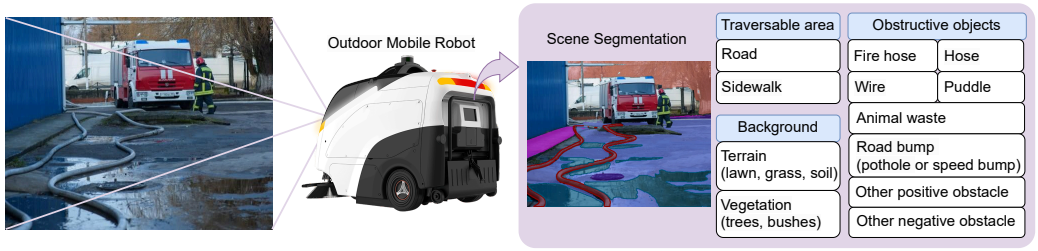
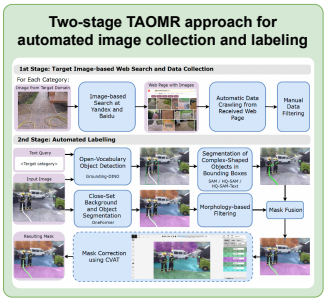
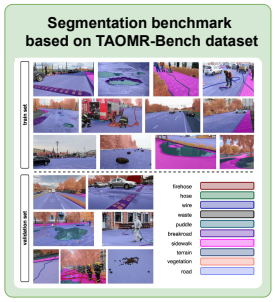
Graphical Abstract

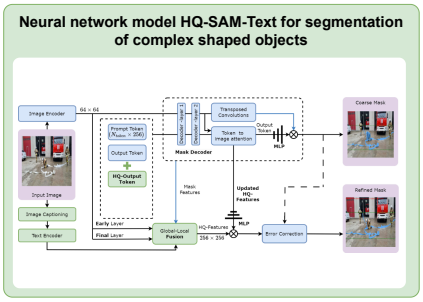
Enhancing Complex Object Segmentation with Text-Guided Refinement

Luka Vetoshkin, Olga Matykina, Margarita Kichik, Uliana Izmesteva, Dmitry Yudin









Highlights

Enhancing Complex Object Segmentation with Text-Guided Refinement

Luka Vetoshkin, Olga Matykina, Margarita Kichik, Uliana Izmesteva, Dmitry Yudin

- We introduce a novel approach for automated dataset collection and labeling called TAOMR (Traversable Area Objects for Mobile Robotics). This approach utilizes state-of-the-art pre-trained open-vocabulary and close-set segmentation models, enabling precise labeling of such obstructive objects as wires, hoses, road bumps, wastes, and puddles, which can significantly impact the safety of the robot's movement.

- Based on the proposed labeling approach, we developed a new large-scale dataset consisting of 75,479 object masks for 8,823 images collected from real mobile robot, Yandex and Baidu search engines, and open datasets of the target domain.

- We developed a novel benchmark for semantic, instance, and open-vocabulary street scene segmentation named TAOMR-Bench which is based on a representative sample of the full dataset. Using it, we evaluated the quality of existing state-of-the-art segmentation models and showed the prospects for their further development.

- For open-vocabulary segmentation improvement, we developed a neural network architecture called HQ-SAM-Text. It allowed us to increase the quality of complex-shaped object segmentation by integrating text descriptions of the objects.

Enhancing Complex Object Segmentation with Text-Guided Refinement

Luka Vetoshkin[a], Olga Matykina[a], Margarita Kichik[a], Uliana Izmesteva[a], Dmitry Yudin[a,b,*]

[a]*Moscow Institute of Physics and Technology, 9 Institutsky per., Dolgoprudny, 141701, Moscow Region, Russia*
[b]*Artificial Intelligence Research Institute (AIRI), 32 Kutuzovsky Ave., Moscow, 121170, Moscow, Russia*

Abstract

In mobile robotics, the challenge of accurately segmenting and recognizing complex objects, particularly those that are rarely encountered in traversable areas of street and park scenes, remains unresolved. Pre-trained object segmentation models often fail to reliably identify obstructive objects such as wires, hoses, bumps, waste, and puddles, which can critically impact the safety and efficiency of robot navigation. In this article, we introduce a novel approach for automated dataset collection and labeling, originally named TAOMR (Traversable Area Objects for Mobile Robotics), now extended with a focus on enhancing complex object segmentation through text-guided refinement. Our method leverages a combination of open-vocabulary and close-set segmentation models, augmented with morphology-based filtering, to accurately label obstructive objects, traversable areas, and background elements. Using this approach, we curated a comprehensive dataset comprising 75,479 object masks across 8,823 images of park environments, sourced from both search engines and real-world mobile robot deployments. To evaluate the performance of modern semantic, instance, and open-vocabulary segmentation methods in this domain, we developed the TAOMR-Bench benchmark. This benchmark has proven to be highly challenging for state-of-the-art neu-

*Corresponding author

Email addresses: lukavetoshkin@yandex.ru (Luka Vetoshkin), omatykina@mail.ru (Olga Matykina), kichik.mg@phystech.edu (Margarita Kichik), izmesteva251@gmail.com (Uliana Izmesteva), yudin@airi.net (Dmitry Yudin)

ral networks, enabling rigorous validation of recognition quality for rare and complex obstructive objects. Furthermore, to address the limitations of existing models, we propose HQ-SAM-Text, an advanced neural network architecture that integrates textual scene encoding into the Segment Anything Model (SAM) with high-quality output tokens. This refinement significantly improves the segmentation of complex-shaped objects, demonstrating the effectiveness of text-guided approaches in enhancing segmentation accuracy and robustness. Our results highlight the potential of combining visual and textual cues for improving object recognition in mobile robotics applications.

Keywords: Automated labeling, neural network, image segmentation, traversable area, obstructive object, mobile robot

---

1. Introduction

In mobile robotics, the problem of identifying specific objects that are rarely found in street scenes and park areas remains unresolved. Existing pretrained object detection and segmentation models struggle to reliably detect obstructive objects like wires, hoses, artificial bumps, wastes, and puddles, which can significantly affect the safety of the ground robot's movement.

This article explores potential solutions to address this issue. In Figure 1, we showed considered segmentation categories of interests including traversable area (road and sidewalk), background (terrain and vegetation), and obstructive objects of several types. These categories were selected because they are the most frequently encountered in scenarios involving the movement of delivery or cleaning robots on city streets and in park areas. Obstructive objects, in particular, can either completely block movement or require adjustments, such as reducing speed or altering the robot's control strategy.

The challenge in segmenting these categories is largely caused by the limited number of suitable datasets, which are either too general (e.g. ImageNet [1], ADE20k [2], COCO-Stuff [3], Elevater [4], etc) or mainly collected for self-driving cars on city roads (e.g. CityScapes [5], Mapillary [6], etc). As a rule, they do not contain specific objects important for mobile ground robots operating in park areas and off-road scenarios.

The rapid development of open-vocabulary segmentation methods such as SAM [7] or SEEM [8], as expected, makes it possible to accelerate the process of creating new specialized datasets. But they also have limitations

and require modernization in the presence of objects of complex shape, when the method confuses the background and the foreground object.

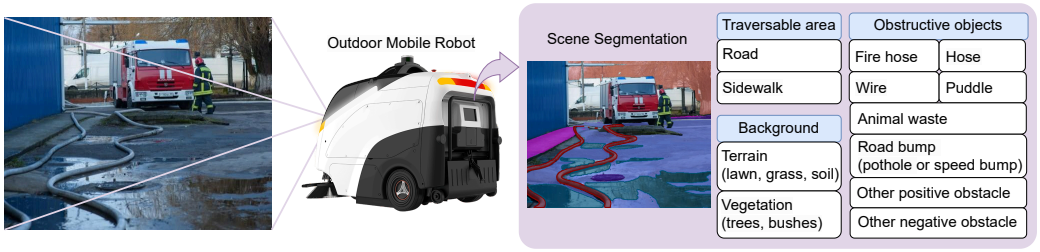


Figure 1: The problem under consideration is scene segmentation in the on-board camera image of a mobile robot (for example, a sweeper robot, a courier robot, etc.). The goal is to identify masks for the traversable area, nearby background objects, as well as obstructive objects, such as hoses, wires, animal waste, puddles, potholes, and artificial road bumps, which have a significant impact on the trajectory and speed of the robot.

An objective study of existing state-of-the-art methods for object segmentation in images on specific benchmarks also poses a separate challenge due to the different image domains. Of particular interest are well-proven convolutional neural network models, for example, YOLO11-Seg [9], and transformer models OneFormer [10], OpenSeed [11], etc.

In this article, we introduce a novel obstructive object labeling approach and segmentation benchmark for the traversable area of mobile ground robots. For this purpose, we collected diverse data from a real outdoor ground mobile robot moving in a park area, images collected from Internet search engines, and data from relevant open datasets.

The main contributions of the article may be summarized as follows:

- We introduce a novel approach for automated dataset collection and labeling called TAOMR (Traversable Area Objects for Mobile Robotics). This approach utilizes state-of-the-art pre-trained open-vocabulary and close-set segmentation models, enabling precise labeling of such obstructive objects as wires, hoses, road bumps, wastes, and puddles, which can significantly impact the safety of the robot's movement.

- Based on the proposed labeling approach, we developed a new large-scale dataset consisting of 75,479 object masks for 8,823 images collected from real mobile robot, Yandex and Baidu search engines, and open datasets of the target domain: NPO [12], InstSnowRoadDetection [13], and RoadObstacle21 [14].

- We developed a novel benchmark for semantic, instance, and open-vocabulary street scene segmentation named TAOMR-Bench which is based on a representative sample of the full dataset. Using it, we evaluated the quality of existing state-of-the-art segmentation models and showed the prospects for their further development.

- For open-vocabulary segmentation improvement, we developed a neural network architecture called HQ-SAM-Text (an acronym for Segment Anything Model with High-Quality output tokens and Text encoding). It allowed us to increase the quality of complex-shaped object segmentation by integrating text descriptions of the objects.

The developed dataset and the basic code are publicly available at the link: `https://github.com/OlgaMatykina/TAOMR`.

2. TAOMR approach to automated dataset collection and labeling

In this work, we propose a two-stage approach for the automated collection and labeling of custom datasets, specifically targeting traversable area objects in on-board camera images of mobile robots operating on city streets and in park areas. The overall process is illustrated in Figure **??**.

In the first stage, Yandex[1] and Baidu[2] image-based search engines are used to obtain data for the desired domain. These engines have an image search function, thanks to which we can find and save images that are as similar as possible to the desired domain. Data collection from the required page was carried out by a program written in the Python programming language using the Selenium[3]. In this case, data selection was carried out manually.

The number of collected images of the desired domain containing certain categories is shown in the second column of Table 1.

For comparison, we also examined a selection of images with relevant categories from existing large-scale datasets like SA-1B [7], containing over 1 billion masks on millions of images. To do this, we used the Semantic Segment Anything (SSA) approach [15], which allows us to assign the names of

[1]Yandex search engine. `https://yandex.ru/images`

[2]Baidu image-based search engine. `https://image.baidu.com`

[3]Selenium library. `https://selenium-python.readthedocs.io/`

Table 1: Number of images obtained in Internet search engines at different stages of the proposed approach

Image-based query categories	Number of images collected at Stage 1	Number of images correctly labeled at Stage 2	Examples of queries for Grounding DINO
Fire hose	681	324	hose, fire hose
Hose	1037	547	hose, wire
Wire	1072	655	hose, wire
Animal waste	183	125	poop, waste,
Puddle	1039	413	puddle, water, pit
Road bump	1022	568	bump,pit, speed bump, manhole

semantic categories to class-agnostic masks of the dataset. However, the SSA engine's image processing was time-consuming, with each annotation taking about 10 seconds, leading to significant delays in dataset creation. Moreover, the annotation quality, especially for critical categories like wires and hoses, often fell short due to the complexity of the objects and limitations of the models, resulting in inaccuracies and inconsistencies in annotations. Despite its efficiency in generating annotated datasets, the SSA engine's processing speed and annotation quality limitations highlight the need for careful evaluation in specific application contexts.

In the second stage, we use the proposed modular automated labeling approach shown at the bottom of Figure **??**. It includes two main module groups: open-vocabulary detection and subsequent segmentation modules for objects specified in text queries, close-set segmentation module with further morphology-based filtering for mask recognition for a fixed list of background and object categories. Next, we implemented the mask fusion module and optional mask correction using the Computer vision annotation tool CVAT[4].

Open-vocabulary object detection module. Object detection for categories such as fire hoses, water hoses, electrical cables, natural waste, puddles, and road surface defects was performed using a pre-trained Grounding-DINO model [16]. This model takes both an image and a text query as input to identify relevant objects.

Module of segmentation of complex-shaped objects in bounding boxes. For object segmentation, we employed the pre-trained HQ-SAM model [17], which is distinguished by its ability to recognize masks of complex-shaped objects. The input to HQ-SAM included the image and

[4]OpenCV CVAT. `https://opencv.github.io/cvat/docs/`

bounding box coordinates provided by Grounding-DINO.

Close-set background and object segmentation module. Semantic segmentation was initially performed using the OneFormer [10] model pre-trained on the Mapillary Vistas [6] dataset. The resulting labels were subsequently remapped to align with the predefined categories of our dataset. This model effectively segments background categories such as public roads, pedestrian sidewalks, and lawns. It was observed that small obstacles on the road were frequently misclassified as part of the road category, as these instances were either not detected or lacked adequate distinction.

To address this, panoptic segmentation was conducted using OneFormer with Ade20k [2] weights. The results showed that in many images, small obstacles either received a distinct label (e.g., "box") or remained unlabeled but were nonetheless delineated by clear contours, preventing them from being included in the road category.

Morphology-based filtering module. Post-processing of the category maps was carried out using morphological operations [18] to refine the segmentation of obstacles. This approach is particularly applicable for labeling a desired list of categories in existing datasets e.g. RoadObstacle21 [14].

Morphological closing followed by morphological opening was applied using a 5×5 kernel to reduce noise and improve the continuity of segment boundaries.

Polygons were then extracted based on their class, specifically focusing on those that did not correspond to the road category. This step aims to highlight small obstacles that are on the road but have the wrong category.

Dilation was applied to the identified segments to define their boundaries.

For segments that were fully enclosed by road pixels, the category "other obstacle positive" was assigned. This reassignment was based on analyzing the border region surrounding each obstacle. If the surrounding area consisted entirely of road pixels, the obstacle was classified under this new category, ensuring that obstacles completely encircled by road were properly identified and labeled.

Finally, the masks of identified obstacles were overlaid onto the original semantic segmentation results obtained in the Close-set segmentation module. This integration allowed for more accurate and comprehensive labeling of the dataset, particularly in areas where small obstacles were originally misclassified as "road".

Mask fusion module. In the Mask fusion module, we combine masks from open-vocabulary and close-set segmentation modules. Besides direct

aggregation we do additional mask processing:

- Filtering masks by their area. A procedure for removing "garbage" masks has been implemented. Specifically, a minimum area threshold was established, and all masks below this threshold were excluded from further processing.

- Merging masks within the same category. To address the issue of mask redundancy, a set of rules for combining masks was formulated. Since the extra masks only had background categories involving large areas (such as roads, vegetation, pedestrian areas, etc.), the strategy adopted was to merge only the background object masks, leaving obstacle object masks unchanged. In this way, we predetermine the categories to be merged and then, independently of other categories, fuse the corresponding background masks and generate a new markup as the output result.

When we further started training semantic segmentation models, in particular OneFormer [10], another requirement for mask fusion became clear. It is unacceptable to overlay several masks on top of each other. There should be only one object class per pixel in the input image. Accordingly, a procedure has been implemented to eliminate the mask overlap with a given category priority.

All of the above procedures were implemented using the functionality of the pycocotools[5] library, which provides tools for working with datasets based on the COCO (Common Objects in Context) format [19]. This library enabled efficient processing of images and their annotations.

Mask correction module using CVAT. Subsequently, the obtained results of segmented objects may be corrected using the CVAT[6] tool. This module is optional but needed for labeling of images obtained from Internet search engines.

The markup obtained during the automatic segmentation of given object categories turned out to be insufficiently correct in such images. It contained some inaccuracies, such as unlabeled objects present in the image, voids and "bald spots" in masks, incorrectly signed object classes, a large number of small masks for background categories, false marking of objects, and merging

[5] `https://github.com/cocodataset/cocoapi`

[6] OpenCV CVAT. `https://opencv.github.io/cvat/docs/`

masks of two adjacent objects. Such labeling errors can significantly impact the performance of the trained models, making manual correction of each mask necessary.

The CVAT tool allows us to resolve all the mentioned errors, except for the issue involving a large number of small masks. The third column of Table 1 shows statistics on correctly labeled categories using sequential inference of Grounding-DINO and HQ-SAM for images obtained in Yandex and Baidu search engines.

The final annotations of the dataset were uploaded in the COCO [19] format, suitable for further training neural network models.

3. TAOMR Dataset

Based on the approach described in the previous section, we formed a large-scale TAOMR Dataset, which contains 8,823 images and 75,479 masks belonging to 12 categories of objects in the traversable area. Detailed information on the distribution of objects within the TAOMR dataset can be found in Table 2.

The dataset have several data sources:

- images collected from real robot (RR), Yandex and Baidu search engines (2nd column of Table 2),
- images from open datasets of target domain: NPO [12], InstSnowRoadDetection [13], and RoadObstacle21 [14] (3rd-5th columns of Table 2).

For datasets such as NPO [12], InstSnowRoadDetection [13], and RoadObstacle21 [14], we augment annotations adapted to our category list. Figure 2 visualizes both the images and corresponding labels from these public datasets. While these datasets mostly lack obstacles specific to mobile robotics, they do include objects whose segmentation is important for autonomous vehicles. To address this, two new categories were introduced: "other-obstacle-negative" and "other-obstacle-positive." This agnostic obstacle labeling allows models to learn segmentation for various obstacles, differing only in their position relative to the roadway plane. The RoadObstacle21 [14] dataset, similar to our domain, contains small road obstacles, while InstSnowRoadDetection [13] consists of images of snowy roads, where our annotations added segmentation of buildings and structures under the "other-obstacle-positive" category. The NPO [12] dataset captures

Table 2: Number of automated labeled images in TAOMR Dataset per data source

Category	RR, Yandex and Baidu		NPO		InstSnowRoadDet		RoadObstacle21		Total	
	Images	Masks	Images	Masks	Images	Masks	Images	Masks	Images	Masks
Fire hose	272	387	0	0	0	0	0	0	272	387
Hose	269	328	0	0	0	0	0	0	269	328
Wire	220	231	0	0	0	0	0	0	220	231
Animal waste	239	1455	0	0	0	0	0	0	239	1455
Puddle	273	520	0	0	0	0	0	0	273	520
Road bump (pothole or speed bump)	2059	3174	2749	13964	84	216	310	2651	5202	20005
Sidewalk	1802	1802	2706	2706	0	0	243	243	4751	4751
Terrain	1753	1753	2839	2839	8	8	320	320	4920	4920
Vegetation	1711	1711	4981	4981	205	205	442	442	7339	7339
Road	2444	2444	5000	5000	240	240	442	442	8126	8126
Other obstacle negative	0	0	4571	10527	2	2	36	130	4609	10659
Other obstacle positive	0	0	3087	11788	194	743	437	4227	3718	16758
Total	3141	13805	5000	51805	240	1414	442	8455	8823	75479

Table 3: Number of images and object masks per category in the TAOMR-Bench dataset

Data	Total	Fire hose	Hose	Wire	Waste	Puddle	Road bump	Sidewalk	Terrain	Vegetation	Road
Training part of TAOMR-Bench											
Masks	6202	59	44	96	508	226	1106	867	896	1640	760
Images	422	40	40	55	45	71	291	296	225	325	291
Validation part of TAOMR-Bench											
Masks	1551	16	11	24	144	43	271	193	267	425	157
Images	107	10	10	14	12	18	79	75	52	81	68

road scenes in various weather conditions, both in urban and rural environments. Its main drawback for autonomous driving tasks was the lack of road surface labeling, which our annotations have addressed.

Based on the publicly available part of the TAOMR Dataset we create a representative image segmentation benchmark named TAOMR-Bench. It consists of 529 images of varying sizes and allows us to evaluate semantic, instance, and open-vocabulary segmentation models. The dataset is split into two subsets: a training set with 422 and a testing set with 107 images. Detailed information on the distribution of objects within the training and test sets can be found in Table 3.

Figure 3 visualizes both the images and their corresponding labels. The dataset captures scenes typical of streets and parks, recorded by the onboard camera of mobile ground robots, such as a courier robot or a cleaning robot. To ensure privacy, any faces present in the images were blurred using the LightFace framework [20].

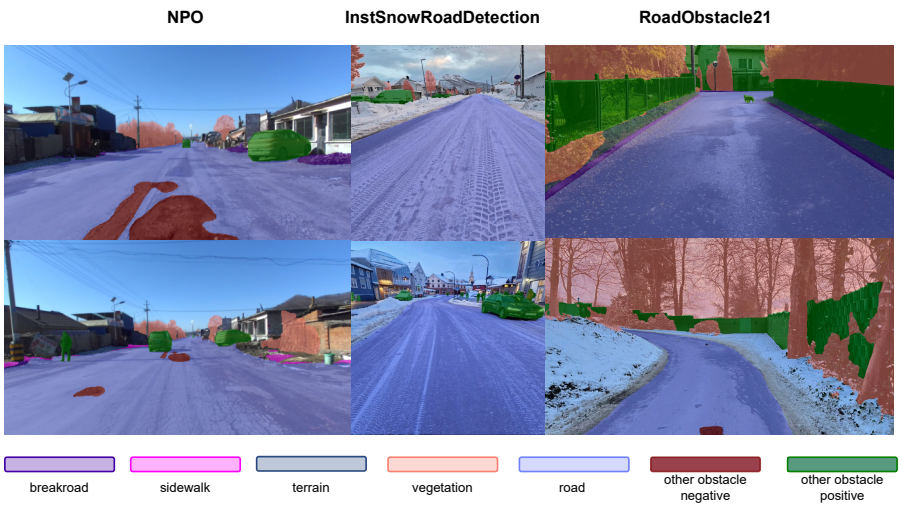


Figure 2: Images with masks from the TAOMR dataset for NPO, InstSnowRoadDetection, and RoadObstacle21 data sources

4. TAOMR method Update

To improve the segmentation of complex-shaped objects such as hoses, wires, ropes, and puddles, we propose an approach that combines **Segment Anything Model (SAM)** with **DINO** and **text-driven refinement**. This method enhances object detection and segmentation accuracy by leveraging both **visual and textual information**.

4.1. Feature Extraction with DINO

- The **DINO model** extracts detailed features from the image, enabling the detection of fine-grained structures and small-scale objects that are often difficult to segment.

4.2. Mask Generation with SAM

- **SAM** is applied to generate precise segmentation masks, ensuring accurate delineation of objects with irregular boundaries.

4.3. Text-Guided Object Identification

- Textual descriptions provide an additional layer of semantic understanding, refining the classification and segmentation process. The integration of text-based object recognition improves the differentiation between visually similar elements.

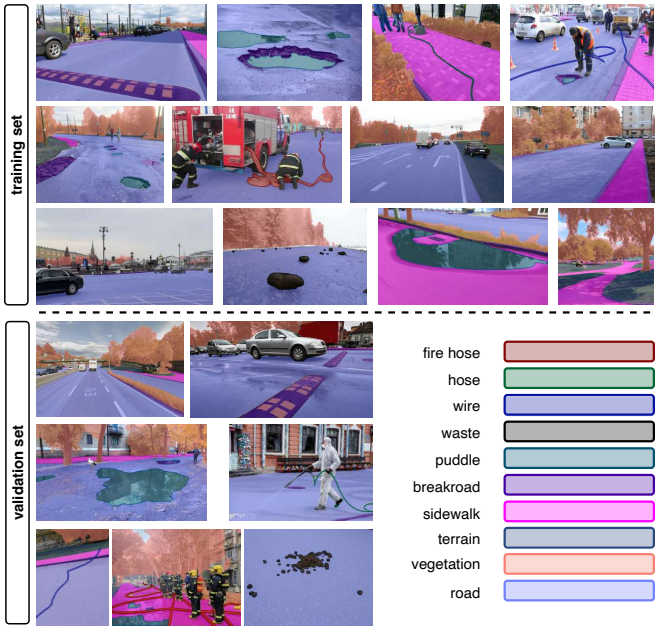


Figure 3: Images with masks from the TAOMR-Bench data

This approach ensures **robust segmentation** by combining **DINO's feature extraction capabilities** with **SAM's high-quality mask generation**, while **textual guidance enhances classification accuracy** for complex objects encountered in robotic navigation environments.

To achieve more accurate open-vocabulary object segmentation, we propose an enhanced version of the HQ-SAM [17] model, referred to as HQ-SAM-text. This approach incorporates an additional text description of the image to improve segmentation performance. Figure 4 shows a diagram of this approach.

The Image Captioning block generates textual descriptions from images, for which we selected the BLIP model [21]. The generated text is then processed by the Text Encoder block, which comprises a BERT encoder [22], a fully connected layer, and convolutional neural networks to reduce the encoded vector to the dimension of image features. The resulting features are summarized with the features of images from the Global-Local Fusion block. We used a combination of the Binary Crossentropy Loss and Dice Loss for training. This method was evaluated on the DIS5K dataset [23] with highly

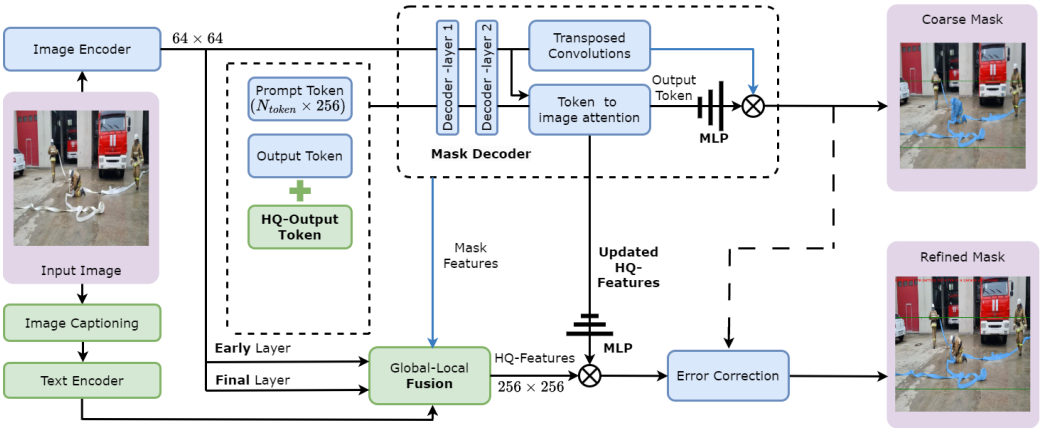


Figure 4: Scheme of the HQ-SAM-Text neural network model designed for segmenting complex-shaped objects. Components that differentiate it from the standard HQ-SAM model are highlighted in green.

detailed masks, and resulted in a 0.6% improvement in mIoU metric (from 77.1% to 77.7%). All models utilized the Vit-B transformer encoder.

As shown in Table 4, HQ-SAM-Text was tested on the TAOMR-Bench dataset and obtained an average 0.5% performance increase. In some categories, the improvement was 1-2%. The mIoU metrics were calculated for the pre-trained and fine-tuned SAM [7] and HQ-SAM [17] models, as well as for the proposed HQ-SAM-Text model. All models utilized the ViT-H transformer encoder. The mask prediction time with HQ-SAM is approximately 0.01 seconds, with HQ-SAM-Text, it is 0.022 seconds.

5. TAOMR benchmark for image segmentation methods (experiments)

5.1. Open-vocabulary segmentation of complex-shaped objects

Figure 5 shows the results of the HQ-SAM-Text, HQ-SAM(fine-tuned), HQ-SAM(pre-trained) models, and ground truth masks. The results demonstrate that the proposed HQ-SAM-Text model, in general, successfully eliminated artifacts of the pre-trained HQ-SAM when the mask fell on nearby objects. Additionally, HQ-SAM-Text outperforms the fine-tuned HQ-SAM model, particularly in segmenting complex-shaped objects, as shown in the first and third rows of Figure 5.

Table 4: Estimation of the object segmentation quality by the IoU metric for the SAM, HQ-SAM, and HQ-SAM-text models on the validation sample of the collected TAOMR-Bench dataset

Model	mIoU	Bump	Manhole	Pit	Puddle	Fire hose	Hose	Wire	Waste
SAM (Pre-trained)	0.792	**0.959**	0.781	**0.926**	0.853	0.838	0.437	0.653	0.89
HQ-SAM (Pre-trained)	0.814	0.951	0.789	0.906	**0.858**	0.847	0.547	0.712	**0.904**
HQ-SAM (Fine-Tuned)	0.828	0.945	0.794	0.904	0.843	0.860	0.654	0.750	0.872
HQ-SAM-Text (ours)	**0.833**	0.943	**0.795**	0.895	0.838	**0.887**	**0.663**	**0.760**	**0.883**

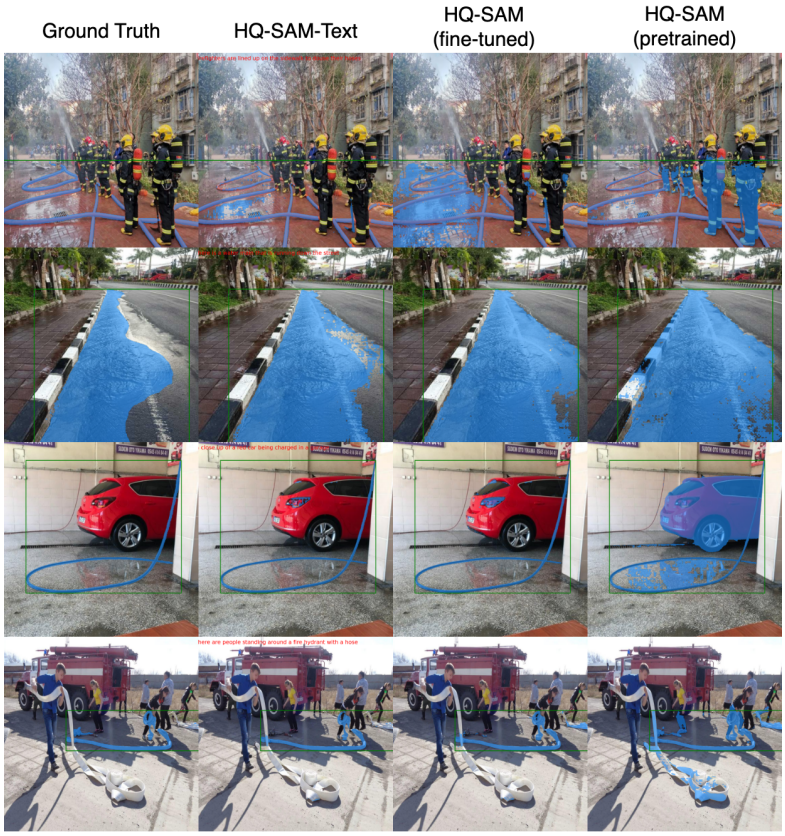


Figure 5: The results of HQ-SAM-Text (second column), fine-tuned HQ-SAM (third column), and pre-trained HQ-SAM (fourth column). Ground truth labeling is in the first column. Segmentation masks are shown in blue.

5.2. Semantic segmentation of road scene

DeepLabV3+ [24], SegFormer [25], and Mask2Former [26] neural networks have each introduced high advancements in semantic segmentation.

DeepLabV3+'s [24] sophisticated atrous convolution enables accurate delineation of roads and pedestrian areas. SegFormer's [25] transformer-based architecture effectively captures long-range dependencies. Mask2Former [26] showcases remarkable performance in identifying different image parts efficiently. OneFormer [10] stands out for its multitasking capability in handling multiple segmentation tasks simultaneously. SAN approach [27] innovates with a decoupled design, enabling accurate recognition of mask proposals by a side network attached to a frozen CLIP model, enabling accurate mask proposal recognition while maintaining a lightweight and adaptable structure through end-to-end training.

We trained listed models on all categories of the TAOMR-Bench dataset: fire hoses, joint hose and wire categories, puddles, road bumps, wastes, sidewalks, terrain, vegetation, and roads. We used 1x Tesla V-100 32GB for OneFormer, while other models were trained with 1x RTX-2080 8GB.

DeepLabV3+ utilized SGD with a learning rate of 0.01, momentum of 0.9, and weight decay of 0.0005, employing polynomial decay over 80,000 steps. Segformer used AdamW with a learning rate of 6e-05 and weight decay of 0.01. It featured a custom parameter-wise setup for the head and disabled weight decay for specific layers. A linear decay for 1500 steps transitioned to polynomial decay, reaching 0 over 160,000 steps. SAN utilized AmpOptimWrapper with AdamW, a learning rate of 0.0001, and a weight decay of 0.0001. It employed gradient clipping and dynamic loss scaling, following polynomial decay for 60,000 steps. For OneFormer, AdamW was used with a base learning rate of 6.0e-05 and weight decay of 0.05. It featured gradient clipping and a WarmupPolyLR scheduler with a polynomial decay pattern. The model achieved its best mIoU at 45,000 steps. Mask2Former employed AdamW with a learning rate of 0.0001, weight decay of 0.05, and gradient clipping. It utilized polynomial decay over 90,000 steps.

For training OneFormer, we used pre-trained weights from the Mapillary Vistas dataset with the Swin Large backbone. The other models were trained using the mmsegmentation framework[7]. DeepLabV3+ utilized $resnet18_v1c$ as its backbone, while SegFormer employed a model pretrained on Cityscapes with the MIT-B0 backbone. Mask2Former utilized the ResNet50 model from Torchvision[8]. Finally, SAN employed a pretrained

[7] `https://github.com/open-mmlab/mmsegmentation`

[8] `https://pytorch.org/vision/`

Table 5: Performance of semantic segmentation models on TAOMR-Bench validation subset. FPS metric is calculated using a Tesla V100 GPU.

Model	Params, M	Image size, px	FPS	mIoU
DeepLabV3+	12.47	640x640	44	53.7
SegFormer	13.68	640x640	27	67.32
Mask2Former	47.4	1024x1024	11	**75.66**
OneFormer	236.32	640x640	10	74.10
SAN	8.37	640x640	18	62.37

Table 6: Per category IoU metric for semantic segmentation models on TAOMR-Bench validation subset.

Model	Unlabel.	Fire hose	Hose	Waste	Puddle	Road bump	Sidewalk	Terrain	Vegetat.	Road
DeepLabV3+	81.51	30.37	33.21	51.47	37.61	33.15	58.8	67.2	85.51	58.17
SegFormer	85.39	50.62	47.99	80.25	38.99	42.49	79.17	78.28	89.43	80.6
Mask2Former	88.32	**79.5**	**58.78**	84.35	50.67	**53.68**	**81.93**	**81.74**	**94.04**	**83.58**
OneFormer	**88.38**	75.15	55.26	**84.55**	**59.62**	52.94	74.87	80.77	93.09	76.37
SAN	85.49	56.3	34.21	72.28	53.75	41.21	61.66	72.21	90.12	56.5

ViT base model with a patch size of 16.

Tables 5 and 6 present the performance metrics for the listed models. The results demonstrate the superiority of the Mask2Former and OneFormer models over the others, providing a mIoU (mean Intersection over Union) metric of more than 74%. However, the data also reveal a considerable number of segmentation errors, suggesting that further improvements could be achieved by expanding the dataset and refining the segmentation methods.

Figure 6 provides a qualitative comparison of semantic image segmentation results across different methods, supporting the quantitative metrics.

5.3. Instance segmentation of obstructive objects

Our benchmarking includes such instance segmentation models as Mask R-CNN [28], YOLOv8-seg [29], YOLO11-seg models [9], OneFormer [10], Mask2Former [26] and OpenSeeD [11]. Mask R-CNN [28] was selected as a traditional and fast CNN model, while YOLOv8 [29] and YOLO11 [9] are the only YOLO (You Only Look Once) family models suitable for instance segmentation, demonstrating strong performance on public datasets. OpenSeeD [11] was chosen for its versatility and adaptability to various datasets and tasks. OneFormer [10] was included due to its unified architecture for both instance and semantic segmentation, offering improved efficiency and generalization. Mask2Former [26] model, with its transformer-based approach, captures long-range dependencies and contextual information, leading to en-

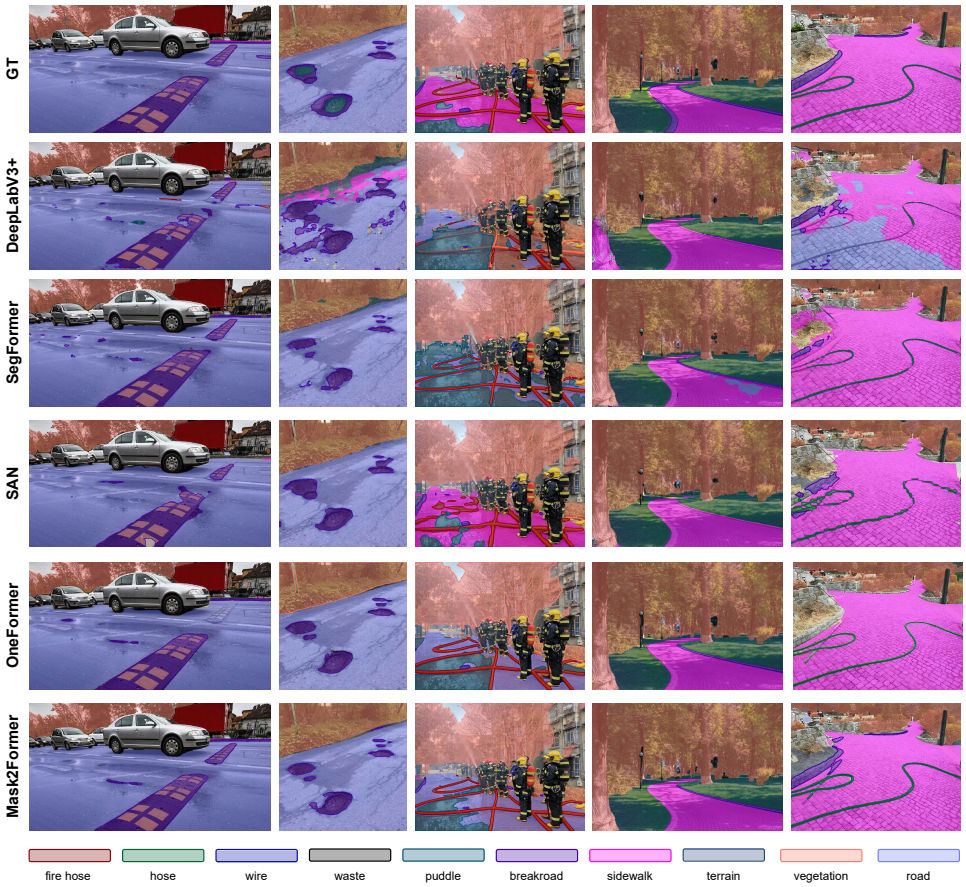


Figure 6: Examples of the different semantic segmentation models work on the TAOMR-Bench dataset images. All models successfully segment background categories, but sometimes confuse the road and sidewalk. OneFormer and Mask2Former better segment objects of complex shapes, such as hoses.

hanced accuracy and robustness in instance segmentation.

We trained these models on the instance segmentation subset of the TAOMR-Bench dataset, which includes categories such as fire hoses, hoses, wires, puddles, road bumps, and animal waste. We used 1x Tesla V-100 32GB for each model.

To train Mask R-CNN we used the SGD optimizer with lr $= 0.005$, momentum $= 0.9$, and weight decay $= 0.0005$; we also used step lr scheduler with step $= 10$ and gamma $= 0.65$. We trained Mask R-CNN for 100 epochs, but the best results were reached around the 40 epoch. We trained YOLOv8-seg

Table 7: Metrics for instance segmentation models on TAOMR-Bench validation subset. FPS metric is calculated using a Tesla V100 GPU. The input image size was 640x640 pixels.

Model	Params, M	FPS	$\mathbf{mAP}_{50}$ (boxes)	$\mathbf{mAP}_{50-95}$ (boxes)	$\mathbf{mAP}_{50}$ (masks)	$\mathbf{mAP}_{50-95}$ (masks)
Mask R-CNN	43.95	15	0.437	0.191	0.217	0.124
YOLOv8n-seg	3.26	**208**	0.392	0.250	0.350	0.170
YOLOv8s-seg	11.78	145	0.451	0.288	0.387	0.205
YOLO11n-seg	2.84	104	0.428	0.272	0.368	0.174
YOLO11s-seg	10.07	106	0.460	0.294	0.390	0.183
OneFormer	218.77	5	0.382	0.253	0.432	0.253
Mask2Former	215.5	1.4	**0.554**	**0.371**	**0.608**	**0.362**
OpenSeeD	197	13	0.377	0.241	0.383	0.222

and YOLO11-seg with the default AdamW optimizer and default parameters for 100 epochs.

For both OneFormer and Mask2Former, we utilized the ADAMW optimizer with a base learning rate of 0.0001 for OneFormer and 1.0e-05 for Mask2Former, both with a weight decay of 0.05. The models were trained with a batch size of 2 and a maximum of 640,000 iterations for OneFormer and 300,000 iterations for Mask2Former. We applied the WarmupPolyLR scheduler with a polynomial decay pattern and enabled gradient clipping with a maximum norm of 0.01. Additionally, AMP training was enabled, and the backbone multiplier was set to 0.1 for both models.

The OpenSeeD model [11], pre-trained on the COCO dataset [19], consists of an image encoder, a text encoder, and a decoder capable of handling foreground, background, and conditional masks. We trained OpenSeeD using the AdamW optimizer with a learning rate of 0.0001, momentum of 0.9, and weight decay of 0.05, along with a step learning rate scheduler with a gamma of 0.1.

During the training of OpenSeeD and OneFormer, we encountered issues related to large image sizes in the dataset. Some images had dimensions exceeding $2,000$ pixels in height or width, leading to memory overflow errors. To address this, we developed a preprocessing procedure to standardize the image sizes. The optimal output image size was determined to be 512×512 pixels. For each image, the compression ratio was calculated based on the larger dimension to preserve the original aspect ratio. The resized image was then centered within a black square of the target dimensions.

Table 7 shows metrics for the listed models. Transformer-based methods such as Mask2Former and OneFormer show better performance in the

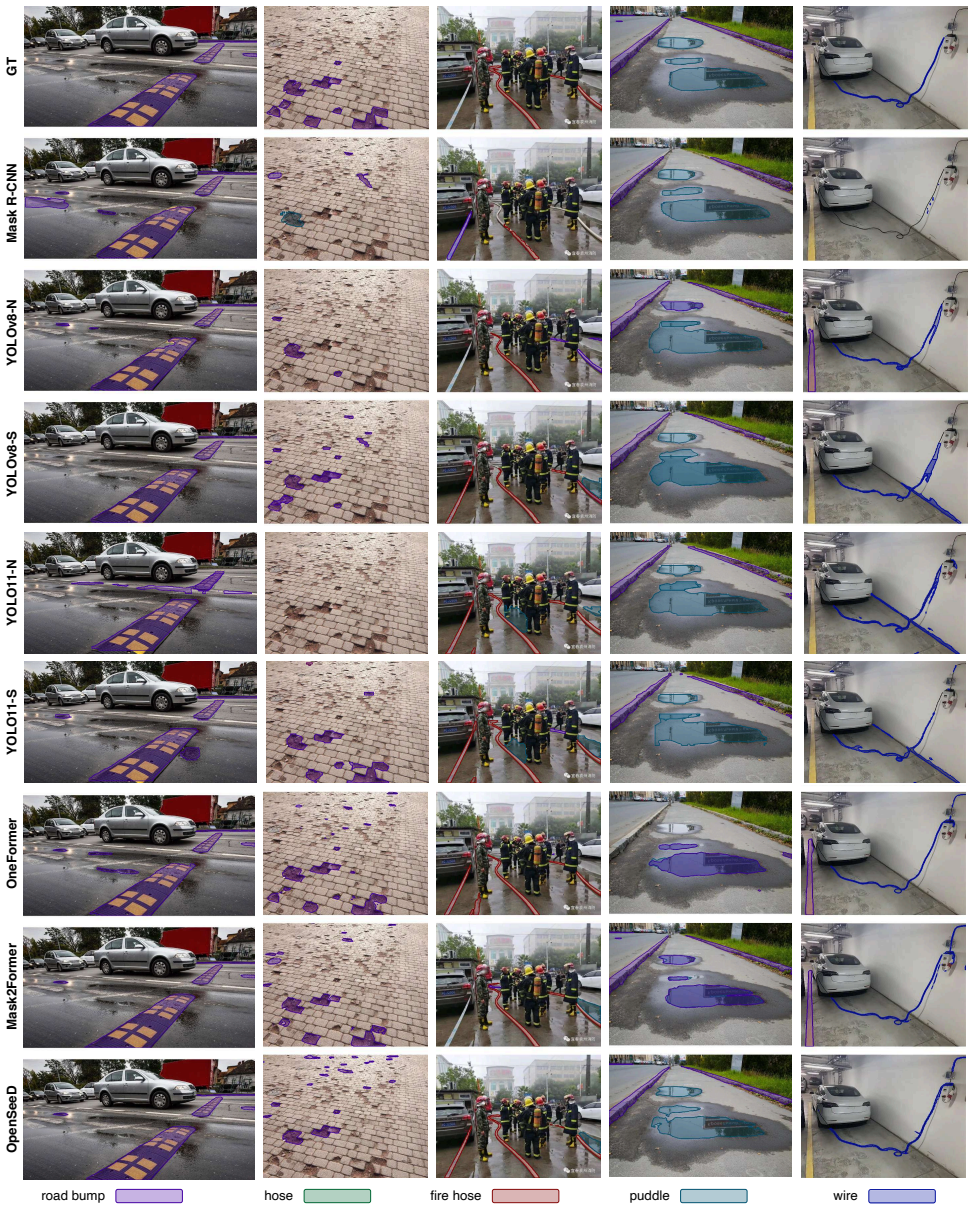


Figure 7: Examples of the different instance segmentation models work on the TAOMR-Open dataset images. All models successfully identify speed bumps and curbs. Transformer models are better at recognizing fire hoses, hoses, and road defects, but a bit worse at detecting puddles.

instance segmentation task. YOLOv8-seg and YOLO11-seg have also demonstrated a high mAP (mean Average Precision) metric for detected bounding boxes. The open-vocabulary OpenSeed method showed some of the worst quality indicators.

The best metric that the Mask2Former method achieves is 0.608 mAP for instance masks with IoU more than 50%. This indicates a large amount of noise in the results of instance segmentation and the significant complexity of the developed dataset for known methods.

In Figure 7, we present the results of the benchmarked models. All models successfully identify speed bumps. OneFormer and Mask2Former better identify road defects, but sometimes these models produce false positives in this category and recognize puddles as road defects. YOLO models are better at identifying hoses and fire hoses than Mask R-CNN, but worse than the transformer models.

6. Conclusion

In this work, we introduced a novel two-stage method for automated dataset collection and labeling called TAOMR. Using this method for different data sources, we created a large-scale dataset, which includes masks for traversable areas, background, and obstructive objects relevant to mobile ground robots.

Based on a representative dataset subsample named TAOMR-Bench, we create a benchmark for an objective assessment of the semantic, instance, and open-vocabulary segmentation quality of specific objects encountered by mobile robots in street and park environments. These objects include wires, hoses of various types, potholes and cracks in the road and sidewalk, natural waste from pets, and speed bumps.

Experiments conducted using existing state-of-the-art neural network models of semantic and instance segmentation on the developed data set highlighted the need for further development and refinement of these models.

To enhance the open-vocabulary segmentation quality for complex-shaped objects, we developed the HQ-SAM-Text model architecture, incorporating image text descriptions.

Limitations and future work. The first stage of the proposed automated data labeling approach effectively collects images from the desired domain but necessitates substantial manual verification and filtering of data obtained from search engines. Integrating trainable algorithms for automatic

data filtering into this process holds promise for enhancing efficiency. Another potential improvement involves using more advanced models instead of OneFormer for close-set segmentation of background objects and their fine-tuning. The HQ-SAM-Text algorithm also offers potential for further development. Specifically, modifying and training it with pre-defined text descriptions of the required segmentation categories, rather than relying on image captioning, could enhance its performance. Expanding the dataset is also promising, both in terms of the number of images and the addition of new categories that the robot may encounter on its way (for example, household waste, building materials, etc.).

Acknowledgment

This work was partially supported by the Analytical Center for the Government of the Russian Federation in accordance with the subsidy agreement (agreement identifier 000000D730321P5Q0002; grant No. 70-2021-00138).

The authors are grateful to the IntegraNT LLC company, which provided data from a real robot, which was partially included in the proposed TAOMR-Open dataset.

References

[1] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems 25 (2012).

[2] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Scene parsing through ade20k dataset, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 633–641.

[3] H. Caesar, J. Uijlings, V. Ferrari, Coco-stuff: Thing and stuff classes in context, in: Computer vision and pattern recognition (CVPR), 2018 IEEE conference on, IEEE, 2018.

[4] C. Li, H. Liu, L. Li, P. Zhang, J. Aneja, J. Yang, P. Jin, H. Hu, Z. Liu, Y. J. Lee, et al., Elevater: A benchmark and toolkit for evaluating language-augmented visual models, Advances in Neural Information Processing Systems 35 (2022) 9287–9301.

[5] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. 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.

[6] G. Neuhold, T. Ollmann, S. Rota Bulo, P. Kontschieder, The mapillary vistas dataset for semantic understanding of street scenes, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 4990–4999.

[7] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al., Segment anything, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 4015–4026.

[8] X. Zou, J. Yang, H. Zhang, F. Li, L. Li, J. Wang, L. Wang, J. Gao, Y. J. Lee, Segment everything everywhere all at once, Advances in Neural Information Processing Systems 36 (2024).

[9] G. Jocher, J. Qiu, Ultralytics yolo11 (2024).
URL `https://github.com/ultralytics/ultralytics`

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

[11] H. Zhang, F. Li, X. Zou, S. Liu, C. Li, J. Gao, J. Yang, L. Zhang, A simple framework for open-vocabulary segmentation and detection, arXiv preprint arXiv:2303.08131 (2023).

[12] Z. Feng, Y. Guo, D. Navarro-Alarcon, Y. Lyu, Y. Sun, Inconseg: Residual-guided fusion with inconsistent multi-modal data for negative and positive road obstacles segmentation, IEEE Robotics and Automation Letters (2023).

[13] INSTSnowRoadDetection Instance Segmentation Dataset, `https://universe.roboflow.com/debigp/instsnowroaddetection/dataset/2`, [Accessed 30-10-2024].

[14] R. Chan, K. Lis, S. Uhlemeyer, H. Blum, S. Honari, R. Siegwart, P. Fua, M. Salzmann, M. Rottmann, Segmentmeifyoucan: A benchmark for anomaly segmentation, arXiv preprint arXiv:2104.14812 (2021).

[15] J. Chen, Z. Yang, L. Zhang, Semantic segment anything, `https://github.com/fudan-zvg/Semantic-Segment-Anything` (2023).

[16] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu, et al., Grounding dino: Marrying dino with grounded pre-training for open-set object detection, arXiv preprint arXiv:2303.05499 (2023).

[17] L. Ke, M. Ye, M. Danelljan, Y. Liu, Y.-W. Tai, C.-K. Tang, F. Yu, Segment anything in high quality, in: NeurIPS, 2023.

[18] OpenCV: Morphological Transformations, `https://docs.opencv.org/4.x/d9/d61/tutorial_py_morphological_ops.html`, [Accessed 30-10-2024].

[19] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft coco: Common objects in context, in: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, Springer, 2014, pp. 740–755.

[20] S. I. Serengil, A. Ozpinar, Lightface: A hybrid deep face recognition framework, in: 2020 Innovations in Intelligent Systems and Applications Conference (ASYU), IEEE, 2020, pp. 23–27. doi:10.1109/ASYU50717.2020.9259802.

[21] J. Li, D. Li, C. Xiong, S. Hoi, Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation, in: International conference on machine learning, PMLR, 2022, pp. 12888–12900.

[22] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[23] X. Qin, H. Dai, X. Hu, D.-P. Fan, L. Shao, L. Van Gool, Highly accurate dichotomous image segmentation, in: European Conference on Computer Vision, Springer, 2022, pp. 38–56.

[24] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 801–818.

[25] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, P. Luo, Segformer: Simple and efficient design for semantic segmentation with transformers, Advances in Neural Information Processing Systems 34 (2021) 12077–12090.

[26] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, R. Girdhar, Masked-attention mask transformer for universal image segmentation, 2022.

[27] M. Xu, Z. Zhang, F. Wei, H. Hu, X. Bai, Side adapter network for open-vocabulary semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 2945–2954.

[28] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961–2969.

[29] G. Jocher, A. Chaurasia, J. Qiu, Ultralytics yolov8 (2023).
URL `https://github.com/ultralytics/ultralytics`