A Universal Railway Obstacle Detection System Based on Optical-Flow Guided Semi-Supervised Segmentation

Qiushi Guo, Bin Cao, Dehao Hao, Cheng Wang, Lijun Chen, Peng Yan
China SWJTU Railway Development Co., Ltd(CSRD), China
{*guoqiushi, caobin, hedehao, wangcheng, chenlijun, yanpeng*}*@csrd.com*

Keywords: Obstacles Detection, Railway Security, Deep Learning.

Abstract: Detecting obstacles in railway scenarios is both crucial and challenging due to the wide range of obstacle categories and varying ambient conditions such as weather and light. Given the impossibility of encompassing all obstacle categories during the training stage, we address this out-of-distribution (OOD) issue with a semi-supervised segmentation approach guided by optical flow clues. We reformulate the task as a binary segmentation problem instead of the traditional object detection approach. To mitigate data shortages, we generate highly realistic synthetic images using Segment Anything (SAM) and YOLO, eliminating the need for manual annotation to produce abundant pixel-level annotations. Additionally, we leverage optical flow as prior knowledge to train the model effectively. Several experiments are conducted, demonstrating the feasibility and effectiveness of our approach.

1 INTRODUCTION

With the rapid advancement of high-speed trains, ensuring the security of railway systems has emerged as a critical public concern. One of the primary challenges is obstacle detection, which plays a crucial role in railway safety. The potential obstacles range from falling rocks to pedestrians, from trucks to animals and eta. Besides, the scenarios is complex due to unpredictable environment conditions. Developing a reliable and robust obstacle detection system can empower train operators and dispatchers to take preemptive actions and mitigate potential accidents.

Deep learning techniques have been widely adopted across various security domains, including mobile payments(Cai et al., 2022), remote sensing(Bischke et al., 2019), disaster detection(Sazara et al., 2019), and fraud detection (Guo et al., 2023). This technology exhibits substantial promise in enhancing railway safety through sophisticated obstacle detection capabilities. Significant efforts have recently been devoted to addressing obstacle detection using deep learning methods(Brucker et al., 2023; Lis et al., 2023). Although these approaches have achieved some success, they also exhibit notable disadvantages:

- Fragility to complex ambient conditions.
- Requirement for extensive manual annotations.

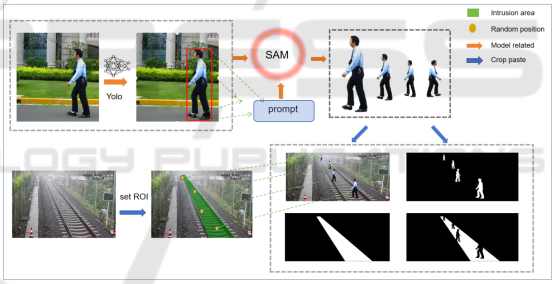

Figure 1: The pipeline for synthetic data generation, utilizing SAM and YOLO to extract target objects from a gallery and superimpose them onto a base image (specifically, a railway image). Notably, this process does not necessitate annotations.

- Difficulty in extending to different scenarios.

Designing an extendable, annotation-free approach with strong generalization ability remains a significant challenge in both industry and academia.

To address the aforementioned issues, we propose a semi-supervised approach guided by optical flow. To mitigate the data shortage problem, we employ SAM (Kirillov et al., 2023) and YOLO (Redmon et al., 2016) to generate highly realistic pseudo-images for training. Instead of manually collecting and annotating images pixel by pixel, we prepare two image sets: base images (fewer than 100 background images with only railway areas annotated) and object

Guo, Q., Cao, B., Hao, D., Wang, C., Chen, L. and Yan, P.
A Universal Railway Obstacle Detection System Based on Optical-Flow Guided Semi-Supervised Segmentation.
DOI: 10.5220/0013057600003941

In *Proceedings of the 11th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2025)*, pages 259-265
ISBN: 978-989-758-745-0; ISSN: 2184-495X

images. The object images include categories such as pedestrians, animals, vehicles,and textures. The bounding boxes of above objects can be detected by YOLO as the prompts for SAM. The pixel-level annotations(masks) can be obtained automatically. These objects are then pasted onto the base images according to the masks. The entire process are illustrated as **Fig.** 1 This process simultaneously generates image and mask pairs without manual effort.

To address the challenges posed by varying weather conditions, we implement two complementary strategies. Firstly, we compile a dataset of base images captured under diverse weather conditions, our experiment sites have the ability to simulate different weather environments,including rainy, foggy, and clear (sunny). Secondly, we utilize optical-flow model RAFT(Teed and Deng, 2020) to provide positional information as prior knowledge. To obtain above predictions, we generate pseudo sequences of obstacles. This involves creating an initial pseudo frame at point $P_i(x,y)$ and subsequently generating a new frame at $P_{i+1}(x+\delta, y+\delta)$ with the same object superimposed.

To validate the effectiveness and robustness of our proposed approach, we conduct experiments on different possible obstacles under different scenarios. The categories of obstacles include pedestrians, rocks, cubes and parcels. Except pedestrians, all categories of obstacles are unseen in training stage. Experimental results indicate that our approach yields satisfactory performance across different weather scenarios.

2 RELATED WORK

2.1 Railway Obstacles Detection

Matthias Brucker *et al.*(Brucker et al., 2023)propose a shallow network to learn railway segmentation from normal railway images. They explore the controlled inclusion of global information by learning to hallucinate obstacle-free images. Zhang Qiang *et al.*(Zhang et al., 2023). combine segmentation model with the LiDAR in their obstacle detection system; Amine Boussik *et al.*(Boussik et al., 2021) propose an unsupervised models based on a large set of generated convolutional auto-encoder models to detect obstacles on railway's track level. To best of our knowledge, there has been no work implementing optical-flow in railway obstacles scenarios.

2.2 Segmentation with Optical Flow

Optical flow is used to detect continuous motion between sequential frames, serving as an important cue for identifying objects in scenarios where the background remains stable and motionless. Laura *et al.* (Sevilla-Lara et al., 2016). demonstrate the effectiveness of jointly optimizing optical flow and video segmentation using an iterative scheme; Volodymyr *et al*(Fedynyak et al., 2024). present an architecture for Video Object Segmentation that combines memory-based matching with motion-guided propagation resulting in stable long-term modeling and strong temporal consistency.

3 METHOD

3.1 Data Acquisition

The pipeline of our approach is illustrated in **Fig.**2. Given a set of base images B and target images T, our objective is to identify potential obstacles within specific regions η. Unlike traditional detection methods that categorically detect each obstacle, we reformulate the problem as a binary segmentation task. Instead of attempting to detect all potential obstacles, which is impractical, our emphasis is on segmenting the railway area, a region that remains consistent over time compared to obstacles. To simulate these scenarios effectively, we generate highly realistic pseudo-images using a copy-paste approach. Additionally, to address challenges posed by extreme weather conditions, which can obscure object segmentation, we introduce optical flow to provide prior information guiding the segmentation model. Pseudo images I_t and $I_{t+\delta}$ are generated by applying a small shift δ to the target object, simulating its movement. The output of the optical flow model is incorporated along with pseudo images as input to facilitate accurate predictions. This section will delve into the detailed methodology employed throughout this process.

3.2 Experimental Site

All experiments were conducted at our experimental site in Xinjin, Chengdu. The site measures approximately 70 meters in length and 8 meters in width. It includes rail lines, sleepers, and road debris to simulate realistic railway conditions. The facility is equipped with rain and fog simulation devices capable of replicating four different levels of rainfall and fog intensity. As illustrated in the figure below, the

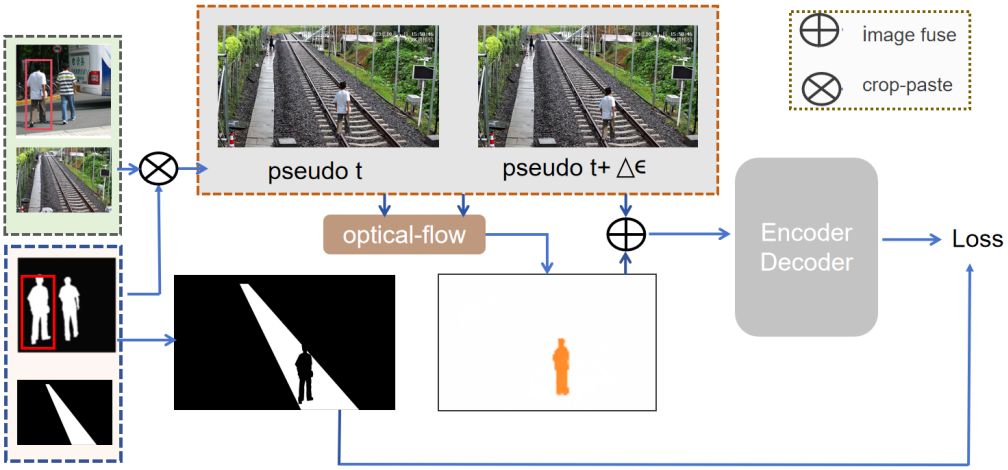

Figure 2: The pipeline of our proposed approach involves generating synthetic images and masks using background images, object images, and their corresponding masks. An image at time step $t + \Delta\epsilon$ is generated based on the image at time step t with a slight displacement of the pasted object. The predictions from the optical flow model, along with the generated pseudo-images, are then fed into an encoder-decoder framework for training.

Figure 3: Experimental site in Chengdu.

spray system covers the entire railway area. The simulation process is controlled by a dedicated control system developed by CSRD.

Base Images are used in our experiments are gathered at our facility in Chengdu, which features a railway spanning over 60 meters and includes simulators for fog and rain conditions. To ensure diversity in our dataset, we capture images under different weather scenarios, specifically rainy, foggy, and sunny conditions **Fig.** 6. Due to the fixed position of the camera, only one mask is required for annotation purposes. Importantly, the railway areas in the base images are devoid of any potential obstacles. Any obstacles present are generated using a copy-paste method.

Figure 4: Sample base images depicting various weather conditions. From left to right, the images illustrate scenes captured under foggy, normal, and rainy conditions.

Object Image dataset comprises three categories: PennFudanPed, Obj365 (part) (Shao et al., 2019), and DTD (Cimpoi et al., 2014). To facilitate fully automated application of our methodology, we proceed under the assumption that no masks are initially available. We focus on selecting categories likely to occur in our scenario, such as animals (e.g., deer, horse, cow) and vehicles (e.g., truck, cart). This ensures our approach is tailored to handle relevant objects effectively.

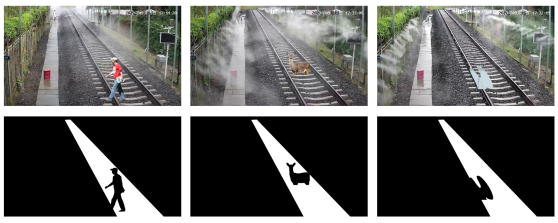

Figure 5: Sample of generated image-mask pairs. From left to right: pedestrian, animal and texture.

The entire process can be delineated into sequential steps: Initially, object images are fed into the YOLO model, which returns a list of bounding boxes identifying detected targets. These bounding boxes serve as inputs for SAM, which generates segmenta-

tion masks to outline the object pixels. Subsequently, these segmented object pixels are integrated into the base images based on the segmentation mask guidance. Here, we elaborate on the detailed methodology.

- **Object Detection with YOLO:** Object images are inputted into the YOLO model, specifically trained on Obj365, to detect objects belonging to predefined target categories fitting our scenario.
- **Segmentation with SAM:** Bounding boxes from YOLO are used as prompts for SAM to generate segmentation masks. These masks delineate object pixels, facilitating their extraction from the object images.
- **Integration with Base Images:** Extracted object pixels are seamlessly integrated into the corresponding regions of base images, aligning with the guidance provided by the segmentation masks.

During the SAM stage, while not every segmentation mask achieves perfection, each contributes to the overall objective of accurately segmenting the railway area rather than focusing on obstacles. To address challenges related to out-of-distribution (OOD) scenarios, we introduce random polygon generation with texture rendering from DTD. Additionally, object resizing and rescaling are applied to enrich image content and bolster model robustness.

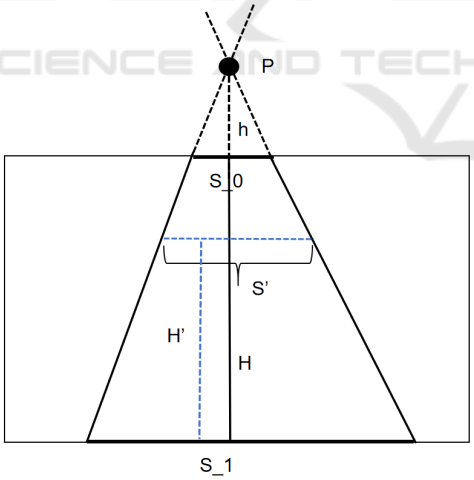

Figure 6: Re-scale illustration. We assume that two rails intersect at point P. S_1 and S_0 are width of two rails in image. H is the height of the image. h is the distance between P and top line of two rails in image space.

The re-scale follow the equation below:

$$\frac{S_0}{S_1} = \frac{h}{h+H} \tag{1}$$

$$\frac{S^{'}}{S_1} = \frac{H-h^{'}+h}{h+H} \tag{2}$$

$$h = \frac{s_0 \cdot H}{S_1 - S_0} \tag{3}$$

$$S^{'} = \frac{S_1 \cdot (h+H-H^{'})}{h+H} \tag{4}$$

We assume that two rails will interact at point P. S_1 and S_0 are distances in 2D images. H is the height of the image and h is the distance between P and top of image. The estimated width of pasted objects in height $H^{'}$ can be calculated as equations above. demonstrated as **Fig .**5

3.3 Optical-Flow

Optical flow is based on the assumption that the intensity of a point in an image remains constant as it moves from one frame to the next.

$$I(x,y,t) = I(x+\Delta t, y+\Delta t, t+\Delta t) \tag{5}$$

In our scenario, we employ RAFT (Recurrent All-Pairs Field Transforms) as our chosen model, which demonstrates robust performance across a wide range of scales from tiny to large. The size of obstacles in our dataset varies, spanning from hundreds of pixels down to less than 50 pixels in size. Utilizing the RAFT model requires two consecutive frames for optical flow estimation. Accordingly, we generate two pseudo images I_t and I_{t+1}, where the same target objects are pasted with a slight positional shift η.

$$Motion = \phi(I_t, I_{t+1}) \tag{6}$$

$$I_{t+1} = I_t(obj_x + \Delta x, obj_y + \Delta y) \tag{7}$$

We set Δx and Δy range between 5-10. The motion prediction will be leveraged as prior information fused with pseudo image to train the model.

4 EXPERIMENTS

4.1 Dataset and Evaluation Metrics

Dataset Our training dataset is consisted of three parts: *obs_person*, *obs_animals* and *obs_textures*, namely person obstacles, animal obstacles and obstacles generated from texture polygons. The details are described as follow: As for the test dataset, we recollect images with various obstacles under different weather conditions in different distance to the camera.

Metrics *mIoU* is used to evaluate the performance of our model. *mIoU* refers to the Mean Intersection over

union, which is a widely used metric in segmentation task. It can be calculated as follow:

$$IoU_i = \frac{TP_i}{TP_i + FP_i + FN_i} \tag{8}$$

$$mIoU = \frac{1}{n}\sum_{i=0}^{n} IoU_i \tag{9}$$

pixel accuracy is also a metric to evaluate the segmentation models.

$$Pixel\_accuracy = \frac{N\_corr}{N\_total} \tag{10}$$

where N_corr is the number of correctly classified pixels, N_total is the number of total pixels.

Table 1: Datasets description.

Name	Volume	Dis(m)	Category
obs_person	4000	0-70	pedestrian
obs_animal	4000	0-70	cow,horse,deer
obs_texture	2000	0-70	see DTD
val_near	200	0-20	pedestrian,rock,cube
val_middle	200	20-50	pedestrian,rock,cube
val_far	200	50-70	pedestrian,rock,cube

4.2 Implementation Details

Our method is implemented using the PyTorch framework and the model is trained on an RTX 3090Ti, with 24 GB memory. CPU processor is Intel i7-12700F with 20 cores. We select Dice loss(Sudre et al., 2017) as the loss function and Adam(Kingma and Ba, 2014) as the optimizer. Starting learning rate is set to 0.001. The batch size is set to 16 and the number of epochs to 25. Albumentation(Buslaev et al., 2020) is utilized to perform data augmentation. Data transformations include horizontal flip, coarse dropout, and random brightness contrast adjustments.

4.3 Results

Compare with Models

To validate the performance of our approach, we conduct experiments on our three self-collected datasets: val_near, val_mid, and val_far. The details are described in **Table** 1. The basic training dataset contains 10,000 images (4,000+4,000+2,000). To fully assess the impact of the number of generated images, we increase the dataset size by 10% ,30% and 50% in rows 4 and 5.

The results are illustrated in **Table**3, which show that both RAFT and segmentation-based approaches can effectively segment obstacles in our railway area experiments. Combining RAFT and pseudo-images enhances model performance. As more generated images are added to the training dataset, the model's performance gradually reaches its limit.

Across Obstacles Categories

To validate the robustness of our proposed approach, we conducted experiments across various categories. The tested classes include cubes, branches, pedestrians, and parcels. We also performed experiments under different distance conditions. The results, presented in **Table**2, indicate that although the mean Intersection over Union (mIoU) decreases as distance increases, the results remain reliable (over 0.72) at a distance of 70 meters in our scenario. Additionally, we observed that branches are particularly challenging targets compared to other objects due to their complex shapes and textures. Another possible reason for this difficulty is the lack of similar objects in the training dataset. In contrast, the accuracy for pedestrians is relatively high, likely because our synthetic images include highly realistic pedestrians from the FudanPenn dataset.

Ablation Study

We conduct ablation experiment to validate the effect of different target objects. The results are demonstrated as **Table** 4 Comparing the row 1,2,3 with row 4, we can find that each obs dataset contributes to improving the robustness and accuracy of the model.

Qualitative Result

Figure 7 presents the segmentation results of our approach with and without optical-flow guidance. It is evident that optical flow enhances the model's performance. The segmentation results are more cohesive (columns 5 and 6) and exhibit greater sensitivity to small objects (columns 1 and 2). For pedestrians in a normal posture, both sets of results are satisfactory. An interesting observation is that the use of optical flow tends to produce false predictions outside of railway areas (columns 2 and 5). However, this does not affect the final outcomes of our analysis,since we only foucs on the obstacles in railway areas.

Table 2: Experimental results across various categories of obstacles at different distances. The metric is IoU.

	rocks	**pedestrians**	**parcel**	**cube(20cm)**	**cube(40cm)**	**branches**	**Average**
5m-10m	0.952	0.964	0.927	0.931	0.937	0.871	0.930
10m-30m	0.931	0.942	0.913	0.904	0.911	0.784	0.898
30m-50m	0.873	0.907	0.871	0.887	0.873	0.685	0.849
50m-70m	0.732	0.784	0.751	0.771	0.737	0.583	0.726
Average	0.872	0.899	0.866	0.873	0.865	0.731	0.848

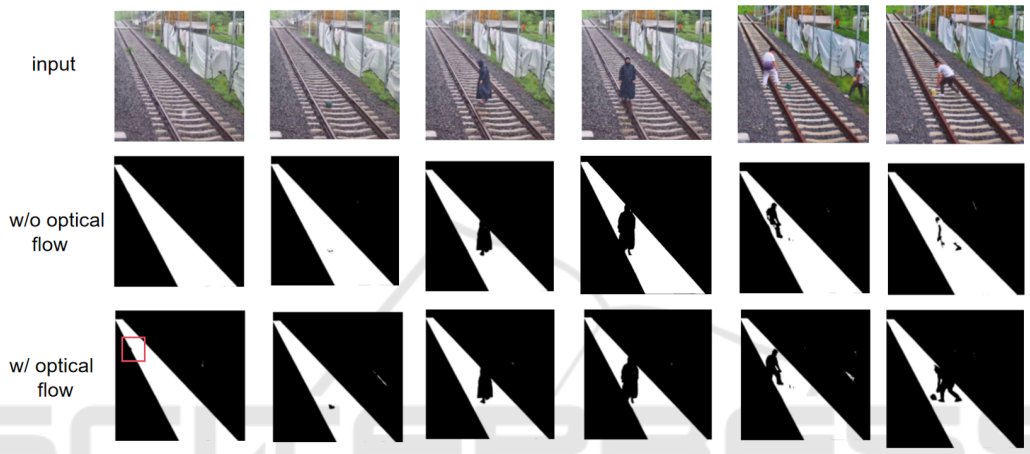

Figure 7: Qualitative results w/wo optical-flow module. The predicted masks with optical-flow provide more robust and accurate especially in boundaries and tiny objects segmentation.

Table 3: The experimental results across different test sets demonstrate that both the Optical flow(OF) and an increased dataset contribute to improve model performance at varying distances. Although the accuracy decreases as the distance increases, the segmentation predictions remain effective in detecting obstacles even in the test set val_far.

	val_near	**val_mid**	**val_far**
Unet	0.804	0.793	0.767
PSPNet	0.817	0.809	0.735
PAN	0.813	0.826	0.749
DeepLabv3	0.825	0.817	0.747
OF	0.735	0.674	0.627
DeepLabv3+OF	0.843	0.828	0.749
DeepLabv3+OF+10%	0.837	0.843	0.734
DeepLabv3+OF+30%	0.851	0.842	0.751
DeepLabv3+OF+50%	**0.863**	**0.851**	**0.782**

Table 4: Ablation study.

	obs_person	obs_animal	obs_texture	mIoU
1	✓	✓	✗	0.781
2	✓	✗	✓	0.817
3	✗	✓	✓	0.732
4	✓	✓	✓	0.849

5 CONCLUSION

This paper introduces a universal segmentation model based on a semi-supervised approach. To address out-of-distribution (OOD) challenges, we generate highly realistic pseudo images instead of relying on manual pixel-level annotations. Additionally, we enhance performance by incorporating optical flow techniques. Experimental results demonstrate satisfactory performance across various potential objects.

REFERENCES

Bischke, B., Helber, P., Folz, J., Borth, D., and Dengel, A. (2019). Multi-task learning for segmentation of building footprints with deep neural networks. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1480–1484. IEEE.

Boussik, A., Ben-Messaoud, W., Niar, S., and Taleb-Ahmed, A. (2021). Railway obstacle detection using unsupervised learning: An exploratory study. In *2021 IEEE Intelligent Vehicles Symposium (IV)*, pages 660–667. IEEE.

Brucker, M., Cramariuc, A., Von Einem, C., Siegwart, R., and Cadena, C. (2023). Local and global information in obstacle detection on railway tracks. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9049–9056. IEEE.

Buslaev, A., Iglovikov, V. I., Khvedchenya, E., Parinov, A., Druzhinin, M., and Kalinin, A. A. (2020). Albumentations: fast and flexible image augmentations. *Information*, 11(2):125.

Cai, H., Lin, J., Lin, Y., Liu, Z., Tang, H., Wang, H., Zhu, L., and Han, S. (2022). Enable deep learning on mobile devices: Methods, systems, and applications. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 27(3):1–50.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. (2014). Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613.

Fedynyak, V., Romanus, Y., Hlovatskyi, B., Sydor, B., Dobosevych, O., Babin, I., and Riazantsev, R. (2024). Devos: Flow-guided deformable transformer for video object segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 240–249.

Guo, Q., Chen, Y., and Liao, S. (2023). Enhancing mobile privacy and security: A face skin patch-based anti-spoofing approach. In *2023 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, pages 52–57. IEEE.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. (2023). Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026.

Lis, K., Honari, S., Fua, P., and Salzmann, M. (2023). Perspective aware road obstacle detection. *IEEE Robotics and Automation Letters*, 8(4):2150–2157.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.

Sazara, C., Cetin, M., and Iftekharuddin, K. M. (2019). Detecting floodwater on roadways from image data with handcrafted features and deep transfer learning. In *2019 IEEE intelligent transportation systems conference (ITSC)*, pages 804–809. IEEE.

Sevilla-Lara, L., Sun, D., Jampani, V., and Black, M. J. (2016). Optical flow with semantic segmentation and localized layers. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3889–3898.

Shao, S., Li, Z., Zhang, T., Peng, C., Yu, G., Zhang, X., Li, J., and Sun, J. (2019). Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8430–8439.

Sudre, C. H., Li, W., Vercauteren, T., Ourselin, S., and Jorge Cardoso, M. (2017). Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*, pages 240–248. Springer.

Teed, Z. and Deng, J. (2020). Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer.

Zhang, Q., Yan, F., Song, W., Wang, R., and Li, G. (2023). Automatic obstacle detection method for the train based on deep learning. *Sustainability*, 15(2):1184.