SynDrone – Multi-modal UAV Dataset for Urban Scenarios

Giulia Rizzoli*   Francesco Barbato*   Matteo Caligiuri*   Pietro Zanuttigh

University of Padova

Department of Information Engineering

Via Gradenigo 6/b

{giulia.rizzoli,francesco.barbato,matteo.caligiuri,pietro.zanuttigh}@dei.unipd.it

Abstract

The development of computer vision algorithms for Unmanned Aerial Vehicles (UAVs) imagery heavily relies on the availability of annotated high-resolution aerial data. However, the scarcity of large-scale real datasets with pixel-level annotations poses a significant challenge to researchers as the limited number of images in existing datasets hinders the effectiveness of deep learning models that require a large amount of training data. In this paper, we propose a multimodal synthetic dataset containing both images and 3D data taken at multiple flying heights to address these limitations. In addition to object-level annotations, the provided data also include pixel-level labeling in 28 classes, enabling exploration of the potential advantages in tasks like semantic segmentation. In total, our dataset contains 72k labeled samples that allow for effective training of deep architectures showing promising results in synthetic-to-real adaptation. The dataset will be made publicly available to support the development of novel computer vision methods targeting UAV applications.

1. Introduction

Unmanned aerial vehicles (UAVs) have revolutionized various applications, including surveillance [1, 2], monitoring [3, 4], agriculture [5], and mapping [6]. In particular, UAVs have shown great potential in urban scene analysis, enabling tasks such as traffic control [3], population assessment [7, 1], urban greenery maintenance [8], and road marking extraction [9]. Despite the availability of UAV datasets for detection, tracking, and behavior analysis, there remains a lack of comprehensive datasets specifically designed for densely-annotated tasks such as semantic segmentation. The existing UAV semantic segmentation datasets [10, 11, 12, 13], have limitations in terms of size, scene variation, sampling rate, and class labeling set. Moreover, most of them do not account for the multiple possible operating altitudes and camera angles. To address
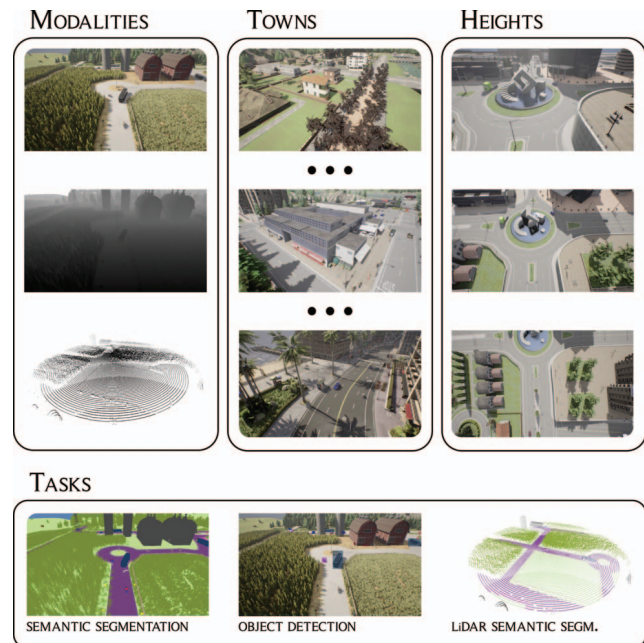

Figure 1: SynDrone is a comprehensive dataset aggregating multi-sensor, multi-location, and multi-altitude data with different task-specific labeling.

these limitations, we present a new synthetic drone dataset, called **SynDrone**[1], which contains significantly larger image sets that capture scenes with a broader variability and increased complexity. In addition to traditional remote sensing images, UAVs equipped with Light Detection and Ranging (LiDAR) and depth cameras have emerged as powerful tools for high-precision positioning systems [9]. These UAVs can dynamically explore uncharted territories, capturing both 3D scans and aerial images in real-time. This capability enables the development of advanced vision tasks, such as height map generation and 3D scene understanding,

---

[1]The code used for analyses and generation of the dataset, and links for its download are available at https://github.com/LTTM/Syndrone.

which can contribute to the accurate analysis and interpretation of the surrounding environment. To this extent, together with standard RGB images `SynDrone` includes co-registered depth maps and LiDAR data, providing the capability for multi-modal analysis as well as supporting other tasks such as depth estimation, navigation, and 3D reconstruction. Moreover, `SynDrone` also enables other vision tasks such as object detection and instance tracking. The dataset includes bounding box annotations and object identifiers, making it suitable for multi-tasking approaches.

In summary, we introduce several contributions:

1. We propose **SynDrone**, a synthetic large-scale multi-modal dataset providing drone imagery with a high-sampling rate at variable altitudes and view angles.

2. We provide registered color, depth, and LiDAR data allowing the development of multi-modal schemes.

3. We provide ground truth data for multiple tasks, including bounding boxes, object class labels, and semantic segmentation labeling, hence the dataset is suitable for object detection and tracking but also for more advanced pixel-level recognition applications.

4. We showcase benchmark results with different models and evaluate the suitability of the data for transfer learning by testing the trained models on real-world datasets.

2. Related Work

In this section, we provide a comprehensive overview of existing methods and datasets focusing on drone imagery in image-level tasks within urban scenarios. A summary of the datasets for object-level tasks can be found in Table 1, while in Table 2 is reported the equivalent for pixel-level tasks. Notice that we focus on works targeting drone-level applications (*i.e.* with a flying altitude below 100m), there is also a wide body of research tackling satellite or high-altitude flying data that represents a different field.

2.1. Vision in UAV urban scenarios

Drone technology has witnessed significant advancements in recent years, revolutionizing urban planning and management. The intersection of urban scene analysis tasks and computer vision encompasses various subdomains, including detection [14, 15, 16], trajectory prediction [17, 18, 19, 20], depth estimation [21, 22, 23], and semantic segmentation [24, 25, 26, 27]. Furthermore, the availability of data has garnered significant interest from the community, leading to the exploration of various learning frameworks such as Continual Learning [28], Cross-Domain Learning [29, 30], Few-shot Learning [31, 32] and Multi-modal Learning [33]. Ultimately, in response to the

limitations and challenges associated with UAVs and their operations, researchers have directed their efforts towards addressing critical issues, *e.g.* adversarial attacks [34, 35], and to the development of lightweight architectures [36].

2.2. Object-level UAV datasets for urban scenarios

Object-level drone datasets have a fundamental significance in advancing research and development in various computer vision tasks. These datasets provide annotated images and videos captured from unmanned aerial vehicles (UAVs), enabling the training and evaluation of algorithms for object detection, tracking, and other related applications. We summarize several notable object-level drone datasets specifically designed for urban scenarios, highlighting their key contributions. Refer to Table 1 for details.

Campus [7] is a large-scale dataset designed for multi-object tracking, activity understanding, and trajectory forecasting within the Stanford University campus. The images were captured using a top-down camera mounted on a multirotor drone hovering at a high altitude.

DBT70 [37] consists of 70 video sequences captured from various sources, including drones and YouTube. The dataset contains manually annotated bounding boxes for pedestrians and vehicles.

UAV123 [42] is a benchmark dataset for UAV tracking tasks, consisting of 100+ video sequences. It encompasses data from professional-grade and consumer-grade UAVs as well as simulator-generated data.

VisDrone [38] is a large-scale benchmark dataset containing a significant number of images with the corresponding annotations. The dataset has been expanded and updated over time to include more data and improve its coverage. It covers various environmental conditions, such as different weather conditions (e.g., sunny, cloudy, rainy), different altitudes, and camera viewpoints.

Anti-UAV [43] includes videos of different UAV types flying in various lighting conditions (day and night), light modes (infrared and visible), and diverse backgrounds. It aims to ensure the diversity of data for tracking purposes.

UAVDT [39] contains 100 video sequences captured from a UAV platform in urban areas, including scenes such as highways and T-junctions. The dataset offers annotations for tracking tasks, including object-bounding boxes.

MDOT [41] is designed for multi-drone single-object tracking. It includes video clips captured by two or three drones simultaneously tracking the same target at different daytime.

AU-AIR [40] comprises of images captured by a multirotor drone flying at low altitudes in an urban scenario. It includes bounding boxes for instances of people and vehicles. Moreover, while target IDs are unavailable, the dataset provides image-level metadata, including drone speed, latitude, and longitude.

Name	Year	Task	S/R	MM	# classes	# images	# sequences	Frequency [Hz]	Height [m]	Size [px]	View Angle
Campus [7]	2016	MOT	R	✗	-	930K	100+	-	80	1400x1904	90
DBT70 [37]	2017	SOT	R	✗	-	-	70	-	-	1280×720	variable
VisDrone-Img [38]	2018	DET	R	✗	10	10209	-	-	-	2000x1500	variable
VisDrone-Vid [38]	2018	DET	R	✗	10	40k	96	-	-	3840x2160	variable
VisDrone-SOT [38]	2018	SOT	R	✗	-	139.3k	167	-	-	-	variable
VisDrone-MOT [38]	2018	MOT	R	✗	-	108.3k	96	-	-	3840x2160	variable
UAVDT [39]	2018	DET, SOT, MOT	R	✗	3	∼ 80k (37.2k + 40.7k)	100	30	10-70+	1080×540	front/side/bird
AU-AIR [40]	2020	DET	R	✗	8	32823	8	5	5-30	1920x1080	45 to 90
MDOT [41]	2020	SOT	R	✗	9	259793	155	-	20-100	1280×720	-
UAV123 [42]	2020	SOT	S+R	✗	-	112578	123	30 to 96	5-25	1280×720 to 3840x2160	-
Anti-UAV [43]	2021	SOT	R	✓	1	318	318	25	-	-	-
HIT-UAV [44]	2023	DET	R	✓	4	2898	-	7	60-130	640×512	30 to 90

Table 1: Object-level UAV datasets. S/R = Synthetic/Real, MM = MultiModal. DET = DETection, SOT = Single Object Tracking, MOT = Multiple Object Tracking. - = not applicable or not explicit in the paper.

Name	Year	MM	BB	# classes	# images	# sequences	Frequency [Hz]	Height [m]	Size [px]	View Angle
Aeroscapes [10]	2018	✗	✗	11	3269	141	-	5-50	1280x720	variable
ICG Drone [45]	2018	✓	✗	20	400	-	1	5-30	6000x4000	90
UDD [12]	2018	✗	✗	4	301	-	-	60-100	4096x2160 or 4000x3000	variable
UAVid [11]	2020	✗	✓	8	270	30	0.2	50	4096x2160 or 3840x2160	45
SynDrone (Ours)	2023	✓	✓	28	(60+12)k	24	25	20, 50, 80	1080×1920	30, 60, 90

Table 2: Pixel-level UAV datasets. BB = Bounding Boxes. - = not explicit in the paper.

HIT-UAV [44] is a high-altitude infrared thermal dataset for object detection on UAVs. It contains infrared thermal images extracted from hundreds of videos captured in various scenarios such as schools, parking lots, and playgrounds. The dataset enables the evaluation of object detection algorithms specifically designed for thermal imaging.

2.3. Pixel-level UAV datasets

Pixel-level UAV datasets with semantic segmentation annotations play a crucial role in developing and evaluating algorithms for various applications, including autonomous navigation, scene understanding, and 3D reconstruction. In this section, we overview several pixel-level UAV datasets, highlighting their strengths and limitations (see Table 2).

Aeroscapes [10] The Aeroscapes dataset stands out by its focus on capturing urban scenes using drones, which enables the collection of more diverse and informative data compared to traditional car-mounted cameras. The dataset includes 11 classes and comprises 141 video sequences, with images having a resolution of 1280x720 pixels.

ICG Drone [45] provides a collection of high-resolution imagery captured from a bird's eye view, facilitating a semantic understanding of residential and green urban scenes. It includes more than 20 houses captured at low altitudes. The dataset offers pixel-accurate annotations for 22 classes, covering a wide range of residential area objects and structures (a notable shortage is the lack of road class). Additionally, it provides valuable supplementary data such as fisheye stereo images, thermal images, ground control points, and 3D ground truth.

UAVid [11] distinguishes itself by providing video sequences captured by small UAVs in various locations. This dataset offers labeled images at a lower frame rate (0.2 FPS) and unlabeled images at a higher frame rate (20 FPS). With 30 video sequences comprising a total of 300 labeled images, UAVid offers the opportunity to explore self-supervised learning approaches for semantic segmentation and 3D reconstruction.

Urban Drone Dataset (UDD) [12] specializes in aiding 3D reconstruction tasks using an improved Structure From Motion (SFM) method. With images captured at altitudes between mid and high altitudes, UDD provides a variety of urban scenes from four different cities in China. The dataset offers annotations for 4 semantic classes.

Most real-world datasets lack an adequate quantity of images or only focus on short sequences (see Table 2), making it challenging to train a network capable of generalizing well to different data. The majority have a low sampling rate because annotating each frame is prohibitively expensive. This limitation hampers the potential for leveraging video semantic segmentation. Additionally, many datasets have a restricted range of classes or do not specifically emphasize driving-related categories. For this reason, despite having a wide range of classes, the ICG Drone dataset, which notably does not include the road class, has restricted applicability to driving or monitoring scenarios.

Ultimately, while other existing methods for generating large-scale synthetic aerial data [13] have been proposed, they lack the capability of simulating relevant dynamic elements such as vehicles and pedestrians.

3. The SynDrone Dataset

In this section, we detail the construction and contents of the proposed SynDrone dataset. It is a multimodal synthetic dataset, developed for the task of drone imagery understanding at both object and pixel-level in urban set-
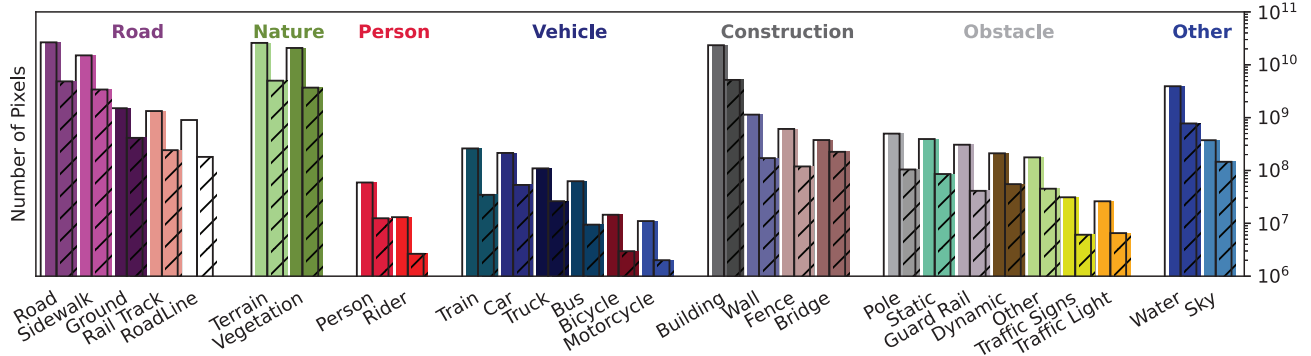
Figure 2: Class distribution in logarithmic scale. Flat bars for the training set, dashed for the test set. Names at the top refer to the coarse grouping, see Table 8 for details.

tings. The dataset contains 72k frames captured from drone views, which are grouped into 8 sequences and further split into 60k images for training and 12k images for testing. The data is densely labeled into 28 semantic classes, with object-level labeling as bounding boxes for the moving objects (vehicles and pedestrians). The data were collected using a modified version of the CARLA simulator [46, 47] (see Section 3.1) at a frequency of 25 Hz and at different heights of 20, 50, and 80 meters above the ground. The images have a resolution of 1920×1080 pixels and are captured from different viewing angles of 30, 60, and 90 degrees. Further detail on the camera sensors and the acquired trajectories are in Section 3.2. The class distribution is shown in Figure 2.

3.1. The CARLA simulator

We decided to employ the CARLA simulator [46], which has been previously used to generate synthetic data in the autonomous driving context [48, 47]. Built upon Unreal Engine 4 (UE4), CARLA offers high-quality rendering, realistic physics powered by NVIDIA PhysX, and basic Non-Player Character (NPC) logic. We employ a modified CARLA 0.9.12 version [47] that provides a diverse range of carefully designed UE4 models, encompassing static objects (e.g., buildings, vegetation, traffic signs) and dynamic objects (e.g., vehicles, pedestrians). These models share a common scale and realistic sizes. The original version includes a blueprint library with 24 car models, 6 truck models, 4 motorbike models, and 3 bike models, each customizable in terms of colors. Additionally, it features 41 pedestrian models of various ethnicities, builds, and attired in a wide array of clothes. Furthermore, CARLA offers 8 meticulously crafted towns (Town01-07 and Town10HD), incorporating over 40 building models. Each town possesses unique features and landmarks, providing 8 simulation environments with distinct visual characteristics. CARLA fa-

cilitates data retrieval from the simulated world through various sensors. These sensors can be precisely positioned, rotated, and attached to parent actors, enabling them to follow rigid or spring-arm-like movements. Sensor data can be collected at each simulation step. When using multiple high-resolution sensors, a synchronous mode ensures that the GPU completes rendering and delivers the data to the client before the subsequent simulation step, guaranteeing a consistent sensor acquisition rate across all sensors. In the modified version, the semantic class set has been extended to ensure compatibility with existing benchmark datasets for autonomous driving [49, 50]. To allow such extension [47] introduced multiple new vehicle models, such as trains, trams, buses, and trucks.

3.2. Acquisition setup

We adopted a camera sensor setup that leverages the capabilities of the CARLA simulator while ensuring data diversity. The acquisition pipeline involves equipping the UAV with multiple co-registered sensors:

RGB camera: It has a resolution of 1920x1080 and enables post-processing effects such as vignette, grain jitter, bloom, auto exposure, lens flare, and depth of field. The vertical Field of View (FoV) is fixed at $90°$, while the viewing direction varies based on the selected flying height, with values of $30°$, $60°$, and $90°$ degrees w.r.t the horizontal axis for altitudes of 20, 50, and 80 meters, respectively. The color images are saved in JPEG format.

Depth camera: The depth camera has the same FoV and resolution as the RGB one. It has a $1km$ maximum range. Depth images are saved in PNG with the format of [51].

Lidar sensor: It is a 64-channel sensor with a $360°$ FoV, operating at bird view ($-89°$ to $-49°$ w.r.t. the horizontal) and collecting $\sim$100k points for each acquisition. The Lidar data is provided for the height of $80m$ and has a maximum range of $100m$. This results in a field of view at road level of
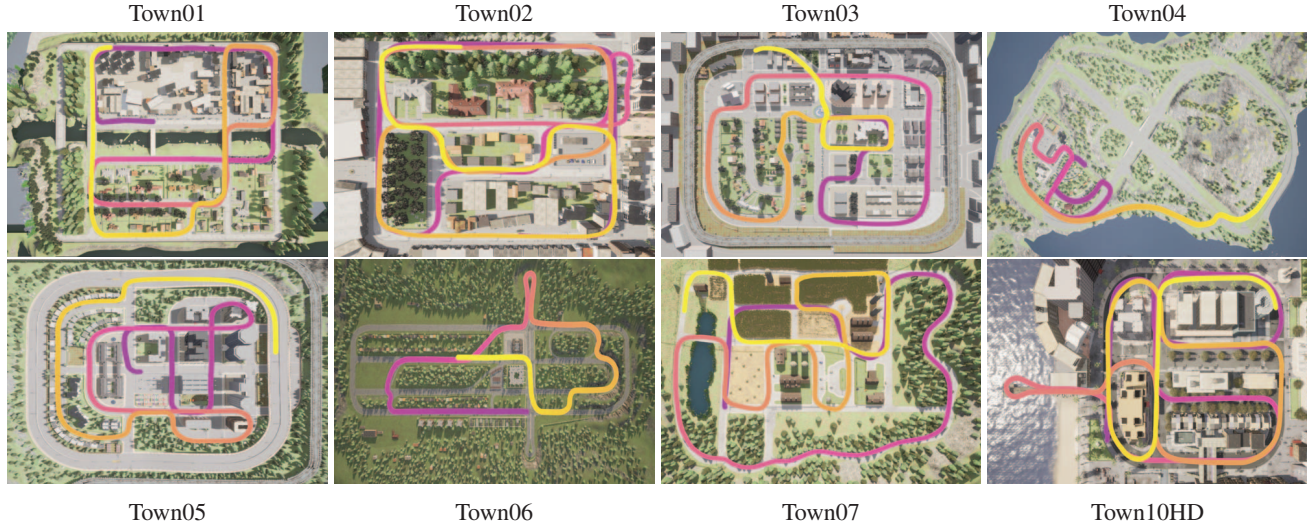
Town01	Town02	Town03	Town04

Town05	Town06	Town07	Town10HD

Figure 3: Trajectories of the drones exploring the 8 different towns. Start (Pink) to Yellow (End).

74°, *i.e.*, 60m of distance from the position perpendicular to the drone. The remaining channels aid in the identification of tall objects (such as buildings or trees). The horizontal resolution is $0.230°$ and the vertical $0.625°$.

The Unreal Engine utilized in SynDrone incorporates a physics engine, specifically based on NVIDIA PhysX, which simulates the movement of vehicles, pedestrians, and sensors. This enables the generation of realistic trajectories and motion blur effects due to drone movements. The velocity vector of the drone is taken into account to calculate the motion blur accurately, providing a more realistic visual representation. The sensor undergoes rotations at a frequency of 25 Hz, which matches the frequency of the cameras used in the dataset. Consequently, each color/depth image has a corresponding full 360° lidar scan.

3.3. Data specifications

The data were acquired by simulating the flight of a drone across the 8 virtual towns. Figure 3 shows the trajectory followed inside each of the towns. Each trajectory has a length of about 2-3km which corresponds to a 2 minutes flight ($\sim 20m/s$). Sensor data is recorded at 25 fps, for a total of 3000 frames for each sensor. To extract testing sequences with a class distribution as close as possible to the training data, while still avoiding too close frames in the train and test sets, we opted to extract 5 equispaced sub-sequences of 100 samples (4 seconds each, 20 seconds in total) from each of the rendered trajectories. This corresponds to a total of $8 * 3 * 500 = 12k$ test samples.

In order to simulate drone trajectories for road surveillance, SynDrone has been designed to mimic real-world scenarios where drones are deployed for applications like monitoring road traffic volume or detecting accidents. In

such cases, the drone's viewpoint can vary as it adjusts its altitude to capture different perspectives. For this reason, to enhance the dataset's robustness and generalization capabilities, SynDrone records data from various heights and view angles. Moreover, we provide ground truth (GT) annotations in both the form of pixel-level semantic maps and 3D bounding boxes with unique identifiers (IDs) for all actors in the scene, including vehicles and pedestrians, at each temporal instant. These annotations enable researchers to perform comprehensive analyses of both semantic segmentation and 2D or 3D object detection methods, enhancing the development of advanced algorithms and systems for UAV-based vision applications.

4. Benchmark and Experiments

We start by reporting some benchmark results of various architectures on our dataset. For consistency and reproducibility, all of the models used the official implementation by the torchvision library[2].

In particular, we employ the widely used **Deeplab-V3** [52] network with both the ResNet50 and MobileNetV2 backbones for the Semantic Segmentation task. The choice of the two backbones follows the idea of having both a highly-performing backbone for server-side computation and a lightweight one that could be used onboard. For the Object Detection task we used **FasterRCNN** [53] and **RetinaNet** [16]. The overall performance and computational cost in terms of MACs (Multiply-Add Cumulation) of the architectures are reported in Table 5. The models were trained for 60k iterations with a batch size of 2. The learn-

---

[2]Pytorch segmentation models available here and the object detection ones here. Accessed 10-July-2023.

Train \ Test	All	20m	50m	80m
All	**61.1**	63.0	60.6	56.2
20m	48.4	**65.1**	46.0	28.1
50m	50.7	41.3	**61.4**	52.5
80m	42.9	25.8	51.8	**57.9**

Table 3: mean Intersection over Union (mIoU) in the semantic segmentation task with data at different altitudes.

Train \ Test	all	t01	t02	t03	t04	t05	t06	t07	t10
all	**61.1**	48	44	58.2	47.1	52.7	41.7	43.1	44
t01	21.2	**55.8**	27.9	16.1	20.3	17.4	15.8	22	6.7
t02	16.8	25.4	**57.5**	12	13.2	12.8	10.8	12.6	7.2
t03	28.3	15.6	19.9	**64.3**	23.4	26.3	24.1	15	15
t04	24.5	17	15.3	21.2	**54.7**	24.8	23	18.3	11.4
t05	25.2	14.2	14.6	25.7	22.8	**58**	25.3	13.8	10.5
t06	16	10.5	9.6	15.1	20.2	17	**48.7**	12.9	8.5
t07	18	15.3	12.6	12.5	20.8	16.3	19.6	**53.1**	4.3
t10	21.6	12.3	13.9	16.6	14.2	17.2	15.8	8	**53.3**

Table 4: mIoU for the semantic segmentation task across different towns (t=town).

Model	Backbone	GMAC [54] @(1080x1920)	mIoU	mAP@50	mAP@75
DeepLabV3 [52]	MNv3	78.9	61.1	-	-
	RN50	1297.22	72.0	-	-
Faster R-CNN [53]	MNv3	8.35	-	31.1	15.7
RetinaNet [16]	RN50	207.94	-	36.2	30.4

Table 5: Comparison of different models over Semantic Segmentation and Object Detection tasks. We also report the computational complexity in terms of MACs. Note for the reader: FLOPs $\simeq 2*$MACs, RN50=ResNet50, MNv3=MobileNetV3 Large.

Data	GMAC[54] @(1080x1920)	mIoU
RGB		61.1
D	78.9	59.1
RGB+D (early)		60.7
RGB+D (late)	161.05	**64.2**

Table 6: Comparison of the training over different modalities.

ing rate was set to 2.5e-4, and a cosine annealing scheduler with a linear warmup for 2000 steps was employed. The semantic segmentation task considers 28 classes, while object detection includes 8 classes, that consist of the the moving objects (*i.e.*, the vehicles and pedestrians). For the object detection task, the rider and motorcycle classes are combined to form the class motorcyclist, while the classes rider and bicycle are merged into the class bicyclist.

Evaluation at different flying altitudes

We performed four different trainings on the model, one for each of the three flying altitudes (20m, 50m, and 80m) and one considering all heights together. In Table 3, we provide a comprehensive overview of the model's performance at different test altitudes. As expected, the model trained on the entire dataset demonstrates the highest overall accuracy when testing on data at all altitudes (61.1%), suggesting that incorporating various heights during training facilitates improved performance across different altitudes. Moreover, the data in the table highlights that altitudes closer to each other exhibit similar performances (with the best performances when training and testing at the same altitude), while the model's generalization tends to decrease as the difference in altitude between training and testing data increases. Furthermore, it can be noticed that the training at the lowest altitude displays a significant drop in accuracy when tested at the highest altitude, achieving a mere 28.1%. As expected, since at higher altitudes the objects appear smaller and thus harder to be recognized,

the model seems to struggle to effectively generalize to this setting, especially when trained at lower altitudes. These observations highlight the importance of considering the interplay between different altitudes.

Evaluation on different towns

To further investigate the model's performance, we conducted comparative training and testing across different towns. By analyzing Table 4, it is evident that the model's accuracy varies significantly across different town scenarios. The highest accuracy is achieved when training and testing are conducted on all towns together, indicating the importance of incorporating more diversified data from a range of towns for improved generalization. On the other hand, the model's accuracy drops considerably when tested on specific towns that were not part of its training set (notice that not all towns have data for all classes and this can impact severely performances if the training town does not include some of the classes in the test one). Interestingly, the results also reveal that certain pairs of towns exhibit reasonable performances (e.g. when trained on town "t01" and tested on town "t02", the model achieves an accuracy of 27.9%), while in other cases performances are very low. This indicates that there might be similarities or shared visual patterns between some towns, allowing the model to generalize well in these particular cases. In general, the heterogeneity in performance between towns highlights the need of gathering and combining data from numerous locations in order to increase the model's capacity to generalize well to previously encountered town settings. The

Setting	Aeroscapes		ICG Drone	UAVid		UDD5		UDD6	
	train	val		train	val	train	val	train	val
Oracle	90.9	71.2	94.1	87.7	70.4	97.1	86.4	97.3	84.8
SynDrone (w/o Resize, w AllClasses)	22.0	27.5	15.6	32.9	33.7	27.2	26.7	26.8	26.2
SynDrone (w Resize, w AllClasses)	22.0	27.5	16.8	36.3	35.6	30.9	30.0	30.2	29.7
SynDrone (w Resize, w/o AllClasses)	24.6	33.1	16.8	51.3	53.6	58.3	57.1	57.2	56.4

Table 7: Performance of models trained on SynDrone and tested on other semantic segmentation datasets. Refer to Table 8 for the class re-mapping. In the latter tests only the valid classes for each specific dataset are considered.

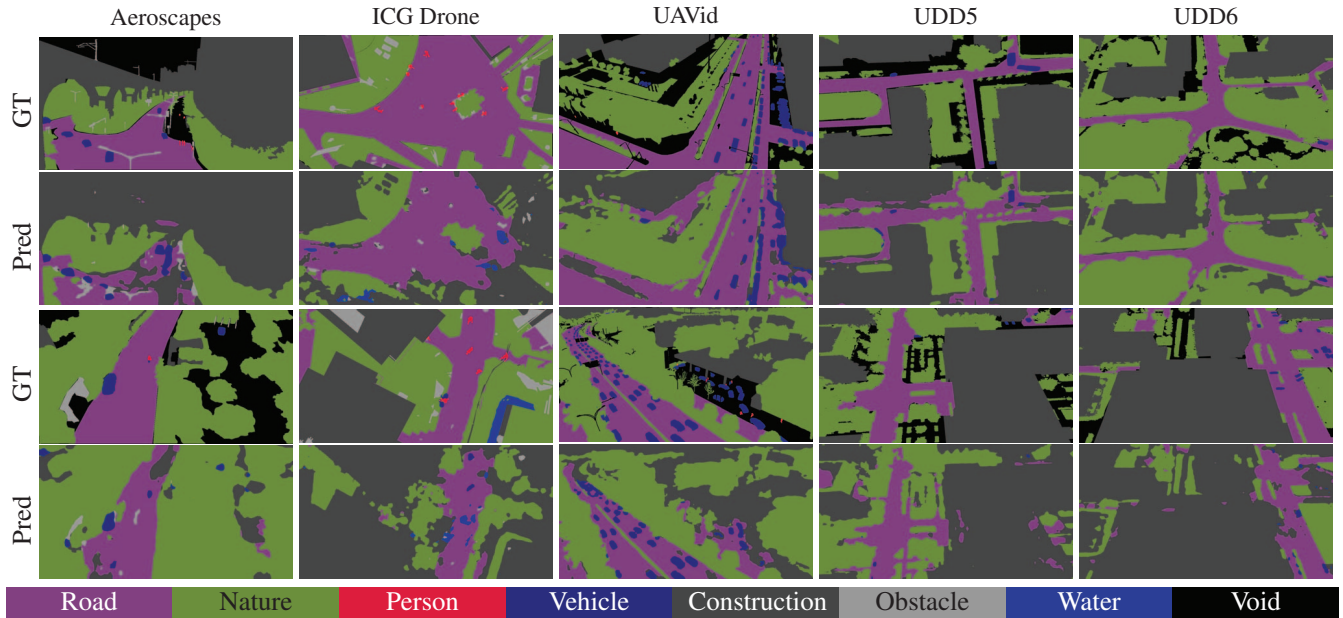| Road | Nature | Person | Vehicle | Construction | Obstacle | Water | Void |

Figure 4: Qualitative results of models trained on SynDrone and tested on real-world data: (GT) Ground-truth semantic map, (Pred) prediction of the model trained on SynDrone. Note that, as generally in semantic segmentation, *void* is ignored during training.

model's performance is influenced by the distinct characteristics and scene variations in each town, highlighting the need for comprehensive training datasets that cover diverse town scenarios.

Evaluation on multi-modal segmentation

Furthermore, we investigate the effectiveness of multi-modal data fusion for the semantic segmentation task. We tested both early and late fusion approaches combining RGB and Depth (D) data (see Table 6). First of all, by looking at the performances of the two modalities alone, it is possible to notice that the mIoU score for the depth is slightly lower than that of color data. Still, similar results suggest that depth information carries useful semantic information as well. The early fusion approach combines the RGB and depth data at the input layer (*i.e.*, a 4-channel

RGBD input), while the late fusion approach performs the fusion at the output stage, requiring twice the computational cost. In particular, for the late fusion, we replicated the whole architecture (decoder included) and opted to merge the predicted logits using a 1×1 convolution, effectively mapping the $2 \times C$ channels into C for the final segmentation prediction. The mIoU score for the latter configuration improves by 3.1% over the RGB results, showing that the multimodal data has a greater information content, although it is important to recognize that it doubles the required operations. Notice that these are just baseline results with naive fusion strategies to offer a starting point for future research. There is a large amount of work on multi-modal segmentation and state-of-the-art strategies will very likely achieve better performances.

Ours (coarse)	Ours (fine)	Aeroscapes	ICG Drone	UAVid	UDD
Road	Road, Ground, Sidewalk, Road Line, Rail Track	Road	Paved Area	Road	Road
Nature	Vegetation, Terrain	Vegetation	Vegetation, Tree, Grass, Dirt, Gravel, Rocks	Vegetation, Tree	Vegetation
Person	Person	Person	Person	Human	
Vehicle	Car, Truck, Bus, Train, Motorcycle, Bicycle	Car, Bicycle	Car, Bicycle	Static Car, Dynamic Car	Vehicle
Construction	Building, Wall, Fence, Bridge	Construction	Roof, Wall, Fence, Window, Door, Fence Pole	Building	Roof, Facade
Obstacle	Other, Pole, Traffic Signs, Guard Rail, Traffic Light, Static, Dynamic	Obstacle	Obstacle		
Water	Water		Water, Pool		

Table 8: Coarse class re-mapping for synthetic-to-real adaptation.

4.1. Synthetic-to-real training

A key aspect in the evaluation of the quality of a synthetic dataset is the capability of models learned on it to perform well on real-world data. Aiming to perform this evaluation on state-of-the-art datasets, we performed a re-mapping of the labels into a common 8 classes set, Table 8 shows how the labels in the different datasets are mapped to our common set. In Table 7 we show the performances of the same model, *i.e.*, DeeplabV3 with MobilenetV3, trained and tested on different datasets. The *Oracle* tests, which assume training and testing on the same dataset, use the same set of parameters as the previous tests with some modifications. Due to the limited size of the datasets, the tests were conducted with 30k iterations to prevent overfitting. Additionally, for all datasets with resolutions ranging from 2-4k, we downscaled the data to full HD while maintaining the original aspect ratio. This adjustment was necessary to ensure compatibility with the network architecture. Notably, for the ICG Drone dataset, since no training and test-

ing splits were provided, the test has not been performed (as such the metric reported for the oracle is basically the training accuracy, which is an overestimation of the performance). Generally, our dataset, without any augmentation or adaptation (*i.e.*, performing *source-only* training), demonstrates good generalization performance across the majority of datasets. However, it faces challenges when tested on more complex datasets, where the accuracy of class mapping is less precise. It is worth mentioning a particular case, *i.e.*, ICG Drone, where the absence of the road class and a focus on non-urban areas, mainly green and residential zones, affect the results. Nevertheless, the model trained on our dataset still achieves promising results in these scenarios, and there is potential for further enhancement by exploring transfer learning and domain adaptation techniques. In figure 4, the qualitative results of the trained model on the real-world data are shown. The reconstruction of semantic maps remains unaffected by factors such as height, viewing angle, or variations in traffic density, encompassing both heavy traffic and sparsely populated roads.

5. Conclusion

In this paper, we introduced a new multimodal synthetic dataset for UAVs, focusing on the costly and scarcely available densely-annotated data. The dataset contains several sequences recorded in different synthetic towns and with a multimodal sensor array, providing ground truth depths, semantic maps, 3D bounding boxes, and semantic LiDAR information. Given the heterogeneous nature of recording heights found in real datasets, we opted to render our samples from three different altitudes (20m, 50m, and 80m) with different camera orientations. In total, our dataset offers 72k samples with pixel-level annotations split into 60k training samples and 12k test samples. We provide multiple benchmark results for semantic segmentation and object detection by training standard networks on our dataset. Additionally, we performed some studies on the generalization capability of the trained architectures when tested on the presence of domain shift (town→town and height→height), highlighting the need for heterogeneous data during training. We also investigate the generalization potential of our dataset in the synthetic-to-real scenario, testing a model trained on our dataset on different real datasets without any explicit adaptation strategies, achieving results that clearly show the potential of the dataset in the task.

In the future, we plan to further extend the dataset including more sensors and a bigger variety of settings. Domain adaptation strategies will be also tested in order to better evaluate the generalization capabilities of the dataset.

References

[1] Naser Hossein. Motlagh, Miloud. Bagaa, and Tarik. Taleb, "Uav-based iot platform: A crowd surveillance use case," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 128–134, 2017.

[2] Hyunbum. Kim, Lynda. Mokdad, and Jalel. Ben-Othman, "Designing uav surveillance frameworks for smart city and extensive ocean with differential perspectives," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 98–104, 2018.

[3] Mouna. Elloumi, Riadh. Dhaou, Benoit. Escrig, Hanen. Idoudi, and Leila Azouz. Saidane, "Monitoring road traffic with a uav-based system," in *2018 IEEE wireless communications and networking conference (WCNC)*. IEEE, 2018, pp. 1–6.

[4] Rodrigo Saar. De Moraes and Edison Pignaton. De Freitas, "Multi-uav based crowd monitoring system," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 2, pp. 1332–1345, 2019.

[5] Saheba. Bhatnagar, Stefano. Puliti, Bruce. Talbot, Joachim Bernd. Heppelmann, Johannes. Breidenbach, and Rasmus. Astrup, "Mapping wheel-ruts from timber harvesting operations using deep learning techniques in drone imagery," *Forestry*, vol. 95, no. 5, pp. 698–710, 2022.

[6] Ziyi. Chen, Cheng. Wang, Jonathan. Li, Nianci. Xie, Yan. Han, and Jixiang. Du, "Reconstruction bias u-net for road extraction from optical remote sensing images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 2284–2294, 2021.

[7] Alexandre. Robicquet, Amir. Sadeghian, Alexandre. Alahi, and Silvio. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*. Springer, 2016, pp. 549–565.

[8] Tanmay Kumar. Behera, Sambit. Bakshi, and Pankaj Kumar. Sa, "Vegetation extraction from uav-based aerial images through deep learning," *Computers and Electronics in Agriculture*, vol. 198, p. 107094, 2022.

[9] Haiyan. Guan, Xiangda. Lei, Yongtao. Yu, Haohao. Zhao, Daifeng. Peng, José Marcato. Junior, and Jonathan. Li, "Road marking extraction in uav imagery using attentive capsule feature pyramid network," *International Journal of Applied Earth Observation and Geoinformation*, vol. 107, p. 102677, 2022.

[10] Ishan. Nigam, Chen. Huang, and Deva. Ramanan, "Ensemble knowledge transfer for semantic segmentation," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 1499–1508.

[11] Ye. Lyu, George. Vosselman, Gui-Song. Xia, Alper. Yilmaz, and Michael Ying. Yang, "Uavid: A semantic segmentation dataset for uav imagery," *ISPRS journal of photogrammetry and remote sensing*, vol. 165, pp. 108–119, 2020.

[12] Yu. Chen, Yao. Wang, Peng. Lu, Yisong. Chen, and Guoping. Wang, "Large-scale structure from motion with semantic constraints of aerial images," in *Pattern Recognition and Computer Vision: First Chinese Conference, PRCV 2018, Guangzhou, China, November 23-26, 2018, Proceedings, Part I 1*. Springer, 2018, pp. 347–359.

[13] Qian. Gao, Xukun. Shen, and Wensheng. Niu, "Large-scale synthetic urban dataset for aerial scene understanding," *IEEE Access*, vol. 8, pp. 42 131–42 140, 2020.

[14] Navaneeth. Balamuralidhar, Sofia. Tilon, and Francesco. Nex, "Multeye: Monitoring system for real-time vehicle detection, tracking and speed estimation from uav imagery on edge-computing platforms," *Remote sensing*, vol. 13, no. 4, p. 573, 2021.

[15] Zhaowei. Cai and Nuno. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6154–6162.

[16] Tsung-Yi. Lin, Priya. Goyal, Ross. Girshick, Kaiming. He, and Piotr. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[17] Caizhen. He, Lanping. Chen, Liming. Xu, Changchun. Yang, Xiaofeng. Liu, and Biao. Yang, "Irlsot: Inverse reinforcement learning for scene-oriented trajectory prediction," *IET Intelligent Transport Systems*, vol. 16, no. 6, pp. 769–781, 2022.

[18] Kaiyang. Zhou, Yongxin. Yang, Andrea. Cavallaro, and Tao. Xiang, "Omni-scale feature learning for person re-identification," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 3702–3712.

[19] Guanshuo. Wang, Yufeng. Yuan, Xiong. Chen, Jiwei. Li, and Xi. Zhou, "Learning discriminative features with multiple granularities for person re-identification," in *Proceedings of the 26th ACM international conference on Multimedia*, 2018, pp. 274–282.

[20] Hao. Luo, Youzhi. Gu, Xingyu. Liao, Shenqi. Lai, and Wei. Jiang, "Bag of tricks and a strong baseline for deep person re-identification," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2019, pp. 0–0.

[21] Logambal. Madhuanand, Francesco. Nex, and Michael Ying. Yang, "Self-supervised monocular depth estimation from oblique uav videos," *ISPRS journal of photogrammetry and remote sensing*, vol. 176, pp. 1–14, 2021.

[22] Mihai. Pirvu, Victor. Robu, Vlad. Licaret, Dragos. Costea, Alina. Marcu, Emil. Slusanschi, Rahul. Sukthankar, and Marius. Leordeanu, "Depth distillation: unsupervised metric depth estimation for uavs by finding consensus between kinematics, optical flow and deep learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3215–3223.

[23] Vlad-Cristian. Miclea and Sergiu. Nedevschi, "Monocular depth estimation with improved long-range accuracy for uav environment perception," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2021.

[24] Xin. He, Yong. Zhou, Jiaqi. Zhao, Di. Zhang, Rui. Yao, and Yong. Xue, "Swin transformer embedding unet for remote sensing image semantic segmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[25] Jia-Xin. Wang, Si-Bao. Chen, Chris HQ. Ding, Jin. Tang, and Bin. Luo, "Ranpaste: Paste consistency and pseudo label for semisupervised remote sensing image semantic segmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2021.

[26] Xinyu. Li, Yu. Cheng, Yi. Fang, Hongmei. Liang, and Shaoqiu. Xu, "2dsegformer: 2-d transformer model for semantic segmentation on aerial images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–13, 2022.

[27] Alina. Marcu, Vlad. Licaret, Dragos. Costea, and Marius. Leordeanu, "Semantics through time: Semi-supervised segmentation of aerial videos with iterative label propagation," in *Proceedings of the Asian Conference on Computer Vision*, 2020.

[28] Valerio. Marsocci and Simone. Scardapane, "Continual barlow twins: continual self-supervised learning for remote sensing semantic segmentation," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2023.

[29] Yansheng. Li, Te. Shi, Yongjun. Zhang, Wei. Chen, Zhibin. Wang, and Hao. Li, "Learning deep semantic segmentation network under multiple weakly-supervised constraints for cross-domain remote sensing image semantic segmentation," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 175, pp. 20–33, 2021.

[30] Bo. Zhang, Tao. Chen, and Bin. Wang, "Curriculum-style local-to-global adaptation for cross-domain remote sensing image segmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–12, 2021.

[31] Mehdi. Khoshboresh-Masouleh and Reza. Shah-Hosseini, "2d target/anomaly detection in time series drone images using deep few-shot learning in small training dataset," in *Integrating Meta-Heuristics and Machine Learning for Real-World Optimization Problems*. Springer, 2022, pp. 257–271.

[32] Emmanouil. Patsiouras, Anastasios. Tefas, and Ioannis. Pitas, "Few-shot image recognition for uav sports cinematography," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 238–239.

[33] Simon. Speth, Artur. Goncalves, Bastien. Rigault, Satoshi. Suzuki, Mondher. Bouazizi, Yutaka. Matsuo, and Helmut. Prendinger, "Deep learning with rgb and thermal images onboard a drone for monitoring operations," *Journal of Field Robotics*, vol. 39, no. 6, pp. 840–868, 2022.

[34] Zhen. Wang, Buhong. Wang, Chuanlei. Zhang, and Yaohui. Liu, "Defense against adversarial patch attacks for aerial image semantic segmentation by robust feature extraction," *Remote Sensing*, vol. 15, no. 6, p. 1690, 2023.

[35] Zhen. Wang, Buhong. Wang, Yaohui. Liu, and Jianxin. Guo, "Global feature attention network: Addressing the threat of

adversarial attack for aerial image semantic segmentation," *Remote Sensing*, vol. 15, no. 5, p. 1325, 2023.

[36] Siyu. Liu, Jian. Cheng, Leikun. Liang, Haiwei. Bai, and Wanli. Dang, "Light-weight semantic segmentation network for uav remote sensing images," *Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 8287–8296, 2021.

[37] Siyi. Li and Dit-Yan. Yeung, "Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, 2017.

[38] Pengfei. Zhu, Longyin. Wen, Dawei. Du, Xiao. Bian, Heng. Fan, Qinghua. Hu, and Haibin. Ling, "Detection and tracking meet drones challenge," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 7380–7399, 2021.

[39] Dawei. Du, Yuankai. Qi, Hongyang. Yu, Yifan. Yang, Kaiwen. Duan, Guorong. Li, Weigang. Zhang, Qingming. Huang, and Qi. Tian, "The unmanned aerial vehicle benchmark: Object detection and tracking," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 370–386.

[40] Ilker. Bozcan and Erdal. Kayacan, "Au-air: A multi-modal unmanned aerial vehicle dataset for low altitude traffic surveillance," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 8504–8510.

[41] Pengfei. Zhu, Jiayu. Zheng, Dawei. Du, Longyin. Wen, Yiming. Sun, and Qinghua. Hu, "Multi-drone-based single object tracking with agent sharing network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 4058–4070, 2020.

[42] Matthias. Mueller, Neil. Smith, and Bernard. Ghanem, "A benchmark and simulator for uav tracking," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer, 2016, pp. 445–461.

[43] Nan. Jiang, Kuiran. Wang, Xiaoke. Peng, Xuehui. Yu, Qiang. Wang, Junliang. Xing, Guorong. Li, Jian. Zhao, Guodong. Guo, and Zhenjun. Han, "Anti-uav: A large multi-modal benchmark for uav tracking," *arXiv preprint arXiv:2101.08466*, 2021.

[44] Jiashun. Suo, Tianyi. Wang, Xingzhou. Zhang, Haiyang. Chen, Wei. Zhou, and Weisong. Shi, "Hit-uav: A high-altitude infrared thermal dataset for unmanned aerial vehicle-based object detection," *Scientific Data*, vol. 10, no. 1, p. 227, 2023.

[45] Graz University. of Technology, "ICG Drone Dataset," http://dronedataset.icg.tugraz.at, [Accessed 07-June-2023].

[46] Alexey. Dosovitskiy, German. Ros, Felipe. Codevilla, Antonio. Lopez, and Vladlen. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*. PMLR, 2017, pp. 1–16.

[47] Paolo. Testolina, Francesco. Barbato, Umberto. Michieli, Marco. Giordani, Pietro. Zanuttigh, and Michele. Zorzi, "Selma: Semantic large-scale multimodal acquisitions in variable weather, daytime and viewpoints," *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[48] Emanuele. Alberti, Antonio. Tavera, Carlo. Masone, and Barbara. Caputo, "Idda: A large-scale multi-domain dataset for autonomous driving," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5526–5533, 2020.

[49] Marius. Cordts, Mohamed. Omran, Sebastian. Ramos, Timo. Rehfeld, Markus. Enzweiler, Rodrigo. Benenson, Uwe. Franke, Stefan. Roth, and Bernt. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[50] Andreas. Geiger, Philip. Lenz, Christoph. Stiller, and Raquel. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[51] German. Ros, Laura. Sellart, Joanna. Materzynska, David. Vazquez, and Antonio M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3234–3243.

[52] Liang-Chieh. Chen, George. Papandreou, Florian. Schroff, and Hartwig. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[53] Shaoqing. Ren, Kaiming. He, Ross. Girshick, and Jian. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[54] Vladislav. Sovrasov, "Ptflops: a flops counting tool for neural networks in pytorch framework," 2018–2023. [Online]. Available: https://github.com/sovrasov/flops-counter.pytorch