicles, vegetation, and shadows often require fine-grained geometric discrimination. Nevertheless, DINO does not provide explicit open-vocabulary language alignment. Therefore, combining CLIP and DINO is a natural strategy: CLIP contributes semantic openness, while DINO contributes geometric precision.

2.5. LLM-Based Prompt Enrichment for UAV OVSS

Text prompt design is important for open-vocabulary recognition. Simple category-name prompts may be insufficient for domain-specific scenes because they omit visual attributes, geometric shapes, and contextual relations [11,40,41]. Large language models provide rich world knowledge and can generate detailed descriptions for categories [42]. For UAV imagery, LLMs can describe not only what a category is, but also how it appears from an aerial viewpoint, what geometric structures it has, and what objects usually surround it.

For example, the class *vehicle* can be expanded to small rectangular objects on roads or parking lots, including cars, trucks, buses, and vans. The class *road* can be described as a long, narrow, continuous transportation surface connected with buildings, vehicles, and sidewalks. Such enriched descriptions provide more discriminative textual cues than raw class names. Motivated by this observation, we use LLM-guided category expansion to construct UAV-aware textual prototypes for open-vocabulary segmentation.

A possible risk of LLM-guided expansion is semantic drift: the generated text may contain overly broad concepts or context that is weakly related to the target class. To reduce this risk, we retain the original class name in every prompt, filter generated descriptions by CLIP text-space consistency, and cache the selected prompts for exact reproduction.

2.6. UAV Open-Vocabulary Benchmarks

A reliable benchmark is crucial for measuring open-vocabulary generalization. Natural-image OVSS benchmarks cannot fully reflect UAV-specific challenges, while existing UAV segmentation datasets are usually evaluated independently under closed-set pro-

protocols [1,2,16]. This makes it difficult to answer whether an open-vocabulary model trained on one UAV domain can generalize to another UAV domain. In remote sensing, recent benchmarks have begun to explore realistic open-vocabulary segmentation and language-guided dense perception [38,43]. In this work, we reorganize representative UAV datasets into a unified UAV OVSS benchmark. The proposed benchmark emphasizes cross-dataset transfer, open-vocabulary label specification, and evaluation across diverse UAV domains, providing a more practical testbed for UAV open-vocabulary scene understanding.

3. Methodology

3.1. Overview

Based on the proposed unified UAV Open-Vocabulary Semantic Segmentation Benchmark, we develop a UAV-oriented open-vocabulary segmentation framework, termed **UAV-OVSeg**. The goal of UAV-OVSeg is to improve the segmentation of arbitrary UAV scene categories specified by textual descriptions. Unlike conventional closed-set semantic segmentation models, which rely on a fixed classifier learned from predefined categories, UAV-OVSeg performs dense visual–text matching and can adapt to different category sets by changing the textual prompts.

We build our method upon CAT-Seg [12], a strong open-vocabulary semantic segmentation baseline based on cost aggregation. Given an input image and a set of category prompts, CAT-Seg first extracts visual features and textual prototypes, constructs category-aware cost maps by dense visual–text matching, and then aggregates these cost maps for final segmentation. This paradigm is suitable for open-vocabulary segmentation because the label space is not fixed by the segmentation head. Instead, the target categories are specified by text, allowing the model to recognize categories that are absent from the training label set.

However, directly applying CAT-Seg to UAV imagery remains suboptimal. UAV images usually contain small targets, dense object layouts, arbitrary orientations, complex backgrounds, and large-scale variations. Moreover, raw category names such as *road*, *building*, *vehicle*, or *vegetation* are often too short to describe their UAV-specific appearances. For example, *road* in UAV imagery usually appears as an elongated continuous surface connected with sidewalks, vehicles, and buildings, while *vehicle* may occupy only a very small region and appear as a tiny rectangular object. Such information is not explicitly encoded in simple class-name prompts.

To address these limitations, UAV-OVSeg improves open-vocabulary UAV segmentation from two complementary perspectives: **textual semantic enrichment** and **visual geometric enhancement**. As shown in Figure 2, the text branch uses a large language model (LLM) to expand each raw category name into UAV-aware descriptions, thereby constructing more discriminative textual prototypes. The visual branch incorporates DINO features [19] to enhance local geometry, object boundaries, and spatial structures. The enriched textual prototypes and geometry-enhanced visual features are then integrated through dense cost aggregation and decoded into pixel-level segmentation masks.

The workflow in Figure 2 further explains how the proposed modules are connected to the CAT-Seg-style baseline. The text branch does not change the segmentation head; instead, it replaces short class-name prompts with structured UAV-aware category descriptions before CLIP text encoding. The visual branch does not replace CLIP semantics; instead, it aligns DINO features to the CLIP feature space and injects them through a category-guided gate. Therefore, the framework preserves open-vocabulary flexibility while adding UAV-specific semantic and geometric priors.

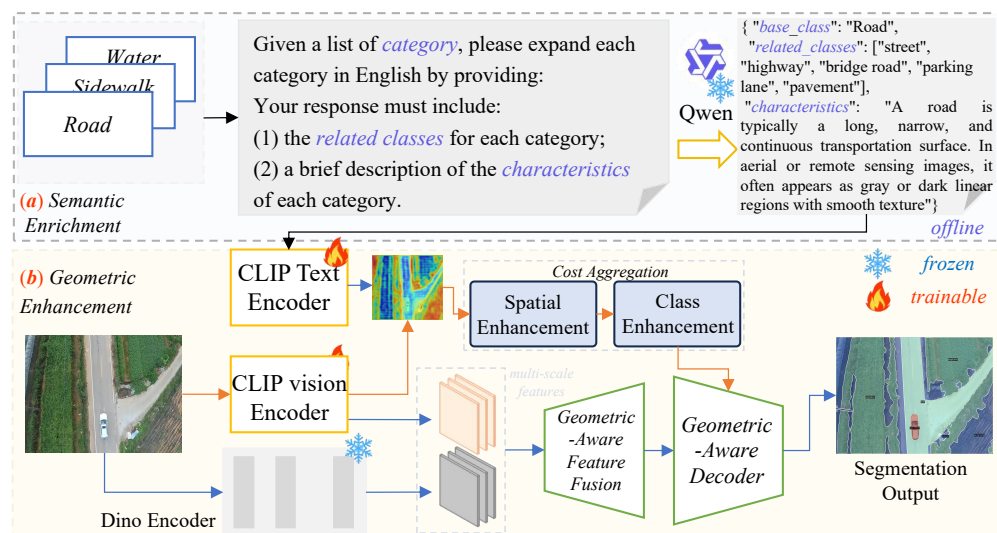


Figure 2. Overall framework of the proposed UAV-OVSeg. The method enhances UAV open-vocabulary semantic segmentation from both text and visual perspectives. The text branch performs LLM-guided semantic enrichment by expanding raw category names into related concepts and UAV-aware descriptions. The visual branch introduces DINO-enhanced geometric feature fusion to improve dense localization. The enhanced textual prototypes and geometry-aware visual features are integrated through CAT-Seg-style cost aggregation and decoded into segmentation masks. The stacked category cards in subfigure (a) denote multiple input labels for prompt expansion, and the snowflake and flame icons indicate frozen and trainable modules, respectively.

3.2. Problem Formulation

Given a UAV image $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$ and an open-vocabulary category set $\mathcal{C} = \{c_1, c_2, \dots, c_N\}$, the objective is to predict a pixel-level segmentation map:

$$\hat{\mathcal{Y}} \in \{1, 2, \dots, N\}^{H \times W}, \quad (1)$$

where each pixel is assigned to one of the categories specified by text. In contrast to closed-set segmentation, the category set $\mathcal{C}$ is not fixed during training and can vary across different testing datasets or application scenarios. Therefore, the model should rely on textual category descriptions rather than a fixed classification layer.

Following the open-vocabulary segmentation paradigm, we first construct a textual prompt p_i for each category c_i . The CLIP text encoder [13] maps the prompt into a textual prototype:

$$t_i = \Phi_t(p_i), \quad (2)$$

where Φ_t denotes the CLIP text encoder and t_i is the text embedding of category c_i . For the input UAV image, the CLIP visual encoder extracts dense visual features:

$$F_{\text{clip}} = \Phi_v(\mathcal{I}), \quad (3)$$

where $F_{\text{clip}} \in \mathbb{R}^{H' \times W' \times D}$ denotes the visual feature map and Φ_v denotes the CLIP visual encoder.

A dense visual-text similarity map is then computed for each category:

$$S_i(h, w) = \frac{F_{\text{clip}}(h, w)^\top t_i}{\|F_{\text{clip}}(h, w)\|_2 \|t_i\|_2}, \quad (4)$$

where $S_i(h, w)$ represents the semantic matching score between pixel location (h, w) and category c_i . The complete set of similarity maps is denoted as follows:

$$S = \{S_i\}_{i=1}^N. \quad (5)$$

Although this formulation enables flexible open-vocabulary prediction, it still has two limitations in UAV imagery. First, the quality of t_i heavily depends on the textual prompt p_i . If p_i only contains the raw class name, the resulting textual prototype may not capture UAV-specific semantics. Second, F_{clip} is mainly optimized for image-level language alignment and may lack dense geometric details, leading to coarse boundaries and incomplete small-object responses. UAV-OVSeg addresses these two limitations by enriching p_i with LLM-generated descriptions and enhancing F_{clip} with DINO-based geometric features.

3.3. Formal Definition of the Unified UAV OVSS Benchmark

To make the benchmark setting reproducible, we define a source UAV dataset as $\mathcal{D}_s = \{(\mathcal{I}_s^j, \mathcal{Y}_s^j)\}_{j=1}^{M_s}$ with source label space $\mathcal{C}_s$, and a target UAV dataset as $\mathcal{D}_t = \{(\mathcal{I}_t^j, \mathcal{Y}_t^j)\}_{j=1}^{M_t}$ with target label space $\mathcal{C}_t$. A model is trained on $\mathcal{D}_s$ and evaluated on $\mathcal{D}_t$ by replacing the textual vocabulary with the target category set $\mathcal{C}_t$, without changing the segmentation head. The evaluation is written as

$$\text{Eval}(f_\theta; \mathcal{D}_s \rightarrow \mathcal{D}_t, \mathcal{C}_t) = \frac{1}{|\mathcal{C}_t^{\text{eval}}|} \sum_{c \in \mathcal{C}_t^{\text{eval}}} \text{IoU}_c(f_\theta(\mathcal{I}_t, \mathcal{P}(\mathcal{C}_t)), \mathcal{Y}_t), \quad (6)$$

where $\mathcal{P}(\mathcal{C}_t)$ denotes the text prompts generated from the target vocabulary and $\mathcal{C}_t^{\text{eval}}$ denotes the valid evaluation categories after unlabeled or ambiguous regions have been ignored.

For category harmonization, we normalize category names by lowercasing, unifying singular and plural forms, merging synonyms, and performing manual verification. Semantically equivalent labels are mapped to a canonical class, while ambiguous parent–child labels are either kept separate or ignored when a reliable mapping cannot be established. We further distinguish two target subsets: $\mathcal{C}_t^{\text{all}}$, containing all valid target categories, and $\mathcal{C}_t^{\text{unseen}} = \{c \in \mathcal{C}_t : c \notin \text{Syn}(\mathcal{C}_s)\}$, where $\text{Syn}(\mathcal{C}_s)$ denotes the source label names and their synonyms. This stricter definition prevents overclaiming when common UAV classes such as *road* or *building* appear in both source and target label spaces.

3.4. LLM-Guided Category Expansion

3.4.1. Motivation for Textual Semantic Enrichment

Text prompts play a crucial role in open-vocabulary segmentation. Most existing methods use raw category names or simple templates, such as “a photo of a [class]”. While such prompts are effective for natural images, they are often insufficient for UAV scenes. The reason is that UAV categories are strongly associated with aerial-view appearance, geometric structure, object scale, and surrounding context. For instance, the category *road* may correspond to street, highway, bridge road, parking lane, or pavement, and it often appears as a long, narrow, continuous region in aerial images. Similarly, *building* is usually characterized by roof texture, rectangular or polygonal shape, and regular spatial layout. These cues are important for segmentation but are not explicitly described by raw category names.

To construct more informative textual prototypes, we introduce an **LLM-guided Category Expansion Module**. The module uses an LLM to automatically expand each category into structured descriptions containing related concepts, visual attributes, geometric properties, and UAV-scene contextual cues. This strategy reduces manual prompt engineering and makes the text branch more suitable for different UAV datasets and testing category sets.

3.4.2. Structured Category Description

For each category c_i , we query the LLM with a UAV-oriented instruction prompt $\mathcal{P}_{\text{uav}}$. The prompt asks the LLM to describe the category in terms of four aspects:

$$d_i = \text{LLM}(c_i, \mathcal{P}_{\text{uav}}), \quad (7)$$

where d_i is the generated structured description. Specifically, we formulate d_i as follows:

$$d_i = \{r_i, a_i, g_i, q_i\}, \quad (8)$$

where r_i denotes related concepts, a_i denotes visual attributes, g_i denotes geometric properties, and q_i denotes UAV-scene contextual cues.

The four components have different functions. The related concepts r_i expand the semantic coverage of the category. For example, the category *road* may be related to street, highway, parking lane, and pavement. The visual attributes a_i describe color, texture, material, and appearance. The geometric properties g_i describe shape, scale, continuity, orientation, and boundary characteristics. The contextual cues q_i describe spatial relations with other objects, such as roads being adjacent to vehicles and sidewalks, or buildings being surrounded by roads and vegetation.

The final prompt is constructed by combining the original class name and the generated description:

$$p_i = \text{Template}(c_i, d_i). \quad (9)$$

This design preserves the identity of the original category while introducing richer UAV-aware semantic information. The textual prototype is then obtained as follows:

$$t_i = \Phi_t(p_i). \quad (10)$$

3.4.3. Prompt Ensemble and Semantic Filtering

LLM-generated descriptions may occasionally include overly broad, redundant, or weakly related concepts. To reduce semantic drift, we generate multiple candidate descriptions for each category and perform semantic filtering before constructing the final textual prototype. Given K' candidate prompts $\{p_i^1, p_i^2, \dots, p_i^{K'}\}$, we measure the semantic consistency between each expanded prompt and the original class-name prompt:

$$\alpha_i^k = \frac{\Phi_t(p_i^k)^\top \Phi_t(c_i)}{\|\Phi_t(p_i^k)\|_2 \|\Phi_t(c_i)\|_2}. \quad (11)$$

The top- K prompts with higher consistency scores are retained. The final textual prototype is obtained by averaging their embeddings:

$$t_i = \frac{1}{K} \sum_{k=1}^K \Phi_t(p_i^k). \quad (12)$$

This prompt ensemble strategy has two advantages. First, it improves the robustness of textual prototypes by reducing the sensitivity to a single prompt. Second, it allows the model to cover multiple visual appearances of the same category, which is important for UAV imagery where the same semantic category may have different scales, textures, and viewpoints.

In the main implementation, we use a Qwen-family instruction-tuned LLM for category expansion, with a temperature of 0.2, top- $p = 0.9$, a maximum output length of 512 tokens, and a fixed random seed for reproducibility. For each category, we generate $K' = 5$ candidate descriptions and retain $K = 3$ prompts according to Equation (10). The complete

prompt template and generated examples are provided in Appendix A. We also compare GPT-4-family, Qwen-family, and Llama-family models in Section 4. Since the LLM is used offline and its outputs are cached, different LLM parameter sizes affect the quality and stability of textual descriptions but do not increase the online segmentation parameters or inference latency.

3.5. DINO-Enhanced Geometric Feature Fusion

3.5.1. Motivation

Although CLIP features provide strong open-vocabulary semantic alignment, they are mainly learned through image-text contrastive learning and are not specifically optimized for dense prediction. As a result, CLIP-based cost maps may highlight category-relevant regions but often lack precise boundaries and local structures. This problem is particularly severe in UAV imagery because objects can be small, thin, rotated, and densely distributed. For example, vehicles may occupy only a few pixels, roads may appear as long and narrow regions, and vegetation boundaries may be irregular and easily confused with shadows.

DINO features provide complementary visual information. Since DINO is trained with self-supervised visual objectives, it tends to capture local structures, object layouts, and geometric patterns. These properties are useful for UAV segmentation, especially for preserving object shapes and improving boundary quality. However, DINO does not provide explicit language alignment. Therefore, we use DINO as a geometric feature source and fuse it with CLIP features rather than replacing CLIP.

3.5.2. Feature Extraction and Projection

Given the input UAV image $\mathcal{I}$, we extract CLIP visual features and DINO visual features:

$$F_{\text{clip}} = \Phi_{\text{clip}}(\mathcal{I}), \quad F_{\text{DINO}} = \Phi_{\text{DINO}}(\mathcal{I}), \quad (13)$$

where Φ_{clip} denotes the CLIP visual encoder and Φ_{DINO} denotes the DINO encoder. Since the two feature streams may have different spatial resolutions, channel dimensions, and feature distributions, we first align the DINO features with the CLIP feature space:

$$\tilde{F}_{\text{DINO}} = \text{Proj}(F_{\text{DINO}}), \quad (14)$$

where $\text{Proj}(\cdot)$ is a lightweight projection module composed of interpolation and linear transformation. The projected feature $\tilde{F}_{\text{DINO}}$ has the same spatial resolution and channel dimension as F_{clip} .

Concretely, with a 512×512 input image, the CLIP ViT-B/16 feature map is aligned to a 32×32 spatial grid with 512 channels. The DINOv2-Base feature is extracted from the last transformer layer, interpolated to the same 32×32 resolution, and projected to 512 channels by a lightweight 1×1 linear projection followed by normalization. Although interpolation may smooth very high-frequency details and channel projection compresses part of the DINO representation, the alignment is performed on dense feature maps rather than on a single global token. Thus, local structural cues are largely preserved and can still complement CLIP's semantic features.

3.5.3. Geometry-Aware Feature Fusion

A naive way to combine CLIP and DINO is to directly add or concatenate their features. However, such fixed fusion ignores the fact that different image regions require different types of information. For example, small vehicles require fine local geometry, while large land-cover regions may rely more on semantic cues. Therefore, we design a geometry-aware fusion mechanism guided by category-aware cost maps.

We first concatenate the CLIP feature, the projected DINO feature, and the initial cost maps:

$$Z = [F_{\text{clip}}, \tilde{F}_{\text{DINO}}, S], \quad (15)$$

where $[\cdot]$ denotes channel-wise concatenation. A lightweight convolutional module predicts a fusion gate:

$$G = \sigma(\text{Conv}(Z)), \quad (16)$$

where $\sigma(\cdot)$ is the sigmoid function and G controls the contribution of semantic and geometric features at each spatial location. The geometry-enhanced visual feature is computed as:

$$F_{\text{geo}} = G \odot F_{\text{clip}} + (1 - G) \odot \tilde{F}_{\text{DINO}}, \quad (17)$$

where $\odot$ denotes element-wise multiplication.

This adaptive fusion allows the model to preserve CLIP's open-vocabulary semantic alignment while selectively introducing DINO's geometric priors. When semantic confidence is high, the model can rely more on CLIP features. When local structures and boundaries are important, the model can incorporate more DINO information.

The initial cost map S is computed once from the original CLIP feature and text prototypes before DINO fusion and is used only as category-aware guidance for predicting G . It is not recursively updated during gate estimation, so the design does not form a circular dependency. Compared with cross-attention or transformer fusion, additive gating keeps the fused representation close to the CLIP-compatible feature space and introduces DINO geometry in a controlled manner. This is important for preserving open-vocabulary alignment while avoiding excessive computation and overfitting under limited UAV segmentation data.

3.5.4. Geometry-Enhanced Cost Maps

After obtaining F_{geo} , we recompute the dense visual–text similarity maps:

$$S_i^{\text{geo}}(h, w) = \frac{F_{\text{geo}}(h, w)^{\top} t_i}{\|F_{\text{geo}}(h, w)\|_2 \|t_i\|_2}. \quad (18)$$

Compared with the original CLIP-only similarity maps, the geometry-enhanced cost maps contain stronger local structures and clearer object boundaries. This is beneficial for UAV images because dense prediction requires not only category-level recognition but also accurate spatial localization.

Figure 3 provides a visual explanation of why DINO features are useful in this framework. The DINO response maps contain sharper local structures and more continuous geometric patterns than CLIP-only dense matching, while the fused representation suppresses noisy activations and better preserves object boundaries. This observation supports the design choice of using DINO as a complementary geometric source rather than as an independent open-vocabulary classifier.

3.6. Geometry-Aware Decoder

After obtaining the geometry-enhanced cost maps $S^{\text{geo}} = \{S_i^{\text{geo}}\}_{i=1}^N$, we use a lightweight decoder to produce the final segmentation map. Following the CAT-Seg-style decoding paradigm, the decoder aggregates category-aware cost maps and visual features to recover high-resolution predictions:

$$\hat{\mathcal{Y}} = \Phi_{\text{Dec}}(S^{\text{geo}}, F_{\text{geo}}), \quad (19)$$

where Φ_{Dec} denotes the segmentation decoder.

The decoder has two objectives. First, it aggregates multi-scale semantic responses from the cost maps, allowing the model to distinguish categories specified by text. Second, it uses geometry-enhanced visual features to refine object boundaries and small regions. This design is particularly important for UAV imagery, where many categories exhibit strong geometric patterns. For example, roads and sidewalks are usually elongated and continuous, vehicles are small and compact, buildings often have regular roof shapes, and water regions may have smooth textures and irregular boundaries.

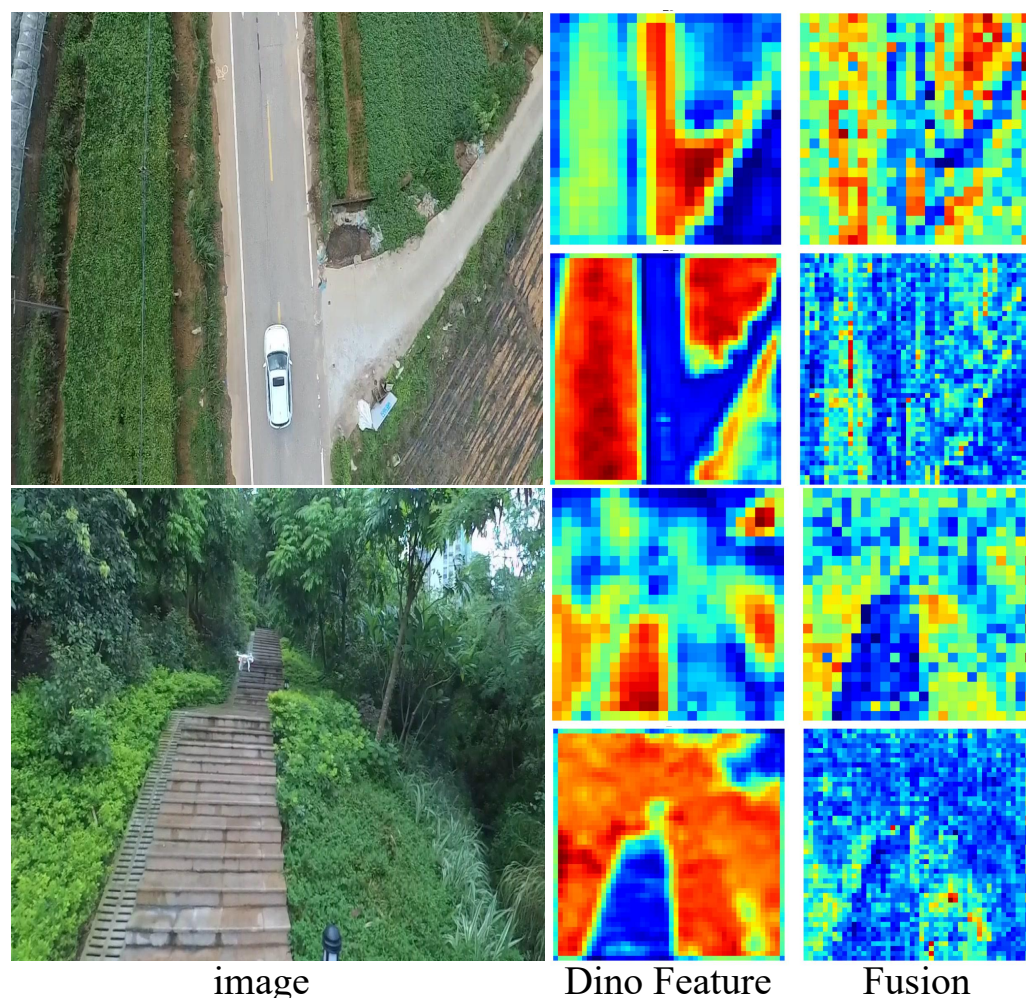


Figure 3. Visualization of DINO-based geometric enhancement. DINO features capture local structures and geometric patterns that are complementary to CLIP semantic features. After fusion, the visual representation becomes more suitable for dense UAV segmentation. In the response maps, warmer colors indicate higher activation values and cooler colors indicate lower activation values.

In practice, the decoder progressively upsamples the low-resolution cost maps and fuses them with visual features. The final prediction is obtained by applying a pixel-wise classifier over the decoded feature map:

$$P = \text{Softmax}(\Phi_{\text{cls}}(\Phi_{\text{Dec}}(S^{\text{geo}}, F_{\text{geo}}))), \quad (20)$$

where $P \in \mathbb{R}^{H \times W \times N}$ is the category probability map and Φ_{cls} denotes the prediction head. The final segmentation result is:

$$\hat{\mathcal{Y}}(h, w) = \arg \max_i P_i(h, w). \quad (21)$$

3.7. Training Objective

The model is trained with a standard pixel-wise cross-entropy loss:

$$\mathcal{L}_{\text{seg}} = \mathcal{L}_{\text{CE}}(\hat{\mathcal{Y}}, \mathcal{Y}), \quad (22)$$

where $\mathcal{Y}$ denotes the ground-truth segmentation map. This loss supervises the final segmentation prediction and encourages the model to assign each pixel to the correct semantic category.

To stabilize feature fusion, we further introduce a cost consistency loss between the original CLIP-based cost maps and the DINO-enhanced cost maps:

$$\mathcal{L}_{\text{cons}} = \frac{1}{NHW} \sum_{i=1}^N \sum_{h=1}^H \sum_{w=1}^W \left\| S_i^{\text{geo}}(h, w) - S_i(h, w) \right\|_2^2. \quad (23)$$

This consistency term prevents the DINO-enhanced branch from deviating too much from the original open-vocabulary semantic alignment. In other words, DINO features are encouraged to improve geometry and localization while preserving the semantic matching ability provided by CLIP.

The total training objective is formulated as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{seg}} + \lambda_{\text{cons}} \mathcal{L}_{\text{cons}}, \quad (24)$$

where λ_{cons} controls the weight of the consistency loss. A small value of λ_{cons} allows the model to benefit from DINO-based geometric enhancement without suppressing the flexibility of the visual-text matching process.

3.8. Inference

During inference, UAV-OVSeg only requires an input image and a list of target category names. For each target category, the LLM-expanded textual prototype can be precomputed and stored. Therefore, the LLM is not involved in the online inference stage. Given a new category set, the model first loads or generates the corresponding textual prototypes, then computes geometry-enhanced cost maps, and finally predicts the segmentation mask through the decoder.

This inference process keeps the flexibility of open-vocabulary segmentation. When the target dataset or application scenario changes, users only need to provide a new category list. The segmentation head does not need to be redesigned or retrained for a fixed label space. This property is especially suitable for UAV applications, where mission-specific categories may vary across urban monitoring, traffic inspection, disaster assessment, environmental observation, and infrastructure analysis.

4. Experiments

All experiments were implemented in Python 3.10 with PyTorch 2.1.0 and CUDA 12.1. Training and inference were performed on NVIDIA H100 GPUs (NVIDIA Corporation, Santa Clara, CA, USA).

4.1. Unified UAV OVSS Benchmark

To evaluate the generalization ability of open-vocabulary semantic segmentation in UAV imagery, we construct a **Unified UAV Open-Vocabulary Semantic Segmentation Benchmark**. Unlike conventional UAV semantic segmentation protocols that train and test models within a fixed dataset and a fixed label space, our benchmark aims to evaluate whether a model can segment UAV scenes according to flexible textual category specifica-

tions. This setting is closer to practical UAV applications, where the target categories may vary with different missions, and the testing scenes may contain categories that are not fully covered by the training annotations.

The proposed benchmark integrates several representative UAV semantic segmentation datasets, including SynDrone [44], VDD [2], UDD6 [16], UAVid [1], Aerialscapes [45], and ICG Drone [16]. These datasets cover diverse UAV scenarios, including synthetic aerial scenes, real urban environments, high-altitude orthophoto imagery, low-altitude oblique-view images, traffic scenes, vegetation regions, buildings, roads, vehicles, sidewalks, water areas, and other man-made structures. Compared with natural-image segmentation datasets, UAV datasets present more severe scale variations, dense object layouts, weak foreground-background contrast, flexible viewpoints, and large differences in annotation taxonomies. Therefore, simply evaluating a model on a single UAV dataset is insufficient for measuring its open-vocabulary generalization ability.

As shown in Figure 4, the benchmark contains two training settings and multiple testing datasets. In **Setting 1**, SynDrone is used as the training dataset, and the model is evaluated on several UAV datasets with different visual distributions and category spaces. This setting evaluates whether a model trained on synthetic UAV imagery can perform open-vocabulary segmentation on more diverse UAV scenes. In **Setting 2**, Aerialscapes is used as the training dataset, and the model is evaluated on the remaining UAV datasets. This setting evaluates the generalization ability of a model trained on real UAV imagery. During testing, the target label space is specified by textual category names or category descriptions, following the open-vocabulary segmentation protocol.

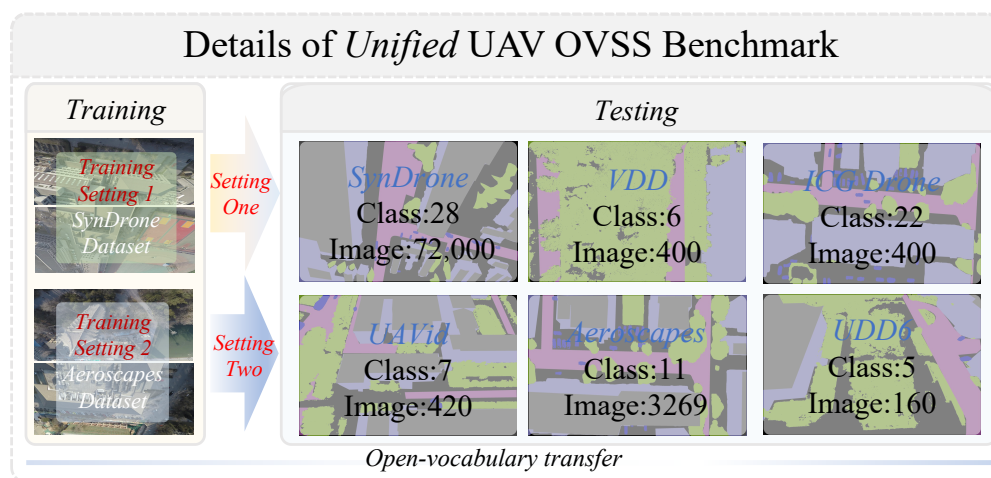


Figure 4. Details of the proposed unified UAV Open-Vocabulary Semantic Segmentation Benchmark. The benchmark contains two training settings and multiple testing datasets. SynDrone and Aerialscapes are used as two representative training sources, while VDD, ICG Drone, UDD6, UAVid, Aerialscapes, and SynDrone are used for evaluation under different settings. This benchmark evaluates open-vocabulary UAV scene understanding under multiple datasets, category spaces, and visual distributions. The colors are used only to distinguish dataset groups and source–target roles in the benchmark protocol; no quantitative value is encoded by color. Numeric values with five or more digits are written using comma separators in the revised figure.

Figure 4 makes the benchmark protocol explicit. SynDrone and Aerialscapes are selected as two representative source domains because they correspond to synthetic and real UAV imagery, respectively. The remaining datasets are used as target domains with different label spaces, viewpoints, resolutions, and annotation styles. This design avoids evaluating OVSS only within a single closed dataset and makes it possible to measure cross-dataset vocabulary transfer and domain generalization.

SynDrone. SynDrone is a synthetic UAV semantic segmentation dataset generated from virtual aerial environments. It contains abundant annotated images, regular scene layouts, and relatively clean object boundaries. These properties make it useful for training segmentation models. However, synthetic UAV images differ from real UAV images in texture, illumination, object appearance, and background complexity. Therefore, using SynDrone as the training source can test whether open-vocabulary models learn transferable UAV representations rather than overfitting to a specific visual style.

VDD. VDD is a real-world drone semantic segmentation dataset containing varied urban and suburban scenes [2]. It includes common aerial categories such as roads, buildings, vegetation, vehicles, and background regions. Realistic shadows, occlusions, scale changes, and irregular object shapes make VDD suitable for evaluating UAV segmentation robustness.

UDD6. UDD6 contains overhead UAV images with representative urban categories. It is useful for evaluating segmentation under near-orthographic viewpoints, dense object distributions, and large-scale variations. Compared with low-altitude oblique-view UAV datasets, UDD6 emphasizes the difficulty of recognizing small and densely arranged objects from top-view aerial imagery.

UAVid. UAVid is an urban UAV video semantic segmentation dataset captured from low-altitude platforms [1]. It contains traffic scenes with small vehicles, roads, buildings, trees, and pedestrians. Since the images are collected from UAV videos, the dataset includes complex viewpoint changes, motion blur, and dynamic urban layouts.

Aeroscapes. Aeroscapes contains low-altitude aerial images with diverse scene types and rich semantic categories. Its oblique viewpoints, perspective distortion, and complex layouts make it suitable for evaluating UAV open-vocabulary segmentation under realistic aerial-scene variations. In our benchmark, Aeroscapes is also used as one of the training datasets to evaluate the effect of using real UAV imagery as the source data.

ICG Drone. ICG Drone contains real UAV scenes with diverse land-cover and man-made object categories [16]. It provides additional visual and semantic diversity for evaluating open-vocabulary UAV segmentation, especially when the testing categories and scene structures differ from common urban UAV datasets.

Table 1 shows that, when trained on SynDrone, UAV-OVSeg obtains the best results on all five target datasets. Compared with CAT-Seg, the overall mIoU increases from 62.9% to 65.5%, and the overall mACC increases from 74.8% to 78.0%. The largest dataset-level gains appear on VDD and ICG Drone, which contain realistic UAV textures, shadows, and complex background layouts. This indicates that the proposed text–visual enhancement is particularly helpful when transferring from synthetic training data to real UAV scenes.

Table 1. Open-vocabulary transfer evaluation under **Setting 1**, where all methods are trained on SynDrone and evaluated on multiple target UAV datasets. Best results are highlighted in bold, and second-best results are underlined.

Method	Backbone	VDD		UDD6		UAVid		Aeroscapes		ICG Drone		Overall	
		mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC
ZegFormer	ViT-B	64.3	76.2	66.0	77.1	61.5	73.6	58.9	70.4	48.6	61.8	59.9	71.8
OpenSeeD	ViT-B	65.9	77.4	67.2	78.3	62.7	74.9	60.2	71.8	49.3	62.6	61.1	73.0
SED [32]	ViT-B	66.7	78.1	66.5	77.8	62.3	74.5	61.3	72.6	51.2	64.1	61.6	73.4
OVSeg [31]	ViT-B	66.5	77.9	67.4	78.7	62.5	74.8	61.0	72.2	50.5	63.5	61.2	73.3
SAN [17]	ViT-B	67.3	78.8	68.3	79.6	63.9	76.2	61.5	73.1	51.6	64.8	62.5	74.5
CAT-Seg [12]	ViT-B	67.8	79.2	68.2	79.5	63.7	76.0	62.3	74.0	52.3	65.4	62.9	74.8
UAV-OVSeg	ViT-B	70.9	82.9	<u>70.6</u>	81.8	65.4	78.7	<u>64.9</u>	76.8	55.7	69.6	<u>65.5</u>	78.0

4.2. Category-Overlap Analysis

Table 2 summarizes the source–target category overlap in the two transfer settings. In Setting 1, SynDrone is used as the source dataset, and 32 out of 51 target categories overlap with the source label space, leading to an overlap ratio of 62.7%. In Setting 2, Aeroscapes is used as the source dataset, and only 27 out of 67 target categories overlap with the source label space, resulting in a lower overlap ratio of 40.3%. This indicates that Setting 2 contains more non-overlapping categories and is therefore more challenging for open-vocabulary transfer.

Table 2. Category-overlap statistics for the two open-vocabulary transfer settings. Overlapping and non-overlapping target categories are listed in separate columns using a unified black-and-white format.

Setting	Source	Target	No. Overlap	#Non-Overlap	#Target Classes	Overlapping Target Categories	Non-Overlapping Target Categories
Setting 1	SynDrone	VDD	5	1	6	wall; road; vegetation; vehicle; water	roof
		UDD6	3	2	5	road; vegetation; vehicle	facade; roof
		UAVid	7	0	7	building; road; tree; low vegetation; moving car; static car; human	–
		Aeroscapes	6	5	11	person; bike; car; vegetation; road; sky	drone; boat; animal; obstacle; construction
		ICG Drone	11	11	22	grass; water; vegetation; wall; fence; fence-pole; person; car; bicycle; tree; bald-tree	paved-area; dirt; gravel; rocks; pool; roof; window; door; dog; ar-marker; obstacle
Setting 2	Aeroscapes	SynDrone	6	21	27	road; vegetation; person; car; bicycle; sky	sidewalk; ground; rail track; road line; terrain; rider; train; truck; bus; motorcycle; building; wall; fence; bridge; pole; static; guard rail; dynamic; traffic signs; traffic light; water
		VDD	3	3	6	road; vegetation; vehicle	wall; roof; water
		UDD6	3	2	5	road; vegetation; vehicle	facade; roof
		UAVid	6	1	7	road; tree; low vegetation; moving car; static car; human	building
		ICG Drone	9	13	22	grass; vegetation; person; dog; car; bicycle; tree; bald-tree; obstacle	paved-area; dirt; gravel; water; rocks; pool; roof; wall; window; door; fence; fence-pole; ar-marker
Setting 1 Total			32	19	51	Overlap ratio: 62.7%	
Setting 2 Total			27	40	67	Overlap ratio: 40.3%	

Note. Category overlap is counted after the lightweight synonym-level normalization used in benchmark construction, e.g., *bike/bicycle*, *human/person*, *moving car* or *static car/car*, *low vegetation* or *tree/grass/bald-tree/vegetation*, and *vehicle*/the car-like vehicle group. Categories without reliable synonym-level correspondence are treated as non-overlapping. Italic text denotes segmentation label names, and bold values in the last two rows denote setting-level totals.

The overlap analysis further clarifies the evaluation scope of our benchmark. For overlapping categories, the main difficulty lies in cross-dataset domain shift, such as viewpoint, resolution, texture, and annotation differences. For non-overlapping categories, the model needs to rely more on textual descriptions and vision–language alignment. Therefore, the benchmark evaluates both domain generalization on shared UAV classes and open-vocabulary recognition on dataset-specific categories.

4.3. Evaluation Protocol and Implementation Details

We adopt mean intersection over union (mIoU) and mean accuracy (mACC) as the main evaluation metrics. mIoU measures the region-level overlap between predicted masks and ground-truth masks and is widely used in semantic segmentation. mACC measures the average per-class pixel accuracy and reflects whether the model can recognize different categories in a balanced manner. For each training setting, we report results on all testing datasets and calculate the overall average to summarize multi-dataset open-vocabulary generalization.

Table 3 further verifies that the improvement is not limited to synthetic-source training. When Aerscapes is used as the real UAV source dataset, UAV-OVSeg still improves the overall mIoU from 61.9% to 64.5% and the overall mACC from 73.7% to 76.8% over CAT-Seg. The consistent gains across SynDrone, VDD, UDD6, UAVid, and ICG Drone suggest that UAV-aware textual expansion and DINO-based geometric fusion remain effective under real-to-real and real-to-synthetic transfer settings.

Table 3. Open-vocabulary transfer evaluation under **Setting 2**, where all methods are trained on Aerscapes and evaluated on multiple target UAV datasets. Best results are highlighted in bold, and second-best results are underlined.

Method	Backbone	SynDrone		VDD		UDD6		UAVid		ICG Drone		Overall	
		mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC
ZegFormer	ViT-B	55.2	66.5	66.1	77.6	62.9	74.2	60.8	72.5	50.3	63.0	59.1	70.8
OpenSeeD	ViT-B	56.4	67.8	67.0	78.5	63.7	75.0	61.5	73.4	51.4	64.1	60.0	71.8
SED [32]	ViT-B	57.2	68.4	67.4	78.9	64.1	75.3	61.9	73.8	52.0	64.8	60.5	72.2
OVSeg [31]	ViT-B	57.0	68.1	67.2	78.7	64.0	75.2	61.6	73.5	51.8	64.5	60.3	72.0
SAN [17]	ViT-B	58.1	69.3	68.0	79.5	64.8	76.1	62.5	74.6	52.7	65.5	61.2	73.0
CAT-Seg [12]	ViT-B	58.8	70.0	68.7	80.2	65.5	76.8	63.0	75.1	53.5	66.3	61.9	73.7
UAV-OVSeg	ViT-B	61.7	73.6	71.3	83.1	68.0	79.5	65.2	77.9	56.4	70.0	64.5	76.8

We use CAT-Seg [12] as the main baseline because it is a strong open-vocabulary semantic segmentation framework based on cost aggregation. CAT-Seg computes dense visual–text matching maps between image features and textual category prototypes and then aggregates these cost maps for final segmentation. This design is suitable for open-vocabulary segmentation because the testing label set can be changed by replacing textual prompts.

For a fair comparison, all methods compared are retrained and re-evaluated under the same experimental protocol. Specifically, each method is trained with a batch size of 8 for 40,000 iterations on four NVIDIA H100 graphics processing units (GPUs). Unless otherwise specified, the input resolution is set to 512×512 . We use AdamW as the optimizer with an initial learning rate of 1×10^{-4} and a weight decay of 5×10^{-2} . For model backbones, CLIP ViT-B/16 is used as the vision–language backbone for open-vocabulary semantic alignment, and DINOv2-Base is used as the geometric feature encoder. The coefficient λ_{cons} for cost consistency regularization is empirically set to 0.1. For each training setting, all methods use the same source training dataset, the same target testing datasets, and the same evaluation category names. For baseline methods, we use the original category-name prompts following their standard open-vocabulary evaluation protocol. For UAV-OVSeg,

the LLM-guided category expansion uses only the textual category names and does not access target images or ground-truth masks. Ignored labels are excluded from both mIoU and mACC during evaluation.

All main results are repeated with three random seeds, and the additional robustness table reports mean and standard deviation. The LLM-generated prompts are produced offline and cached. Unless otherwise specified, the prompt-generation setting uses $K' = 5$ candidates and the top- $K = 3$ retained prompts. Computational cost is measured at the same input resolution and on the same hardware to clarify the additional overhead introduced by DINO feature extraction and lightweight fusion.

4.4. Comparison with Existing OVSS Methods

We compare UAV-OVSeg with representative open-vocabulary semantic segmentation methods, including two-stage mask-classification methods and one-stage dense matching methods. Two-stage methods, such as OVSeg [31], first generate class-agnostic masks and then classify each region using textual embeddings. One-stage methods, such as SAN [17], SED [32], and CAT-Seg [12], directly compute dense visual–text similarities for open-vocabulary prediction. In contrast to single-dataset evaluation, our benchmark evaluates each method across multiple UAV datasets and flexible testing category sets. This protocol provides a more comprehensive view of whether existing OVSS methods can be directly applied to UAV imagery.

Table 4 clarifies the methodological positioning of UAV-OVSeg. Existing natural-image OVSS methods provide dense open-vocabulary prediction, but they are not designed around UAV cross-dataset evaluation. Recent remote-sensing or aerial VLM methods improve aerial semantic alignment, but they usually do not jointly provide a unified UAV benchmark, LLM-based structured category expansion, and category-guided DINO fusion. Therefore, the novelty of UAV-OVSeg lies in the UAV-specific benchmark-method coupling rather than simply combining existing modules.

Table 4. Methodological comparison with representative OVSS and remote-sensing OVSS methods. A checkmark indicates that the corresponding property is explicitly supported by the method.

Method	Main Setting	Dense OVSS	UAV Benchmark	LLM Prompt	DINO Fusion	Cross-Dataset Eval.
OVSeg [31]	Natural image	✓				
SAN [17]	Natural image	✓				
SED [32]	Natural image	✓				
CAT-Seg [12]	Natural image	✓				
SegEarth-style [34]	Remote sensing	✓				Partial
AerialCLIP-style [28]	UAV/aerial	✓	Partial			Partial
UAV-OVSeg	UAV OVSS	✓	✓	✓	✓	✓

4.5. Open-Vocabulary Transfer Evaluation with SynDrone Training

In the first setting, all models are trained on SynDrone and evaluated on multiple UAV testing datasets. This protocol tests whether the model can learn useful UAV segmentation knowledge from synthetic aerial images and transfer it to diverse testing scenes through textual category descriptions. The challenge mainly lies in the differences in image appearance, category definition, scene layout, object scale, and annotation style among different UAV datasets.

As shown in Table 1, CAT-Seg achieves strong baseline performance, indicating that dense visual–text cost aggregation is effective for UAV open-vocabulary segmentation. However, its performance is still limited by raw textual prompts and CLIP-only visual features. UAV-OVSeg consistently improves over CAT-Seg on all testing datasets. The improvement indicates that LLM-expanded textual prototypes help reduce category ambi-

guity, while DINO-enhanced geometric features improve localization under scale variation, cluttered backgrounds, and complex UAV scene layouts. These results demonstrate that the proposed text–visual enhancement strategy is effective when the training source is synthetic UAV imagery.

4.6. Open-Vocabulary Transfer Evaluation with Aeroscapes Training

In the second setting, all models are trained on Aeroscapes and evaluated on other UAV datasets. Compared with SynDrone, Aeroscapes contains real aerial images with richer textures and more complex scene layouts. This setting evaluates whether the proposed method remains effective when the training data already comes from real UAV imagery. The main challenges include inconsistent category definitions, different viewpoints, different image resolutions, and diverse scene compositions among datasets.

Table 3 shows that UAV-OVSeg also achieves the best overall performance when trained on Aeroscapes. This suggests that the proposed improvements are not restricted to synthetic training data. LLM-based semantic enrichment improves category-level transfer across different UAV label spaces, while DINO-based geometric enhancement improves the robustness of dense prediction under viewpoint, resolution, and scene-layout variations. The consistent gains over CAT-Seg further show that simply using a strong OVSS baseline is not sufficient for UAV imagery; UAV-specific text and visual enhancements are necessary.

4.7. Component Ablation

Unless otherwise specified, all ablation studies are conducted under **Setting 1**, where SynDrone is used as the source training dataset, and the results are averaged over all target datasets in Table 1.

We conduct ablation studies to evaluate the contributions of the two proposed components. The baseline is CAT-Seg [12] with raw category-name prompts and CLIP visual features. We then progressively add the LLM-guided Category Expansion Module and the DINO-enhanced Geometric Feature Fusion Module.

As shown in Table 5, both textual semantic enrichment and visual geometric enhancement improve the CAT-Seg baseline under the same Setting 1 protocol. Adding the LLM-guided Category Expansion Module improves the average performance from 62.9% to 64.0% mIoU, indicating that UAV-aware textual descriptions provide more discriminative category prototypes than raw class names. Adding the DINO-enhanced Geometric Feature Fusion Module achieves a slightly larger gain, improving the average mIoU to 64.4%, which suggests that geometric cues are especially useful for dense localization in UAV scenes. When the two modules are combined, UAV-OVSeg reaches 65.5% mIoU and 78.0% mACC, which are consistent with the overall results reported in Table 1. These results show that the two modules are complementary: the text branch mainly improves category-level recognition, while the visual branch strengthens spatial localization and boundary delineation.

Table 5. Component ablation of UAV-OVSeg under **Setting 1**. The results are reported as the overall average across all target datasets in Table 1. A checkmark indicates that the corresponding module is used, and bold values indicate the best results.

Method	LLM Exp.	DINO Geo.	mIoU	mACC
CAT-Seg Baseline [12]			62.9	74.8
Baseline + LLM Exp.	✓		64.0	76.1
Baseline + DINO Geo.		✓	64.4	76.8
UAV-OVSeg	✓	✓	65.5	78.0

More specifically, the LLM expansion alone contributes a 1.1 mIoU gain, while the DINO geometric branch alone contributes a 1.5 mIoU gain. Their combination brings a 2.6 mIoU improvement over the CAT-Seg baseline, which is larger than the gain from either single branch. This additive trend indicates that the two modules address different failure sources: semantic ambiguity from insufficient prompts and localization errors from coarse visual features.

As illustrated in Figure 5, the original category-specific cost maps often respond to only partial object regions or activate nearby background areas. After geometric enhancement, the high-response regions become more compact and better aligned with object structures. This supports the claim that DINO fusion improves dense localization at the cost-map level, before final mask decoding.

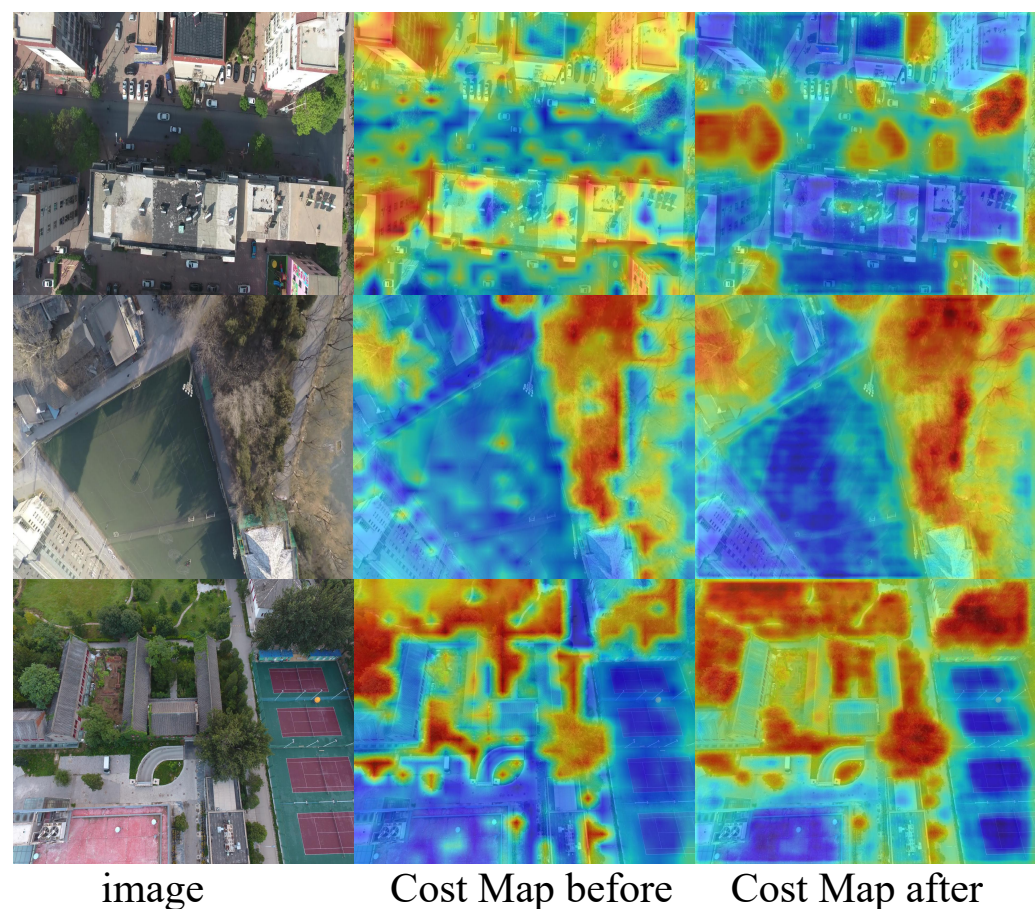


Figure 5. Visualization of category-specific cost maps before and after geometric enhancement. Compared with the original cost maps, the enhanced cost maps show more coherent object responses and clearer structural boundaries, indicating that DINO-based geometric cues improve dense visual-text matching in UAV scenes.

Figure 6 shows representative qualitative cases where the baseline produces fragmented masks or misses thin and small structures. UAV-OVSeg gives more complete regions for roads and vegetation and more stable predictions for small man-made objects. The qualitative improvement is consistent with the later scale-wise and localization analyses, where the largest relative improvements are observed for small objects and boundary-sensitive metrics.

Figure 7 further illustrates the remaining difficulty of UAV OVSS under cluttered backgrounds and complex geometric layouts. Compared with CAT-Seg, UAV-OVSeg reduces category inconsistency and produces smoother object boundaries, but ultra-small targets

and visually similar regions may still be challenging. These observations motivate the additional scale-wise, boundary, and failure-case analyses reported in the revised manuscript.

4.8. Effect of LLM-Guided Category Expansion

To analyze the influence of different text prompt strategies, we compare raw class names, hand-crafted templates, LLM-generated attributes, LLM-generated contextual descriptions, and the full LLM-guided category expansion strategy. In this study, the DINO-enhanced geometric fusion module is kept unchanged, so the comparison mainly reflects the effect of different textual prototypes.

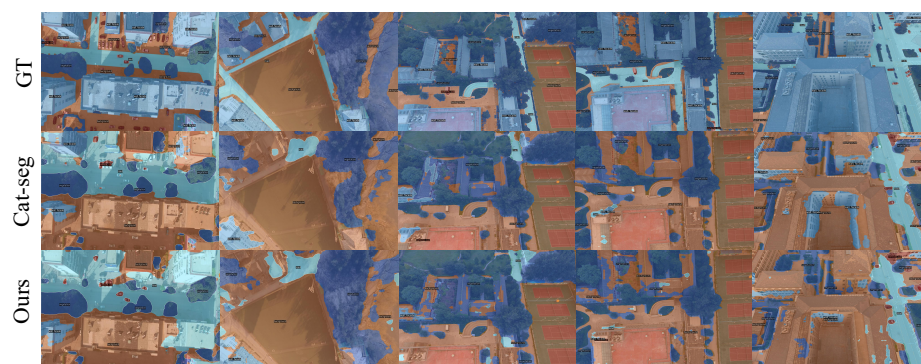


Figure 6. Qualitative comparison between CAT-Seg and the proposed UAV-OVSeg under complex UAV scenes. Compared with CAT-Seg, UAV-OVSeg produces more complete segmentation masks and better preserves object structures, especially for small objects, roads, buildings, and vegetation regions. The colors in the masks denote different semantic categories and are used only for visual distinction.

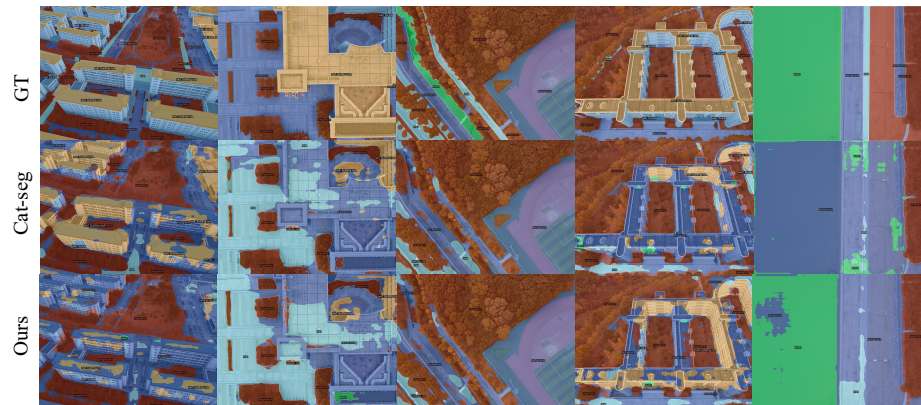


Figure 7. Additional qualitative comparison on challenging UAV scenes. UAV-OVSeg improves boundary quality and category consistency compared with CAT-Seg, particularly in regions with small targets, thin structures, and complex geometric layouts. The mask colors denote semantic categories and are used for visual comparison across methods.

Table 6 shows that prompt design has a clear impact on UAV open-vocabulary segmentation. Using only raw class names achieves 64.4% mIoU, which corresponds to the setting where DINO-enhanced visual features are used without textual expansion. Hand-crafted templates bring a small improvement, but their gain is limited because they provide generic linguistic patterns rather than UAV-specific visual knowledge. LLM-generated attributes and contextual descriptions further improve performance, suggesting that both appearance-level cues and aerial-scene relations are useful for constructing more informative textual prototypes. The full LLM expansion achieves the best result by jointly incorporating related concepts, visual attributes, geometric properties, and UAV-scene context. This result indi-

cates that richer textual descriptions can reduce category ambiguity, especially for broad or visually similar UAV categories such as *road*, *building*, *vegetation*, and *vehicle*.

The gradual improvement from raw class names to full LLM expansion also shows that the gain is not merely caused by adding longer text. Attribute-only and context-only prompts improve different aspects of the textual prototype, while the full structured description performs best because it preserves category identity and adds UAV-specific appearance, geometry, and scene-context cues simultaneously.

Table 6. Effect of different text prompt strategies under **Setting 1**. The DINO-enhanced geometric fusion module is kept unchanged, and the results are reported as the overall average across all target datasets in Table 1. Bold values indicate the best results.

Prompt Strategy	mIoU	mACC
Raw class name	64.4	76.8
Hand-crafted template	64.7	77.1
LLM attributes only	65.0	77.4
LLM context only	64.9	77.3
LLM attributes + context	65.2	77.7
Full LLM expansion	65.5	78.0

4.9. Effect of DINO-Enhanced Geometric Fusion

To evaluate the visual branch, we compare CLIP-only features, DINO-only features, simple summation, feature concatenation, gated fusion, and the full geometry-aware fusion strategy. In this study, the full LLM-guided category expansion strategy is used by default, so the comparison focuses on different ways of integrating CLIP semantics and DINO geometry. CLIP features provide strong language alignment but usually have coarse spatial responses. DINO features provide stronger local structures and geometry-aware representations [19], but they are not directly aligned with textual category descriptions. Therefore, the key problem is how to effectively combine CLIP-based open-vocabulary semantics with DINO-based geometric cues.

The results in Table 7 show that CLIP-only features achieve 64.0% mIoU when the full LLM-guided textual expansion is used, which is consistent with the “Baseline + LLM Exp.” result in Table 5. DINO-only features perform worse because they contain useful local structures but lack explicit open-vocabulary language alignment. Simple summation and concatenation improve over the single-feature settings, showing that CLIP and DINO provide complementary information. However, fixed fusion strategies cannot adaptively determine whether semantic alignment or local geometry should dominate at different spatial locations. The gated fusion strategy further improves performance by introducing adaptive feature weighting. The full geometry-aware fusion obtains the best average result, reaching 65.5% mIoU and 78.0% mACC, which matches the final UAV-OVSeg result in Table 5 and Table 1. These results suggest that DINO-based geometric cues are most effective when they are selectively integrated with CLIP features and category-aware cost maps.

Table 7. Effect of different visual feature fusion strategies under **Setting 1**. The full LLM-guided category expansion strategy is used by default, and the results are reported as the overall average across all target datasets in Table 1. Bold values indicate the best results.

Feature Strategy	mIoU	mACC
CLIP visual feature only	64.0	76.1
DINO feature only	63.5	75.6
CLIP + DINO summation	64.7	77.2
CLIP + DINO concatenation	65.0	77.5
CLIP + DINO gated fusion	65.1	77.1
Full geometry-aware fusion	65.5	78.0

The comparison also explains why the initial cost map S is used in the gate. Without category guidance, simple summation or concatenation improves only moderately because the model cannot determine where semantic alignment or local geometry should dominate. The full geometry-aware fusion uses S as a fixed semantic prior and then recomputes the enhanced cost maps, avoiding a circular dependency while making DINO information category-aware.

4.10. Additional Robustness and Reproducibility Analyses

In response to the reviewers' suggestions, we add additional analyses without changing the original main comparison tables. These experiments focus on LLM choice, prompt filtering, object scale, localization quality, consistency-loss sensitivity, statistical stability, and computational overhead.

Table 8 compares different LLM families while keeping the trainable segmentation network unchanged. Qwen-family and GPT-4-family prompts lead to very similar performance, while Llama-family prompts yield slightly lower performance but are still competitive. This indicates that UAV-OVSeg is not tied to one proprietary LLM; the LLM mainly affects offline prompt quality and does not change the online model size or inference pipeline.

Table 8. Sensitivity to different LLM families for category expansion under Setting 1. Bold values indicate the best results.

LLM for Prompt Generation	Trainable Seg. Params	mIoU	mACC
Llama-family	Unchanged	65.0	77.4
Qwen-family	Unchanged	65.5	78.0
GPT-4-family	Unchanged	65.6	78.1

Table 9 explains the choice of the retained prompt number in Equation (10). Using only one prompt is more sensitive to a single LLM generation, while retaining all five candidates may introduce redundant or weakly related descriptions. The default value $K = 3$ provides the best balance between semantic diversity and category consistency, and is therefore used in the main experiments.

Table 9. Sensitivity to the retained prompt number K in Equation (10), with $K' = 5$ candidate prompts. Bold values indicate the best results.

Retained K	mIoU	mACC
1	65.0	77.3
2	65.3	77.7
3	65.5	78.0
5	65.2	77.6

Table 10 directly addresses the concern about small UAV objects. Compared with CAT-Seg, the DINO-enhanced branch improves Small IoU from 43.8% to 47.6%, and the full UAV-OVSeg further improves it to 49.1%. The gain on small objects is larger than the gain on large regions, confirming that local geometric cues are especially beneficial for tiny targets and thin structures in UAV imagery.

Table 10. Object-scale evaluation under Setting 1. Small, medium, and large regions are grouped according to the mask area after resizing to the evaluation resolution. Bold values indicate the best results.

Method	Small IoU	Medium IoU	Large IoU
CAT-Seg [12]	43.8	61.7	73.2
Baseline + DINO Geo.	47.6	63.4	73.8
UAV-OVSeg	49.1	64.6	74.4

Table 11 complements the visual examples with quantitative localization evidence. UAV-OVSeg improves Boundary IoU from 58.4% to 61.7% and Small-object IoU from 43.8% to 49.1%. These two metrics are more sensitive to boundary quality and tiny target recovery than overall mIoU, so they provide direct support for the effectiveness of DINO-enhanced geometric fusion.

Table 11. Additional localization metrics for boundary quality and small-object segmentation. Bold values indicate the best results.

Method	Boundary IoU	Small-Object IoU
CAT-Seg [12]	58.4	43.8
UAV-OVSeg	61.7	49.1

Table 12 shows that the cost-consistency term is useful but should not be over-weighted. Removing it reduces performance because the DINO-enhanced branch may drift away from CLIP-compatible semantic alignment. A moderate value of $\lambda_{\text{cons}} = 0.10$ gives the best result, while a large value such as 0.50 overly constrains the fused representation and weakens the benefit of geometric enhancement.

Table 12. Sensitivity analysis of the cost-consistency weight λ_{cons} . Bold values indicate the best results.

λ_{cons}	mIoU	mACC
0	65.0	77.2
0.05	65.3	77.6
0.10	65.5	78.0
0.20	65.2	77.5
0.50	64.7	76.9

Table 13 reports both statistical stability and computational overhead. The standard deviations are small across three runs, indicating that the 2.6 mIoU improvement over CAT-Seg is larger than the observed stochastic variation. The online cost increases from 92 ms/image to 126 ms/image, and GPU memory increases from 7.8 GB to 9.6 GB, mainly due to DINO encoding. Since the LLM prompts are generated offline and cached, they do not affect online latency.

Table 13. Statistical robustness and computational overhead. Performance is reported as mean $\pm$ standard deviation over three runs under Setting 1.

Method	mIoU	mACC	GPU Mem.	Inference Time
CAT-Seg [12]	62.9 $\pm$ 0.2	74.8 $\pm$ 0.3	7.8 GB	92 ms/img
UAV-OVSeg	65.5 $\pm$ 0.2	78.0 $\pm$ 0.2	9.6 GB	126 ms/img

Note. Inference time and GPU memory are measured on a single NVIDIA GeForce H100 GPU with a 512×512 input resolution and a batch size of 1. The LLM-generated textual prototypes are precomputed and cached before online inference.

Overall, the additional results show that the reported gains are stable across random seeds and are not solely caused by a single prompt generation. The scale-wise results further verify that the DINO-enhanced branch brings the largest relative improvement on small objects, which is consistent with the motivation of introducing local geometric cues. The efficiency analysis also shows that the LLM does not affect online latency after prompt caching; the main overhead comes from DINO encoding and the lightweight projection/fusion module.

4.11. Qualitative Analysis

Figure 5 further visualizes the effect of geometric enhancement. Before enhancement, the cost maps tend to contain fragmented responses and noisy activations around object boundaries. After introducing DINO-based geometric cues, the responses become more spatially coherent, and object structures are better preserved. This visualization suggests that the visual branch improves spatial localization and boundary consistency, rather than merely changing category-level confidence scores.

Qualitative results are shown in Figure 6. Compared with CAT-Seg [12], UAV-OVSeg produces more complete and spatially coherent segmentation masks in complex UAV scenes. CAT-Seg tends to generate fragmented predictions when objects are small, densely distributed, or visually similar to surrounding regions. In contrast, our method better preserves the shapes of roads, buildings, vegetation regions, and small man-made objects.

Figure 7 further presents additional qualitative examples. The proposed method shows more stable predictions in challenging regions with small targets, thin structures, and complex geometric layouts. These improvements are consistent with the role of DINO-enhanced geometric fusion, which introduces local structural cues into dense visual-text matching.

Taken together, Figures 5–7 provide qualitative evidence that the proposed improvement is not limited to category-level confidence. The cost-map visualization shows feature-level localization refinement, while the mask examples show output-level improvements in completeness, boundary consistency, and small-object recovery. This qualitative trend is aligned with the scale-wise and boundary IoU results, strengthening the mechanistic explanation of the DINO-enhanced module.

Figure 8 provides direct visual evidence for the effect of LLM-guided category refinement. Without LLM refinement, the category responses are relatively diffuse and contain activations in visually similar surrounding regions. After introducing UAV-aware descriptions, the responses become more concentrated on semantically relevant regions, while several confusing background activations are suppressed. This controlled comparison complements the quantitative prompt ablation in Table 6 and directly supports the conclusion that structured LLM descriptions improve category discrimination rather than merely increasing prompt length.

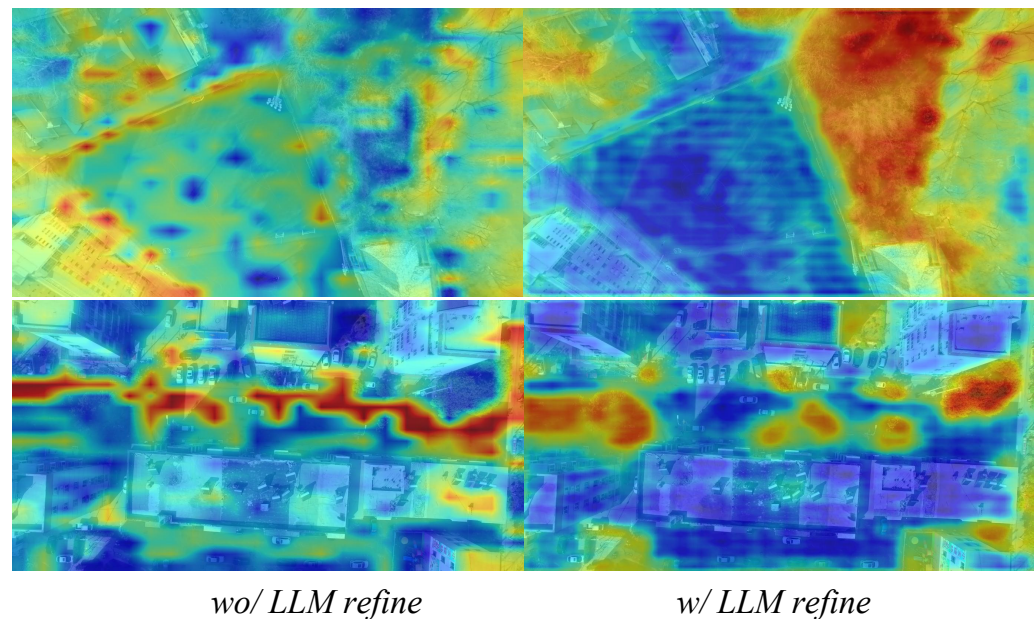


Figure 8. Visualization of category-specific response maps without and with LLM-based semantic refinement. Compared with raw category prompts, LLM-refined descriptions produce more concentrated responses on category-relevant regions and suppress confusing background activations. Warmer colors indicate higher category-response values, and cooler colors indicate lower response values.

5. Discussion

This section discusses the mechanisms, comparison with existing methods, and limitations of the proposed UAV-OVSeg framework.

5.1. Mechanistic Interpretation

The improvements achieved by UAV-OVSeg come from two complementary mechanisms. LLM-guided category expansion modifies the text side by replacing short class names with structured UAV-aware descriptions. This increases the semantic separation between visually ambiguous classes, such as *road*, *sidewalk*, *roof*, and *building*, while still preserving the original category identity through CLIP-based consistency filtering. DINO-enhanced fusion modifies the visual side by injecting local geometric cues into dense visual–text matching. As a result, the cost maps become sharper around object boundaries and more coherent for small or thin structures. The prompt-strategy ablation and category-specific response maps provide direct evidence that semantic enrichment improves discrimination for broad or ambiguous classes, whereas the scale-wise and localization analyses show larger gains for small objects and boundary-sensitive regions.

5.2. Comparison with SOTA Methods

Compared with CAT-Seg, UAV-OVSeg keeps the same dense cost-aggregation principle but introduces UAV-aware textual prototypes and geometry-aware DINO-CLIP fusion. Compared with SAN and OVSeg, our method is designed for cross-dataset UAV transfer rather than only natural-image OVSS. Compared with recent remote-sensing OVSS methods, the proposed benchmark explicitly organizes multiple UAV datasets through category harmonization, source–target transfer, seen- and unseen-category analysis, and reproducible prompt templates. Therefore, the contribution is not simply an engineering combination of an LLM, CLIP, and DINO, but a UAV-specific benchmark–method formulation.

5.3. Factors Affecting Performance

Several factors influence UAV-OVSeg performance. First, prompt quality affects category-level recognition. Overly generic or overly broad LLM descriptions can introduce semantic drift, which is why we use top- K semantic filtering and prompt caching. Second, object scale strongly affects dense prediction. The scale-wise evaluation shows that small objects show the largest gain from DINO fusion, but ultra-small objects remain difficult. Third, DINO-CLIP feature alignment affects the balance between open-vocabulary semantics and local geometry. The consistency loss stabilizes this process, while an excessively large λ_{cons} can restrict geometric adaptation. Finally, cross-dataset domain shift and category overlap affect whether an evaluation is cross-domain open-vocabulary transfer or strictly unseen-category recognition; the revised benchmark explicitly reports this distinction.

5.4. Limitations and Failure Cases

Despite the improvements, UAV-OVSeg still has limitations. The DINO encoder increases online computational cost, which may pose challenges for real-time UAV deployment without model compression or keyframe-based inference. LLM-generated descriptions may occasionally contain weakly related contextual information, especially for rare or domain-specific categories. Moreover, extremely small targets, severe shadows, motion blur, and dense clutter can still cause incomplete masks or category confusion. Future work will explore lightweight DINO distillation, uncertainty-aware prompt selection, higher-resolution geometric fusion, and large-scale UAV-specific vision–language pre-training.

6. Conclusions

In this paper, we studied open-vocabulary semantic segmentation for UAV imagery from both benchmarking and methodological perspectives. We constructed a Unified UAV Open-Vocabulary Semantic Segmentation Benchmark by reorganizing multiple UAV semantic segmentation datasets into a cross-dataset evaluation protocol, providing a practical testbed for assessing open-vocabulary generalization under UAV domain shifts. Based on this benchmark, we proposed UAV-OVSeg, a UAV-oriented OVSS framework built upon a CAT-Seg-style baseline. It improves dense visual–text matching through two complementary modules: an LLM-guided Category Expansion Module that enriches raw class names with UAV-aware textual descriptions, and a DINO-enhanced Geometric Feature Fusion Module that introduces fine-grained geometric cues for better boundary delineation and small-object localization. Experiments on the unified UAV OVSS benchmark show that UAV-OVSeg consistently improves cross-dataset segmentation performance. Under SynDrone training, UAV-OVSeg achieves 65.5%/78.0% mIoU/mACC, outperforming CAT-Seg by 2.6 mIoU and 3.2 mACC. Under AerialScapes training, it achieves 64.5%/76.8% mIoU/mACC, improving upon CAT-Seg by 2.6 mIoU and 3.1 mACC. Ablation studies further verify the complementarity of textual semantic enrichment and visual geometric enhancement. Future work will extend the benchmark to more UAV datasets and explore more efficient text–visual enhancement strategies for real-time UAV deployment.

Author Contributions: Conceptualization, K.W., W.L. and X.L.; methodology, K.W. and X.L.; software, K.W.; validation, K.W., D.H. and J.Y.; formal analysis, K.W. and X.L.; investigation, K.W.; resources, W.L. and X.L.; data curation, K.W.; writing—original draft preparation, K.W.; writing—review and editing, W.L., X.L., D.H. and J.Y.; visualization, K.W.; supervision, W.L. and X.L.; project administration, W.L. and X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The proposed Unified UAV OVSS Benchmark is constructed using publicly available UAV semantic segmentation datasets. The original datasets can be accessed from their official repositories: SynDrone (<https://github.com/LTTM/Syndrone?tab=readme-ov-file>, accessed on 9 November 2025), AEROSCAPES (<https://github.com/ishann/aeroscapes>, accessed on 9 November 2025), VDD (<https://github.com/RussRobin/VDD>, accessed on 9 November 2025), UDD6 (<https://github.com/MarcWong/UDD/tree/master>, accessed on 9 November 2025), UAVid (<https://research.utwente.nl/en/datasets/uavid-dataset/>, accessed on 9 November 2025), and ICG Drone (https://fmi-data-index.github.io/semantic_drone.html, accessed on 9 November 2025). The original datasets remain subject to their respective licenses and distribution policies. To support reproducibility, we will publicly release the unified category mappings, source–target transfer configurations, category-overlap definitions, benchmark split files, dataset conversion scripts, complete LLM prompt templates, cached category descriptions, evaluation scripts, and UAV-OVSeg implementation.

Acknowledgments: The authors would like to thank all contributors for their helpful discussions and technical support for this work.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

UAV	Unmanned Aerial Vehicle
OVSS	Open-Vocabulary Semantic Segmentation
VLM	Vision–Language Model
LLM	Large Language Model
CLIP	Contrastive Language–Image Pre-Training
mIoU	mean Intersection over Union
mACC	mean Accuracy
GPU	Graphics Processing Unit
JSON	JavaScript Object Notation

Appendix A. LLM Prompt Template and Generated Examples

For reproducibility, we provide the complete instruction template used for LLM-guided category expansion. The LLM is asked to output a JavaScript Object Notation (JSON)-style structure for each input category, including related concepts, visual attributes, geometric properties, and UAV-scene contextual cues. In the main experiments, we generate $K' = 5$ candidate descriptions and retain $K = 3$ prompts after filtering for semantic consistency in CLIP text space.

Given a list of UAV semantic segmentation categories, expand each category in English for aerial-view open-vocabulary semantic segmentation. For each category, return a JSON object with the following fields:

1. `base_class`: the original category name;
2. `related_concepts`: visually or semantically related class names;
3. `visual_attributes`: color, texture, material, and appearance cues;
4. `geometric_properties`: shape, scale, continuity, orientation, boundary cues;
5. `uav_context`: typical aerial-view spatial context and neighboring objects.

Please keep the description category-specific and avoid concepts that are not visually observable in UAV imagery.

To provide broader coverage of benchmark categories, we report representative outputs for shared classes and dataset-specific classes. The examples are formatted as code blocks to preserve the exact JSON-style structure used by the prompt-generation pipeline.

```
{
  "base_class": "road",
```

```

    "related_concepts": ["street", "highway", "parking lane", "pavement"],
    "visual_attributes": "gray or dark transportation surface with smooth texture",
    "geometric_properties": "long, narrow, continuous, and often linear or curved",
    "uav_context": "usually adjacent to vehicles, sidewalks, buildings, and vegetation"
  }

  {
    "base_class": "vehicle",
    "related_concepts": ["car", "truck", "bus", "van"],
    "visual_attributes": "compact bright or dark object with a roof-like appearance",
    "geometric_properties": "small rectangular region with a relatively clear boundary",
    "uav_context": "usually located on roads, parking areas, or near buildings"
  }

  {
    "base_class": "building",
    "related_concepts": ["house", "warehouse", "apartment", "industrial building"],
    "visual_attributes": "roof-dominated region with regular texture and material patterns",
    "geometric_properties": "rectangular or polygonal footprint with relatively straight edges",
    "uav_context": "commonly surrounded by roads, vegetation, parking areas, or other buildings"
  }

  {
    "base_class": "vegetation",
    "related_concepts": ["grass", "tree", "shrub", "low vegetation"],
    "visual_attributes": "green or yellow-green region with irregular and textured appearance",
    "geometric_properties": "irregular region with fragmented or clustered boundaries",
    "uav_context": "often distributed near roads, buildings, water, or open land"
  }

  {
    "base_class": "roof",
    "related_concepts": ["rooftop", "building top", "cover structure"],
    "visual_attributes": "bright, gray, red, or dark planar surface with repeated texture cues",
    "geometric_properties": "compact polygonal region with straight or piecewise-linear boundaries",
    "uav_context": "located on top of buildings and often adjacent to roads or vegetation"
  }

  {
    "base_class": "drone",
    "related_concepts": ["UAV", "quadcopter", "multirotor aircraft"],
    "visual_attributes": "small dark or bright aerial object with sparse structural appearance",
    "geometric_properties": "very small compact region with thin protruding structures",
    "uav_context": "appears above ground objects or against visually complex aerial backgrounds"
  }
}

```

References

1. Lyu, Y.; Vosselman, G.; Xia, G.S.; Yilmaz, A.; Yang, M.Y. UAVid: A semantic segmentation dataset for UAV imagery. *ISPRS J. Photogramm. Remote Sens.* **2020**, *165*, 108–119. <https://doi.org/10.1016/j.isprsjprs.2020.05.009>.
2. Cai, W.; Jin, K.; Hou, J.; Guo, C.; Wu, L.; Yang, W. Vdd: Varied drone dataset for semantic segmentation. *J. Vis. Commun. Image Represent.* **2025**, *109*, 104429.
3. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
4. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of the Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015*; Proceedings, Part III; Springer: Cham, Switzerland, 2015; pp. 234–241.

5. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495.
6. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848.
7. Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J.M.; Luo, P. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 12077–12090.
8. Xian, Y.; Choudhury, S.; He, Y.; Schiele, B.; Akata, Z. Semantic Projection Network for Zero- and Few-Label Semantic Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8256–8265.
9. Bucher, M.; Vu, T.H.; Cord, M.; Pérez, P. Zero-shot semantic segmentation. In Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019.
10. Li, B.; Weinberger, K.Q.; Belongie, S.; Koltun, V.; Ranftl, R. Language-driven semantic segmentation. *arXiv* **2022**, arXiv:2201.03546.
11. Xu, M.; Zhang, Z.; Wei, F.; Lin, Y.; Cao, Y.; Hu, H.; Bai, X. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *Proceedings of the European Conference on Computer Vision*; Springer: Cham, Switzerland, 2022; pp. 736–753.
12. Cho, S.; Shin, H.; Hong, S.; Arnab, A.; Seo, P.H.; Kim, S.J. Cat-seg: Cost aggregation for open-vocabulary semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 4113–4123.
13. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 8748–8763.
14. Huang, B.; Li, J.; Luan, W.; Tan, J.; Li, C.; Huang, L. Expanding Open-Vocabulary Understanding for UAV Aerial Imagery: A Vision–Language Framework to Semantic Segmentation. *Drones* **2025**, *9*, 155.
15. Li, F.; Zhang, Z.; Wang, X.; Wang, X.; Xu, Y. Bridging Vision Foundation and Vision–Language Models for Open-Vocabulary Semantic Segmentation of UAV Imagery. *Remote Sens.* **2025**, *17*, 3704.
16. Chen, Y.; Wang, Y.; Lu, P.; Chen, Y.; Wang, G. Large-scale structure from motion with semantic constraints of aerial images. In *Proceedings of the Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*; Springer: Cham, Switzerland, 2018; pp. 347–359.
17. Xu, M.; Zhang, Z.; Wei, F.; Hu, H.; Bai, X. Side adapter network for open-vocabulary semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 2945–2954.
18. Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.T.; Parekh, Z.; Pham, H.; Le, Q.; Sung, Y.; Li, Z.; Duerig, T. Scaling up visual and vision-language representation learning with noisy text supervision. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 4904–4916.
19. Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; Joulin, A. Emerging Properties in Self-Supervised Vision Transformers. In Proceedings of the International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021.
20. Liu, F.; Yao, L.; Zhang, C.; Wu, T.; Zhang, X.; Jiang, X.; Zhou, J. Boost uav-based object detection via scale-invariant feature disentanglement and adversarial learning. *IEEE Trans. Geosci. Remote Sens.* **2025**, *63*, 5622113.
21. Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P.H.; et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 6881–6890.
22. Cheng, B.; Misra, I.; Schwing, A.G.; Kirillov, A.; Girdhar, R. Masked - attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2022; pp. 1290–1299.
23. Cheng, B.; Schwing, A.; Kirillov, A. Per - pixel classification is not all you need for semantic segmentation. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 17864–17875.
24. Yao, L.; Liu, F.; Xu, S.; Zhang, C.; Di, S.; Ma, X.; Jiang, J.; Wang, Z.; Zhou, J. UEMM-Air: Enable UAVs to Undertake More Multi-modal Tasks. In Proceedings of the 33rd ACM International Conference on Multimedia, Dublin, Ireland, 27–31 October 2025; pp. 12792–12798.
25. Luo, T.; Liu, Y.; Xu, Y.; Liu, R.; Wang, X.; Zeng, H.; Huang, S.; Zhang, H. Stroke-Based Perception: Discover Novel Oracle Characters. *IEEE Trans. Multimed.* **2026**. <https://doi.org/10.1109/TMM.2026.3685003>.
26. Liu, R.; Luo, T.; Huang, S.; Wu, Y.; Jiang, Z.; Zhang, H. CrossMatch: Cross-View Matching for Semi-Supervised Remote Sensing Image Segmentation. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5650515. <https://doi.org/10.1109/TGRS.2024.3507050>.
27. Yang, K.; Wang, Y.; Luo, T. Assignment-Driven Hash Learning in a Hyper-Semantic Space for On-the-Fly Category Discovery. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Denver, CO, USA, 3–7 June 2026; pp. 11303–11312.

28. Jia, P.; Gao, Y.; Li, W.; Gao, Q.; Pan, F. AerialCLIP: Lightweight Open-Vocabulary Segmentation for UAV-Based Aerial Images. In *Proceedings of the 2025 44th Chinese Control Conference (CCC)*; IEEE: New York, NY, USA, 2025; pp. 8193–8198.
29. Chen, Z.; Xie, Y.; Wei, Y. Toward High-Resolution UAV Imagery Open-Vocabulary Semantic Segmentation. *Drones* **2025**, *9*, 470.
30. Ghiasi, G.; Gu, X.; Cui, Y.; Lin, T.Y. Scaling open-vocabulary image segmentation with image-level labels. In *Proceedings of the European Conference on Computer Vision*; Springer: Cham, Switzerland, 2022; pp. 540–557.
31. Liang, F.; Wu, B.; Dai, X.; Li, K.; Zhao, Y.; Zhang, H.; Zhang, P.; Vajda, P.; Marculescu, D. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Vancouver, BC, Canada, 17–24 June 2023; pp. 7061–7070.
32. Xie, B.; Cao, J.; Xie, J.; Khan, F.S.; Pang, Y. Sed: A simple encoder-decoder for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, 16–22 June 2024; pp. 3426–3436.
33. Cao, Q.; Chen, Y.; Ma, C.; Yang, X. Open-vocabulary high-resolution remote sensing image semantic segmentation. *IEEE Trans. Geosci. Remote Sens.* **2025**, *63*, 5620614.
34. Li, K.; Liu, R.; Cao, X.; Bai, X.; Zhou, F.; Meng, D.; Wang, Z. Segearth-ov: Towards training-free open-vocabulary segmentation for remote sensing images. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, Nashville, TN, USA, 10–17 June 2025; pp. 10545–10556.
35. Ye, C.; Zhuge, Y.; Zhang, P. Towards open-vocabulary remote sensing image semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*; AAAI Press: Menlo Park, CA, USA, 2025; Volume 39, pp. 9436–9444.
36. Li, K.; Cao, X.; Liu, R.; Wang, S.; Jiang, Z.; Wang, Z.; Meng, D. Annotation-free open-vocabulary segmentation for remote-sensing images. *arXiv* **2025**, arXiv:2508.18067.
37. Dutta, S.; Vasim, A.; Gole, S.; Rezaatofighi, H.; Banerjee, B. AerOSeg: Harnessing SAM for Open-Vocabulary Segmentation in Remote Sensing Images. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, Nashville, TN, USA, 10–17 June 2025; pp. 2254–2264.
38. Li, B.; Huo, T.; Dong, H.; Zhang, D.; Zhao, Z.; Gao, J.; Li, X. Towards Realistic Open-Vocabulary Remote Sensing Segmentation: Benchmark and Baseline. *arXiv* **2026**, arXiv:2604.15652.
39. Yao, L.; Liu, F.; Chen, D.; Zhang, C.; Wang, Y.; Chen, Z.; Xu, W.; Di, S.; Zheng, Y. Remotesam: Towards segment anything for earth observation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, Dublin, Ireland, 27–31 October 2025; pp. 3027–3036.
40. Zhou, C.; Loy, C.C.; Dai, B. Extract free dense labels from CLIP. In *Proceedings of the European Conference on Computer Vision*; Springer: Cham, Switzerland, 2022; pp. 696–712.
41. Zhang, X.; Zhou, C.; Huang, J.; Zhang, L. TPOV-Seg: Textually Enhanced Prompt Tuning of Vision-Language Models for Open-Vocabulary Remote Sensing Semantic Segmentation. *IEEE Trans. Geosci. Remote Sens.* **2025**, *63*, 4420117.
42. Shi, H.; Dao, S.D.; Cai, J. LLMFormer: Large language model for open-vocabulary semantic segmentation. *Int. J. Comput. Vis.* **2025**, *133*, 742–759.
43. Li, B.; Zhang, D.; Huo, T.; Zhao, Z.; Gao, J.; Li, X. An Open-Source Benchmark and Baseline for Multi-temporal Referring Segmentation. *arXiv* **2026**, arXiv:2606.00987.
44. Rizzoli, G.; Barbato, F.; Caligiuri, M.; Zanuttigh, P. Syndrone—Multi-modal uav dataset for urban scenarios. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Paris, France, 2–6 October 2023; pp. 2210–2220.
45. Nigam, I.; Huang, C.; Ramanan, D. Ensemble knowledge transfer for semantic segmentation. In *Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*; IEEE: New York, NY, USA, 2018; pp. 1499–1508.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.