Automatic Detection and Localization of Roadwork

Srinivas Narasimhan[1]

Robert Tamburo[2]

FINAL RESEARCH REPORT

Contract #69A3551747111 The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. This report is funded, partially or entirely, by a grant from the U.S. Department of Transportation's University Transportation Centers Program. The U.S. Government assumes no liability for the contents or use thereof.

[1] https://orcid.org/0000-0003-0389-1921

[2] https://orcid.org/0000-0002-5636-9443

Table of Contents

1. Project Investigators and Contributors

This is a summary report combining multiple sub-projects conducted at the University Transportation Center at Carnegie Mellon University in relation to the overall project titled “Automatic Detection and Understanding of Roadworks”. The participants who contributed to the different sub-projects are listed below. Student contributions are noted in the separate sub-projects.

- Srinivasa Narasimhan (CMU RI Professor, ORCID: 0000-0003-0389-1921)
- Robert Tamburo (CMU RI Senior Project Scientist, ORCID: 0000-0002-5636-9443)
- Christoph Mertz (CMU RI Principal Project Scientist, ORCID: 0000-0001-7540-5211)
- Dinesh Reddy (CMU RI Ph.D. Student, ORCID: 0009-0005-5945-4212)
- Khiem Vuong (CMU RI M.S. Student, ORCID: 0009-0006-2474-2270)
- Anurag Ghosh (CMU RI M.S. Student, ORCID: 0000-0002-1617-0851)
- Shefali Srivastava (CMU RI M.S. Student)
- Neha Boloor (CMU RI M.S. Student)
- Tiffany Ma (CMU RI M.S. Student)
- Michael Cardei (CMU RISS Student, ORCID: 0009-0006-7574-7979)
- Nicholas Dunn (CMU RISS, ORCID: 0000-0002-0625-7638)
- Hailiang Zhu (CMU RISS, ORCID: 0009-0000-5955-4161)

CMU = Carnegie Mellon University, RI = Robotics Institute, RISS = Robotics Institute Summer Scholars

2. Introduction

Roadwork zones present a serious impediment to vehicular mobility. Whether new construction or maintenance is taking place, work in road environments causes lower vehicle speeds, congestion, increased risk of rear-end collisions, and more difficult maneuvering. For example, around 24% of non-recurring congestion occurs in roadwork zones, which equates to 482 million vehicle hours of delays[3]. To combat the disruption of roadwork zones, Departments of Transportation[4] are employing predictive analysis tools to deploy more efficient roadwork configurations. Further efficiencies can be gained for individual drivers with crowd-sourced navigation systems like Waze, but those data must be manually entered causing a visual, motor, and cognitive distraction for the driver. Google maps now automatically shows roadwork, but those data are often slow to update and do not distinguish between active/inactive work zones or specify lane restrictions/changes. Additionally, driver assistive and autonomous driving technology do not reliably function or navigate a vehicle through roadwork especially when lane markings are absent. For example, a driverless autonomous vehicle recently navigated itself into a construction site then drove itself into an open trench[5].

In this work, we focus on using visual data from affordable cameras to address some of these problems of roadwork identification by developing computer vision and machine learning methods for automatic detection and localization of roadwork zones. We envision that the calculated information can be shared with other drivers and enable dynamic route planning for navigation systems, driver assist systems, and self-driving cars for efficiently and safely maneuvering through or around road work zones. Moreover, a comprehensive view of road work activity in a region can be constructed from information shared and distributed by users. Such a view would prove to be a useful tool for dynamic detour route adjustment to optimize traffic flow.

Figure 1: Example roadwork zones exemplifying their heterogenous appearance.

[3] FHWA, "Making Work Zones Work Better". https://ops.fhwa.dot.gov/aboutus/one_pagers/wz.htm
[4] Kirkpatrick, Rich."PREDICTIVE WORK ZONE ANALYSIS TOOL OFFICIALLY DEPLOYED JULY 1". PennDOT Bureau of Innovations, July 28, 2021. https://www.penndot.pa.gov/PennDOTWay/pages/Article.aspx?post=451
[5] Zigoris, Julie. "Driverless Waymo Car Almost Digs Itself Into Hole—Literally". San Francisco Standard, January 15, 2023. https://sfstandard.com/2023/01/15/driverless-waymo-car-digs-itself-into-hole-literally/

3. Methods

Identifying roadwork from visual data from cameras, as well as data from other sensors such as LIDAR, is an extremely challenging problem because road work zones are dynamic and heterogeneous in appearance (Figure 1). No two roadwork sites (construction or maintenance) look alike. Because of this, driver-assist and self-driving systems have difficulty with navigation within these zones. For example, longitudinal road markings not changed during lane shifts may cause lane keep assist systems to steer towards barriers or other objects. In this example, a system that automatically recognizes a roadwork zone on approach could provide a warning to the driver and disable lane keep assist. To our knowledge, there is little work being done in this area. Our approach was to detect objects commonly located within roadwork sites and based on their proximity to each other, location relative to surfaces (e.g., roads, sidewalks, bike lanes, etc.), other heuristics, determine whether roadwork is present. Achieving this objective required development in the following areas.

3.1. Roadwork Dataset Creation (Dinesh Reddy, Khiem Vuong, Anurag Ghosh, Tiffany Ma, Shefali Srivastava, Neha Boloor)

There are not any known datasets that provide labeled roadwork zones or labeled objects commonly found within roadwork zones. Therefore, a novel, comprehensive dataset was created. To create the dataset roadwork images from 19 different U.S. cities were manually labeled (assigned a class) and segmented (delineated with a polygon), collectively referred to as annotated in this report. Descriptive tags were assigned to each image to capture additional information about the roadwork zone and the environment. Additionally, a general description describes the roadwork objects and roadwork zone.

Images were obtained from cameras mounted on vehicles driving around in 19 U.S. cities. We collected images in the Greater Pittsburgh Area with an Apple iPhone 14 Pro Max mounted on the inside windshield of a standard passenger vehicle. Areas of roadwork were found by following data provided by the PA government, e.g., 511PA[6] and random searches. Once a roadwork zone was found, images were captured by means of a wireless (Bluetooth) remote camera trigger. Images were filtered to reduce the number of similar viewpoints of the same roadwork zone. Images not containing roadwork were also captured for testing purposes. In total, there are 3,171 Pittsburgh images that we captured in the Roadwork Dataset.

Table 1: Number of images for each data source and city included in the roadwork dataset resulting in a total of 8,556 images.

Data Source: City	# Images	Data Source - City	# Images
CMU: Pittsburgh	3,171	Roadbotics: Indianapolis	95
Roadbotics: Philadelphia	161	Roadbotics: San Antonio	381
Roadbotics: Washington DC	266	Roadbotics: Boston	861
Roadbotics: Pittsburgh	597	Roadbotics: Phoenix	42
Roadbotics: Houston	63	Roadbotics: Minneapolis	116
Roadbotics: Charlotte	223	Roadbotics: San Francisco	236
Roadbotics: Detroit	522	Roadbotics: Seattle	293
Roadbotics: New York City	124	Roadbotics: Denver	465
Roadbotics: Jacksonville	38	Roadbotics: Los Angeles	674
Roadbotics: Chicago	118	Roadbotics Columbus	110

[6] https://www.511pa.com/

To add more instances of roadwork objects and roadwork zones in Pittsburgh and other U.S. cities, images were used from the RoadBotics Open Data Set[7]. Using images in other cities also captures the variability of objects, work vehicles, etc. present in other cities. The RoadBotics dataset includes videos captured by windshield mounted devices, GPS data, and accelerometer data. Frames were extracted from the video files by assuming that most roadwork zones have traffic cones. A simple traffic cone detector was used on the video files to save the image frames. Often the vehicle was stopped resulting in a sequence of images capturing the same exact scene. In such cases, the image sequence was manually filtered by deleting repetitive images. In some cases, images were kept if they captured unique obstructions of roadwork objects or roadwork zones. 8,556 images were saved for the dataset. The number of images for each data source and city are summarized in Table 1. While images were annotated, models were trained and evaluated to determine which objects were rarer and required more annotations. Further efforts to filter images focused on more rare objects hence the lower number of images that were saved for some cities.

The objects common to roadwork zones found in the United States were identified and named according to the Federal Highway Administration's Manual on Uniform Traffic Control Devices (MUTCD)[8]. Additionally, the main surfaces found in a road environment were also of interest for localization potential. Objects were segmented either completely manually or with the guidance of semantic segmentation (Figure 2). Any objects that were occluded were marked as such within the native CVAT framework. The results of semantic segmentation still required manual intervention for assigning labels to relevant objects and, at times, correcting polygon points to follow object boundaries better. Non-work people and non-work vehicles were also sparsely included since they were reliably delineated with semantic segmentation. The Computer Vision Annotation Tool (CVAT)[9], which is a free and open-source web-based tool was used for annotating the images. CVAT was installed on a server in a secure lab where members of the team and external labeling companies (SRN Tech Solutions and Label Your Data) segmented the images. Most labels were assigned to segmented objects by a single person at CMU. A list of the objects that were annotated are in Table 2.

Figure 2: Left shows an example image of a work zone. Right shows an example with objects of interest that have been manually segmented by polygons. The polygons are overlayed with a unique color. Each segmented object is outlined with a solid line if it is not occluded and a dashed line if it is occluded.

[7] https://www.roadbotics.com/2021/03/15/roadbotics-open-data-set/
[8] https://mutcd.fhwa.dot.gov/pdfs/2003/Ch6F.pdf
[9] https://www.cvat.ai/

Table 2: List of objects that were manually annotated. Light gray objects are not typically associated with roadwork and those shown in dark gray are the main surfaces found in a road environment.

Object	Object	Object
Worker	Barricade	Non-Work Person
Work Vehicle	Vertical Panel	Non-Work Vehicle
Work Equipment	Tubular Marker	Road
TTC* Sign	Cone	Sidewalk
Guide Sign	Drum	Bike Lane
Lane Ends Sign	Fence	Off Road
TTC Message Board	Other Roadwork Objects	Roadside
Arrow Board	Police Officer	Work Zone
Barrier	Police Vehicle	

* TTC: Temporary Traffic Control

The descriptive info included with each image was manually assigned. Almost all of the info was assigned by the same person at CMU. The categories are below followed by each of the options.

- Environment: Urban, Suburban, Highway, Rural, Unknown, Other
- Time: Dark, Light, Twilight, Unknown, Other
- Work Zone?: Yes, No, Unsure
- Travel Alteration: Lane Shift, Partially Blocked, Fully Blocked, None, Other
- Weather: Snow, Sunny, Wet, Cloudy, Partly Cloudy, Fog or Mist, Ice, Other, Unknown

A General Description was also manually written to describe the roadwork objects and zone. The dataset is available at https://cs.cmu.edu/~ILIM/roadwork_dataset

3.2. Dataset Augmentation (Nicholas Dunn, Anurag Ghosh)

It is costly to capture and annotate the many thousands of images needed to train models for each of the objects of interest. To augment these real-world images, a method was developed to paste manually segmented objects into other images allowing us to create roadwork zones in images that do not contain roadwork zones. Roadwork object models can be trained on many more images to improve detection results.

Achieving instance segmentation with performance capabilities reliable enough for real-world applications, machine learning models are typically pre-trained on a large, annotated dataset of common objects and then fine-tuned on a dataset representative of the context in which the model will be deployed and containing the categories relevant for detection. However, building large-scale, annotated datasets is a very costly and time-consuming process. To mitigate this issue, methods of creating new annotated images by augmenting existing datasets have been explored. For example, Copy-Paste methods place manually segmented objects into another image randomly[10] or based on surrounding context[11], and placing objects in different random locations within the same image[12].

[10] G. Ghiasi, Y. Cui, A. Srinivas, R. Qian, T.-Y. Lin, E. D. Cubuk, Q. V. Le, and B. Zoph, “Simple copy-paste is a strong data augmentation method for instance segmentation,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 2918–2928.

[11] N. Dvornik, J. Mairal, and C. Schmid, “Modeling visual context is key to augmenting object detection datasets,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 364–380.

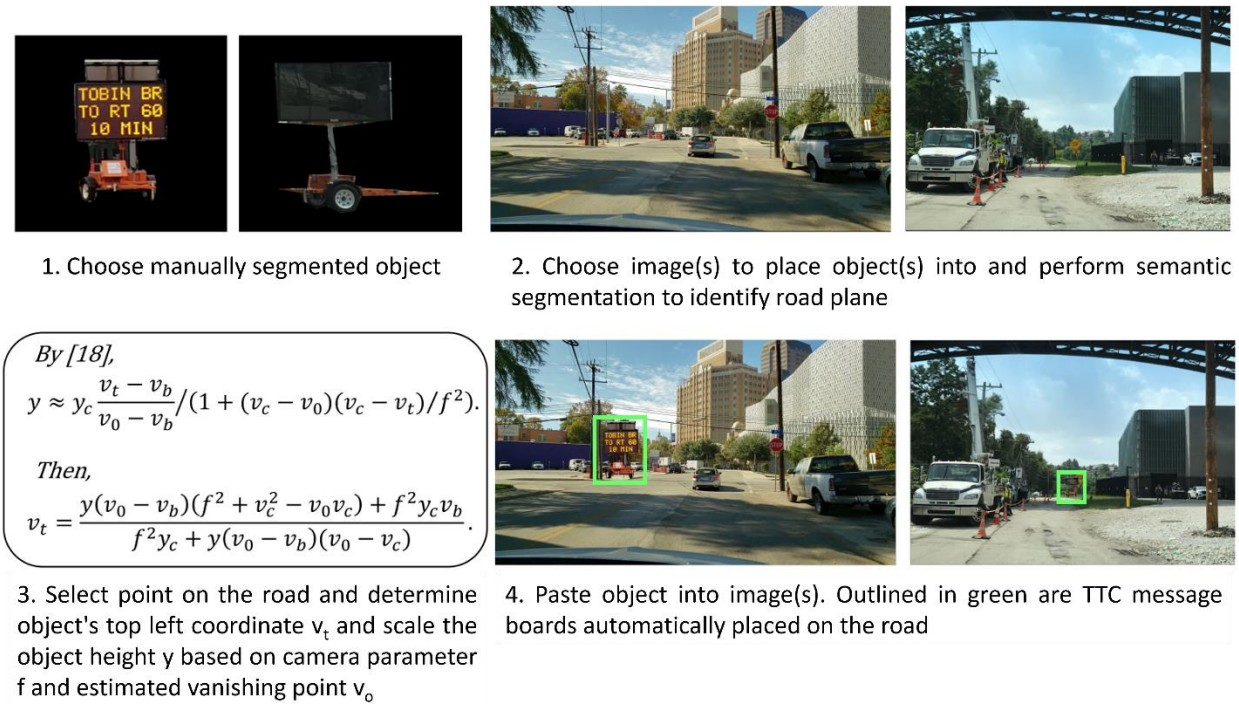

Figure 3: Overview of the GeometryPaste method. Objects are copied from their original images and pasted into new images using geometry and context.

A limitation of these methods is the lack of automatic object scaling. We developed a Copy-Paste method called GeometryPaste to incorporate geometric information to ensure pasted objects are of the correct scale relative to their depth from the camera. The result is realistic data augmentation to increase instance segmentation performance on rare object categories. GeometryPaste copies objects from their original images and paste them onto new background images using both the geometry and context of the objects and background images. First, we randomly choose an object and a background image. Then we choose a random point on the existing road segmentation in the background image to paste the object onto. Next, the object is scaled to the appropriate size based on the location of the chosen point. The object is then pasted onto the new image, allowing for partial truncation. Finally, existing object annotations are updated to account for occlusions from the pasted object. The entire system overview is shown in Figure 3. Each step of the method is described below, and results can be found in Section 4.2.

Object Selection: Although multiple objects can be used with our method, we focus on objects that are underrepresented and difficult for our model to detect based on AP scores. Specifically, we choose only the TTC Message Board from our dataset to paste into new background images. Because TTC Message Boards have many sizes, configurations, and messages, we select 27 TTC Message Boards with high-quality annotations from our training dataset to ensure diversity in the generated images. Despite being the highest quality, some quality issues exist with the selected TTC Message Boards, including small occlusions from other objects and missing parts due to annotation errors.

---

[12] H.-S. Fang, J. Sun, R. Wang, M. Gou, Y.-L. Li, and C. Lu, "Instaboost: Boosting instance segmentation via probability map guided copypasting," in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 682–691.

Background Selection: Background images are chosen from our training dataset and contain existing objects. Because most images contain at most one TTC Message Board, we select background images without TTC Message Boards. Again, due to annotation errors, many images do not contain road segmentations. Since the objects will be pasted onto points selected from the road segmentation, we ensure the chosen images contain a road segmentation. Another constraint is that the image must contain its predicted vanishing point. This constraint is imposed to assist with visual analysis of the quality of the vanishing point prediction, which is used in our scaling function. In total, there are 1,640 candidate background images.

Pasting, Truncation, and Occlusions: To maintain appropriate context, objects are pasted onto randomly chosen points from the road segmentation of background images. Truncation occurs when the object is pasted such that the image boundary partially occludes it. Our policy for truncation is the same as in Dwibedi, et al[13]. That is, we ensure at least 25% of the object's bounding box remains in the image. If, after scaling the object, the chosen location results in a truncation of more than 75%, a new location is selected. In addition to truncation, occlusions of existing objects may occur after pasting the object into the new image. Existing object annotations are updated accordingly to account for occlusions for objects that remain at least 25% visible and removed otherwise.

Object Scaling: Having the camera parameters for our dataset and ensuring the vanishing line is within the image allows us to scale the object to the appropriate size. Let y be the object height, f the camera focal length, v_c the camera optical center y-coordinate, y_c the camera height, v_0 the y-coordinate of the horizon line, and v_t and $v_b$the object top and bottom coordinates, respectively.

$$y \approx y_c \frac{v_t - v_b}{v_0 - v_b} / \left(1 + (v_c - v_0)(v_c - v_t)/f^2\right).$$

Therefore,

$$v_t = \frac{y(v_0 - v_b)(f^2 + v_c^2 - v_0 v_c) + f^2 y_c v_b}{f^2 y_c + y(v_0 - v_b)(v_0 - v_c)}.$$

Letting v_b be the y-coordinate of the new location, the new height becomes $v_t - v_b$, and the width is scaled accordingly to maintain the aspect ratio. The y-coordinate of the horizon line v_0 is predicted using NeurVPS[14], a deep neural network vanishing point detector. The object's height y in its original image is obtained using its ground-truth annotation, known camera parameters, and predicted vanishing point with Equation (5) from[15].

Blending: We apply either no blending or Gaussian blurring to pasted objects. When Gaussian blurring is applied, each image is generated twice: once with no blending and once with Gaussian blurring. The background image remains the same, and the object maintains the same position and scale, with the only difference being the blending strategy.

---

[13] D. Dwibedi, I. Misra, and M. Hebert, "Cut, paste and learn: Surprisingly easy synthesis for instance detection," in Proceedings of the IEEE international conference on computer vision, 2017, pp. 1301–1310.

[14] Y. Zhou, H. Qi, J. Huang, and Y. Ma, "Neurvps: Neural vanishing point scanning via conic convolution," Advances in Neural Information Processing Systems, vol. 32, 2019.

[15] D. Hoiem, A. A. Efros, and M. Hebert, "Putting objects in perspective," International Journal of Computer Vision, vol. 80, pp. 3–15, 2008.

3.3. Road Marking Detection (Hailiang Zhu, Anurag Ghosh)

Markings on the pavement such as center markings, edge markings, and lane markings are crucial for providing guidance to drivers. Often in roadwork zones, these pavements markings are painted over and repainted, which can be confusing to drivers. A system that automatically detects road markings in roadwork zones can warn drivers or even provide more accurate information to driver assist systems. Unfortunately, detecting roadwork markings in roadwork zones is very challenging because there are not any known datasets. In this work, we developed a method to transfer labels for roadwork markings from a publicly available dataset to images in our roadwork dataset.

The roadwork dataset has 26 classes, but there are many more classes that could be relevant for detecting roadwork zones. For example, detecting road markings and comparing them to past data can be indicative of roadwork activity. Manually annotating road markings would be extremely time consuming. However, other datasets may have road markings annotated. This work focuses on a framework for training a segmentation model that incorporates annotations from two separate datasets (Figure 4) Motivated by the need for a unified and versatile instance segmentation model, we explored how to train such a model effectively using datasets with diverse label spaces. Leveraging the power of transformer-based architectures, particularly the Mask2Former model, we built an instance segmentation model capable of detecting a broader array of objects than individual datasets alone.

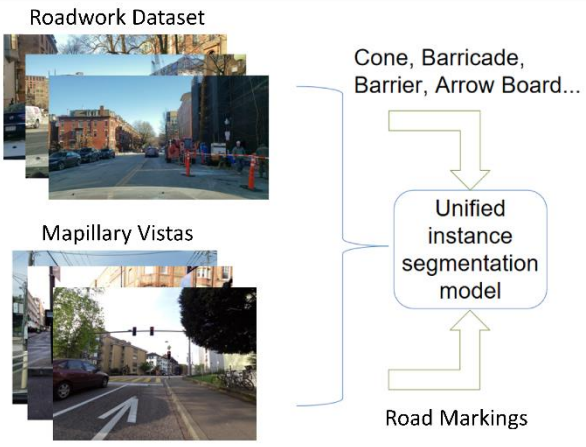

Figure 4: Training a segmentation model that incorporates annotations from two datasets.

The primary research question revolves around understanding how the Mask2Former model can generalize and adapt to different label spaces, thereby enabling the detection of additional categories in an efficient and accurate manner. To accomplish this, we adopt a two-step approach: firstly, we train the Mask2Former model using the Mapillary Vistas dataset[16], and secondly, we fine-tune the model on an augmented dataset, obtained by pseudo-labeling the Roadbotics dataset with instances of the "lane marking-general" category from Mapillary Vistas. The whole training process is done with mmdetection[17] and Detectron2[18], two open source libraries that provides all kinds of Mask2Former segmentation models. By examining the model's performance on the test set, including Precision-Recall curves and Qualitative results, we aim to assess the effectiveness of our proposed method.

[16] Connor Shorten and Taghi M Khoshgoftaar. "A survey on image data augmentation for deep learning". In: Journal of big data 6.1 (2019), pp. 1–48.
[17] Kai Chen et al. "MMDetection: Open MMLab Detection Toolbox and Benchmark". In: arXiv preprint arXiv:1906.07155 (2019).
[18] Yuxin Wu et al. Detectron2. https://github.com/facebookresearch/detectron2. 2019.

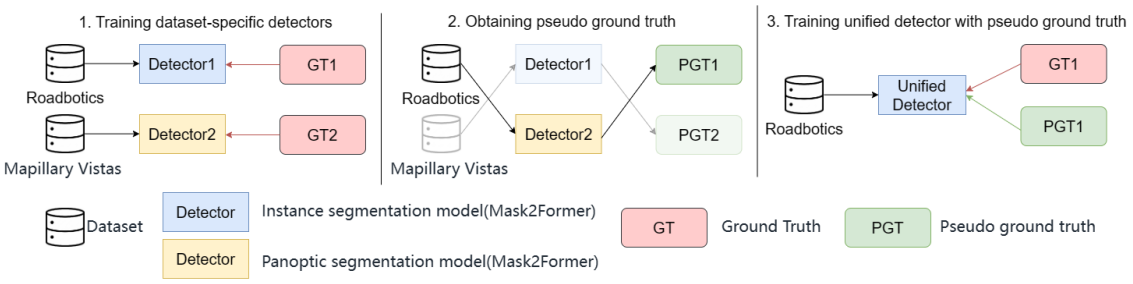

Figure 5: Three step pipeline for training a unified detector.

A three-step pipeline (Figure 5) was developed for training a unified object detector from two separate datasets. First, two separate detectors are trained on each of the datasets. Second, a pseudo ground truth is obtained. Finally, the unified detector is trained with the pseudo ground truth.

Datasets: Our study utilizes two critical datasets, namely the publicly available Mapillary Vistas dataset and the in-house collected Roadbotics dataset. The Mapillary Vistas dataset comprises a diverse collection of street scene images. On the other hand, the Roadbotics dataset focuses on construction zone-related objects.

1) Mapillary Vistas Dataset: Mapillary Vistas dataset is a large-scale street-level image dataset containing 25,000 high resolution images annotated into 66/124 object categories of which 37/70 classes are instance-specific labels (v.1.2 and v2.0, respectively). In the Mapillary Vistas dataset, the “lane marking-general” category is labeled as ‘stuff’, representing the continuous and amorphous regions of lane markings. Since lane markings are not treated as individual instances but rather as part of the background in the instance segmentation setting, we utilized a panoptic segmentation model to inference and segment the semantic component of the “lane marking-general” category.

2) Roadbotics Dataset: The Roadbotics dataset focuses on objects closely related to construction zone areas. The whole dataset has 5071 images and is then split into 3 sets, where the training set has 3584 images and 44537 annotations; the validation set has 513 images and has 6331 annotations; the test set has 974 images and 16932 annotations. By utilizing the approach of pseudo labeling, we then add 31475 annotations of lane markings to the training set, 4477 annotations of lane markings to the validation set and 9371 annotations of lane markings to the test set.

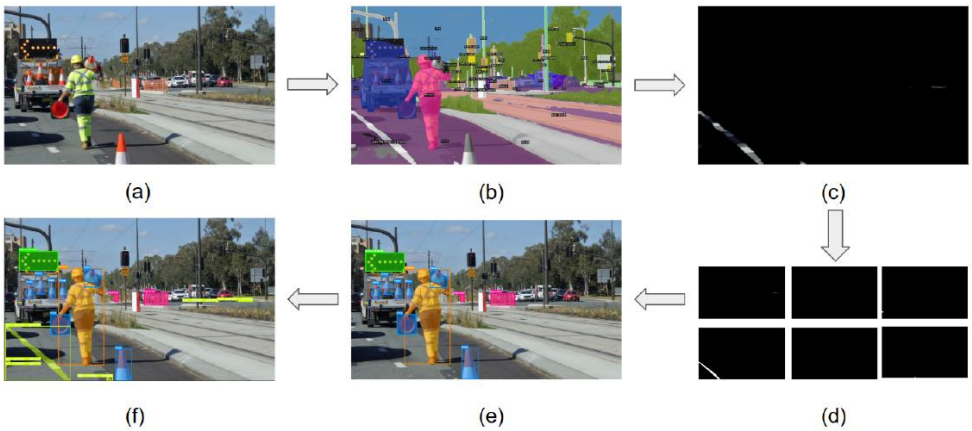

Figure 6: Example of pseudo-labeling an image: (a) Original Image; (b) Inference with pretrained panoptic segmentation model; (c) Extracting the mask; (d) Segmenting masks based on connectivity; (e) Original annotation of the image; (f) Image with pseudo labels added.

Pretrain the Panoptic Segmentation Model: We select the panoptic segmentation model of Mask2Former pretrained on the Mapillary Vistas dataset from the model of Detectron2. The pre-trained Mask2Former model with 200 queries utilizes the Swin-L backbone trained on the IN21k dataset for 300,000 iterations. It achieves competitive performance with a PQ (Panoptic Quality) score of 45.5 and an mIoU (mean Intersection over Union) of 60.8 on the Mapillary Vistas dataset.

Pseudo Labeling: Given that the original label space of the Roadbotics dataset does not include the lane marking, the pretrained model analyzes each image and generates pixel-level predictions in the form of a two-dimensional matrix, representing the semantic information for various object categories present in the scene. This process corresponds to the subfigure (a)-(b) in Figure 6. Once we have the inference results, we need to extract the mask corresponding to the lane markings from the generated matrix. We define specific rules and criteria to isolate the lane markings based on their semantic class labels. These rules help us segment and extract the specific pixels that represent lane markings from the inference matrix. This process corresponds to the subfigure (c)-(d) in Figure 6.

After successfully extracting the mask for lane markings, the next step is to convert these masks into a COCO compliant format. We follow the COCO annotation format guidelines and organize the mask information into suitable JSON files. Each instance of lane marking is represented as a separate mask, and its corresponding category information is associated with a unique identifier. This allows us to create an augmented dataset in the COCO format, which includes the pseudo-labeled instances of lane markings, along with their corresponding semantic information. This process corresponds to the subfigure (e)-(f) in Figure 6.

Model Training: In this phase, we conducted fine-tuning of the instance segmentation models, utilizing the pseudo-labeled dataset generated earlier. The Mask2Former architecture was employed, and the training was carried out in two different frameworks, namely mmdetection and detectron2. To expedite parameter updates and promote faster convergence, a strategic decision was made to employ the ResNet-50 backbone for the Mask2Former model in both mmdetection and detectron2 frameworks. This choice was carefully made, considering computational efficiency while ensuring the retention of strong feature representation capabilities.

To harness transferable knowledge and facilitate faster convergence, both models underwent pretraining on the original Roadbotics dataset before fine-tuning on the augmented pseudo-labeled Roadbotics dataset. This approach allowed the models to efficiently adapt to the new dataset and effectively recognize lane markings. Results can be found in Section 4.3.

3.4. Roadwork Detection and Localization (Michael Cardei)

Current methods for road work zones mainly rely either on object detection or fine-tuning large pre-trained image classification models. Methods that categorize work zones only on object detection neglect to interpret the dynamics, structure, and functional significance of the work zone while also using a limited list of items missing crucial elements that have great significance. On the other hand, current methods that employ transfer learning on large pre-trained image classification models do not make any progress toward localization and fail to refute spurious correlations or unintentional memorization.

In this work, we present a novel approach to road work zone detection and localization that leverages two distinct deep learning models in tandem. In this study, we define detection as the process of

Figure 7: Road view images from the roadwork dataset and their respective instance segmentation predicted mask in a variety of work zone scenarios. The mask is converted to binary before overlay resulting in uniform representation irrespective of class.

identifying the presence or absence of a work zone, while localization refers to the process of identifying the spatial location of the work zones. Our research begins with the introduction of an annotated roadwork dataset, which encompasses road-work-specific categories such as arrow boards, work vehicles, workers, and barricades, among others, as demonstrated in Figure 7. For the detection task we employ transfer learning with EfficientNet[19] pre-trained on ImageNet[20] to accurately classify a wide range of work zone scenarios. If a work zone is detected, we then utilize our second model, a Mask R-CNN[21] instance segmentation model fine-tuned for this task. After applying the convex hull algorithm to

[19] Tan, M. and Le, Q.V. (2019) EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning, ICML 2019, Long Beach, 9-15 June 2019, 6105-6114.

[20] J. Deng, W. Dong, R. Socher, L. -J. Li, Kai Li and Li Fei-Fei, ”ImageNet: A large-scale hierarchical image database,” 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 2009, pp. 248- 255, doi: 10.1109/CVPR.2009.5206848.

[21] K. He, G. Gkioxari, P. Doll´ar and R. Girshick, ”Mask RCNN,” 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 2980-2988, doi: 10.1109/ICCV.2017.322.

the output, a “work zone” mask is generated, providing detailed localization. This dual-model approach is designed to mitigate false positives in scenarios where the instance segmentation model identifies objects associated with work zones in the absence of one. This cooperation of models ensures robustness and precision in our detection and localization tasks. The methodology is illustrated in Figure 8 and explained in detail below.

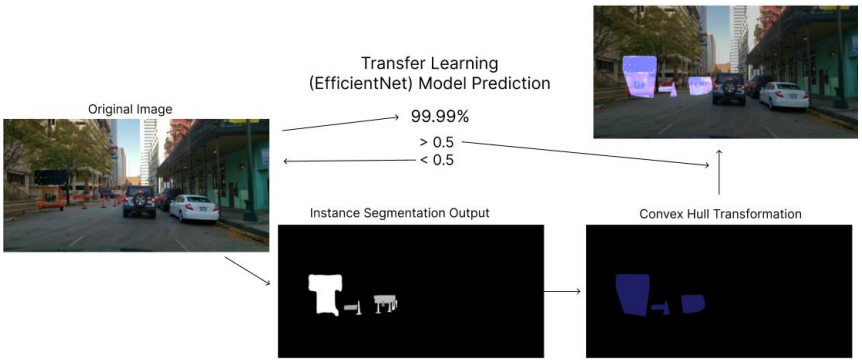

Figure 8: Overview of method on a sample image. Each image passes through two models simultaneously, EfficientNet-B3 for work zone detection and Mask R-CNN for localization. The instance segmentation output is converted to binary and then the Convex Hull algorithm is applied. If the EfficientNet model detects a work zone the mask is then overlayed.

Detection: Transfer learning is a powerful technique that enables training deep learning models for specific or complex tasks that may not have sufficient data available by leveraging feature representations learned from larger, more encompassing datasets[22]. In this study, we used transfer learning to finetune the EfficientNet-B3 model for the purpose of work zone detection, a binary classification task. The EfficientNet models, which are pre-trained on the ImageNet dataset, provide great performance, exceptional parameter efficiency, and speed in comparison to similar-performing model architectures.

We modified the EfficientNet-B3’s architecture by replacing its final output layer with a linear layer customized for our binary classification task and utilized a CUDA-enabled GPU. For training, we used cross-entropy loss as the objective function, and the Adam algorithm for optimization, with a learning rate set at 0.001. These configurations enable our model to accurately recognize and discern the presence of a wide variety of work zone situations that are represented in the dataset.

We evaluate detection performance using the Area under the Receiver Operating Characteristic (ROC) curve (AUC), supplemented by confusion matrices. The AUC provides a comprehensive measure of a model’s binary classification ability, assessing its skill in distinguishing positive from negative instances across all classification thresholds. We apply both same-distribution and different-distribution evaluation setups. The former splits images from all cities into training and testing sets, while the latter uses images from 11 cities for training and a distinct set of 8 cities for testing. This approach allows us to assess the model’s generalization capacity on unseen cities.

[22] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, ”A Comprehensive Survey on Transfer Learning,” in Proceedings of the IEEE, vol. 109, no. 1, pp. 43-76, 2021, doi: 10.1109/JPROC.2020.3004555.

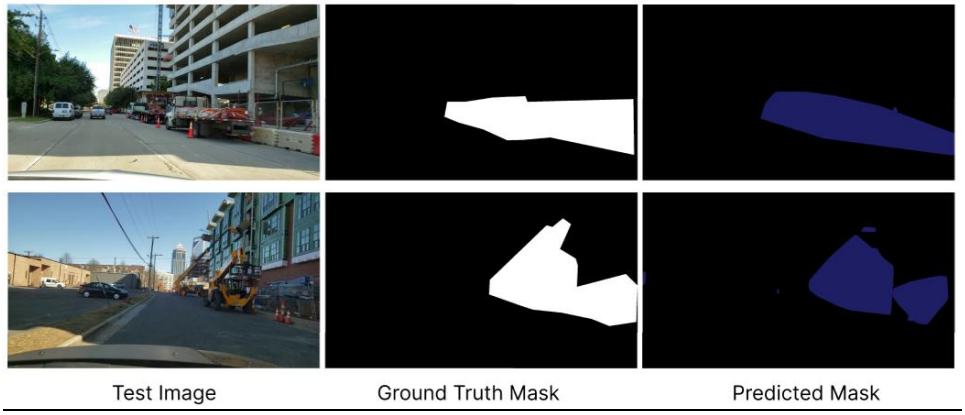

Figure 9: Image mask prediction and ground truth for work zone evaluation.

Localization: Similar to the task of detection, we utilize transfer learning for work zone localization. We employed a Mask R-CNN model with a ResNet50 backbone pre-trained on the COCO dataset[23]. The choice of Mask R-CNN was motivated by its excellent performance in instance segmentation tasks. The Mask R-CNN model is fine-tuned on the manually annotated RoadBotics dataset using the MMDetection[24] opensource computer vision toolbox. After generating the instance segmentation mask, a convex hull algorithm is applied to generate a more encapsulating mask that encompasses all objects in a broader area.

The Convex Hull algorithm is a mathematical tool commonly used in image processing for tasks such as object recognition and noise reduction. In our approach, we apply the Convex Hull algorithm to the output masks generated by the instance segmentation model. A Convex Hull can be understood as the minimum convex set encompassing a given point set S[25]. In the context of images, provided a binary image, the convex hull is the set of active pixels that form the smallest convex polygon for each disconnected region. An intuitive description oftentimes used is given nails in a board, a stretched rubber band around the nails is the convex hull. Before the Convex Hull operation, we perform a dilation operation to connect proximal objects and reduce small holes. After this we perform labeling, the process of clustering groups and assigning a unique pixel value (label) per group. We then independently use the Convex Hull algorithm for each group and overlay the resulting matrix onto the original image. We apply two separate evaluation methodologies for localization: the first concerns the performance of the instance segmentation model on work zone-related objects, while the second focuses on evaluating our methods' ability to accurately identify work zone areas as demonstrated in Figure 9. Results for our detection and localization method can be found in Section 4.4.

[23] T.-Y. Lin et al., "Microsoft Coco: Common Objects in Context," Computer Vision – ECCV 2014, pp. 740–755, 2014. doi:10.1007/978-3-319-10602-148.
[24] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, Z. Zhang, D. Cheng, C. Zhu, T. Cheng, Q. Zhao, B. Li, X. Lu, R. Zhu, Y. Wu, J. Dai, J. Wang, J. Shi, W. Ouyang, C. C. Loy, and D. Lin, "MMDetection: Open MMLab Detection Toolbox and Benchmark," arXiv preprint arXiv:1906.07155, 2019.
[25] R. V. Chadnov and A. V. Skvortsov, "Convex hull algorithms review," Proceedings. The 8th Russian-Korean International Symposium on Science and Technology, 2004. KORUS 2004., Tomsk, Russia, 2004, pp. 112-115 vol. 2, doi: 10.1109/KORUS.2004.1555560.

4. Findings

4.1. Roadwork Dataset Creation (Dinesh Reddy, Khiem Vuong, Anurag Ghosh, Tiffany Ma, Shefali Srivastava, Neha Boloor)

All 8,556 images in the dataset were manually labeled and segmented. The total number of annotated roadwork objects are shown in Figure 10. Shown in Figure 11 is a breakdown of annotated roadwork objects for each city. The dataset is available on the website:
https://www.cs.cmu.edu/~ILIM/roadwork_dataset/

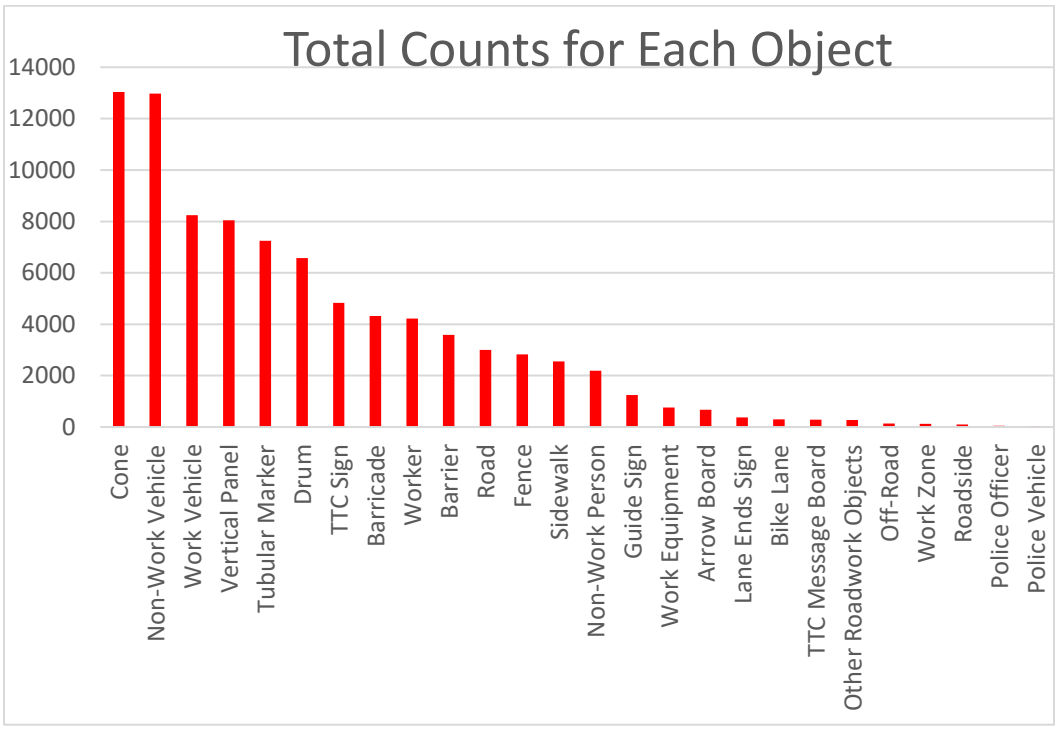

Figure 10: Number of instances for each type of object in the Roadwork Dataset.

	Boston	Charlotte	Chicago	Columbus	Washington DC	Denver	Detroit	Houston	Indianapolis	Jacksonville	Los Angeles	Minneapolis	New York City	Philadelphia	Phoenix	San Antonio	San Francisco	Seattle	Pittsburgh
Worker	276	204	104	75	144	160	113	31	52	32	441	36	129	93	51	363	228	93	1592
Work Vehicle	829	343	172	123	229	339	568	65	127	57	867	101	206	175	55	395	437	175	2975
Work Equipment	99	3	8	13	65	26	39	3	35	6	26	19	119	16	7	6	23	16	234
TTC Sign	416	108	16	120	85	424	150	18	44	19	388	113	29	74	37	430	59	74	2232
Guide Sign	45	26	12	86	69	82	62	51	16	10	133	92	33	18	54	85	30	18	328
Lane Ends Sign	0	5	5	8	26	0	11	10	3	3	225	22	8	13	3	3	12	13	9
TTC Message Board	18	8	13	6	25	9	34	7	7	0	26	2	1	3	0	11	8	3	106
Arrow Board	12	11	23	13	47	98	69	10	15	3	88	0	5	5	6	6	32	5	230
Barrier	557	95	21	64	213	148	314	41	39	10	237	113	115	111	46	76	155	111	1116
Barricade	149	116	169	104	22	184	231	46	17	28	768	205	59	82	72	218	121	82	1645
Vertical Panel	31	0	0	9	10	1006	1	6	1	0	14	1	8	0	32	217	3	0	6703
Tubular Marker	3586	125	13	58	16	486	300	11	11	3	128	64	18	9	3	488	76	9	1840
Cone	1564	142	64	43	208	1669	236	20	62	33	580	44	97	129	14	1192	184	129	6624
Drum	594	432	193	425	533	183	1042	299	193	15	23	345	114	242	0	913	2	242	790
Fence	268	112	24	74	112	222	381	19	69	10	203	69	59	74	37	169	78	74	775
Other Roadwork Objects	71	0	0	0	0	17	0	0	0	0	0	0	0	0	0	4	0	0	182
Police Officer	17	0	0	0	0	1	0	0	0	0	0	0	0	0	0	20	0	0	16
Police Vehicle	7	0	0	0	0	3	0	0	0	0	0	0	0	0	0	23	1	0	8
Non-Work Person	403	0	0	0	0	703	0	0	0	0	0	0	0	0	0	189	0	0	900
Non-Work Vehicle	1829	0	0	0	0	3128	0	0	0	0	0	0	0	0	0	2013	0	0	6010
Road	276	0	0	0	0	497	0	0	0	0	0	0	0	0	0	381	0	0	1849
Sidewalk	270	0	0	0	0	610	0	0	0	0	0	0	0	0	0	413	0	0	1260
Bike Lane	16	0	0	0	0	40	0	0	0	0	0	0	0	0	0	2	0	0	237
Off-Road	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	132
Roadside	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
Work Zone	0	73	0	0	0	0	0	57	0	0	0	0	0	0	0	0	0	0	0

Figure 11: Total number of objects that were annotated for each city in the Roadwork Dataset.

4.2. Dataset Augmentation (Nicholas Dunn, Anurag Ghosh)

Dataset: We fine-tuned our pre-trained model on a specialized dataset of images containing roadwork objects from different cities from the Roadwork Dataset. However, many images in our dataset contain missing or low-quality annotations, and there are also inconsistent labeling issues. In constructing our dataset, irrelevant categories and images without annotations are removed, yielding 15 categories of roadwork objects and 4,908 images, from which we create training/validation/testing splits of sizes 70%/10%/20%. The distribution of objects in the dataset are shown in Figure 12.

Category	Train (3435)	Val (492)	Test (981)
Cone	7604	1203	2224
Fence	880	139	411
Drum	1189	150	1141
Barricade	1436	185	566
Barrier	1275	212	401
Work Vehicle	3106	457	870
Vertical Panel	5175	879	1912
Tabular Marker	4269	567	1562
Arrow Board	212	29	105
TTC Message Board	90	14	33
Other Roadwork Objects	191	20	63
Guide Sign	420	59	59
Road	2030	276	697
TTC Sign	2500	320	681
Work Equipment	280	44	41

Figure 12: Distribution of objects in the baseline dataset.

Augmented Datasets: We built several augmented datasets according to different strategies for evaluation: random Copy-Paste, Geometry- Paste, and GeometryPaste blended. In the random Copy-Paste dataset, objects are pasted onto randomly chosen locations and randomly scaled to 0.1 - 2.0 times their original size. The GeometryPaste and GeometryPaste blended datasets are constructed using our method, with the only difference being that GeometryPaste blended contains both blended and nonblended images, whereas GeometryPaste contains only nonblended images. We utilized 27 TTC Message Boards 64 times each, generating 1,728 new training images for the random Copy-Paste and GeometryPaste datasets and 3,456 new training images for the GeometryPaste blended dataset.

Training and Evaluation: We employ the Mask2Former[26] architecture with a ResNet-50 backbone[27] pre-trained on COCO to evaluate our method. Mask2Former builds on MaskFormer[28] and utilizes the Detectron2[29] framework. We follow the baseline Mask2Former settings, which include using the AdamW[30] optimizer, an initial learning rate of 0.0001, and a batch size of 16. We fine-tune on each dataset for 45k iterations on 8 Bridges2[31] GPUs. The baseline training regime utilizes only the original

[26] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 1290–1299.
[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
[28] B. Cheng, A. Schwing, and A. Kirillov, "Per-pixel classification is not all you need for semantic segmentation," Advances in Neural Information Processing Systems, vol. 34, pp. 17 864–17 875, 2021.
[29] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," https://github.com/facebookresearch/detectron2, 2019.
[30] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," arXiv preprint arXiv:1711.05101, 2017.
[31] S. T. Brown, P. Buitrago, E. Hanna, S. Sanielevici, R. Scibek, and N. A. Nystrom, "Bridges-2: A platform for rapidly-evolving and data intensive research," in Practice and Experience in Advanced Research Computing, ser. PEARC '21. New York, NY, USA: Association for Computing Machinery, 2021.

dataset, whereas all training with augmented datasets consists of the augmented and original datasets. We report the overall AP and TTC Message Board AP scores.

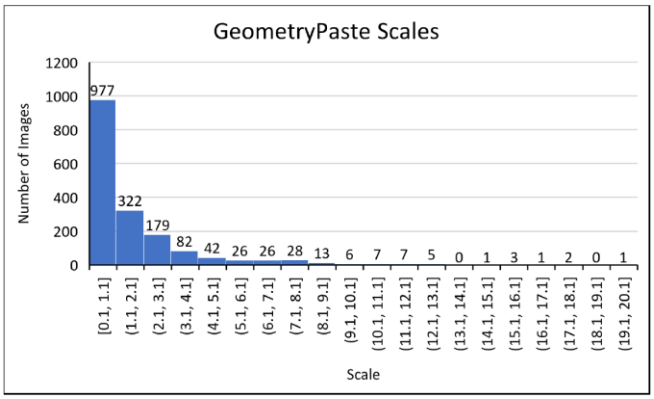

Figure 13: The object scales generated by GeometryPaste follow a right-skewed.

Results: Figure 13 shows that our method generates a right-skewed distribution of scales ranging from 0.1 - 20.1, with the vast majority being from 0.1 – 1.1. The overall appearance of the images generated is more realistic than the ones generated with random Copy-Paste. However, some objects are placed off the ground due to low-quality road segmentations. This, along with vanishing point prediction errors, resulted in some objects being scaled incorrectly since our scaling function depends on the coordinates of the road and vanishing point. Figure 14 shows each method's overall AP and TTC Message Board AP scores. We see that the highest AP score for the TTC Message board was 35.5, achieved by GeometryPaste with the model trained for 30k iterations. We also see that the highest overall AP score was 31.3, achieved by GeometryPaste with Gaussian blurring with the model trained for 35k iterations. Fig. 5 provides an example where GeometryPaste was the only method that detected the TTC Message Board in the given image.

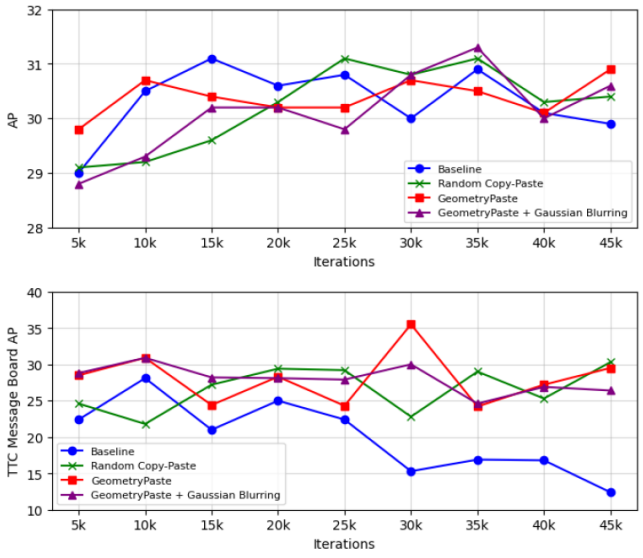

Figure 14: AP scores for each method. The highest overall AP score was 31.3, achieved by GeometryPaste with Gaussian blurring with the model trained for 35k iterations. The highest TTC Message board AP score was 35.5, achieved by GeometryPaste with the model trained for 30k iterations.

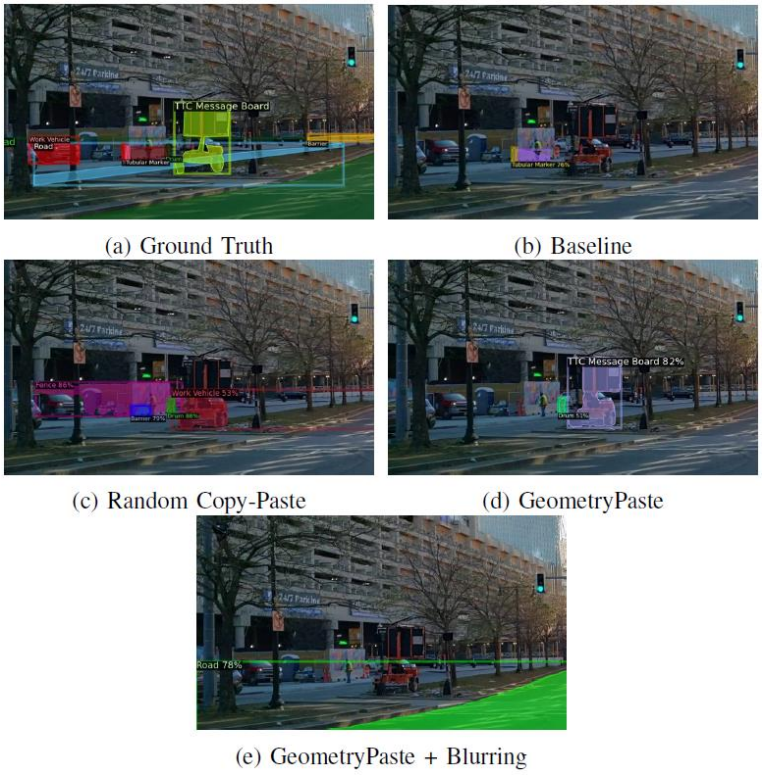

Figure 15: Visualization of results obtained from the model trained for 30k iterations. The TTC Message Board was only detectable with GeometryPaste without blending.

4.3. Road Marking Detection (Hailiang Zhu, Anurag Ghosh)

In the evaluation section of our methodology, we employed the widely recognized COCO metric[32] to rigorously assess the performance of the new detector. Specifically, we leveraged the Precision-Recall (PR) curve, a well-established evaluation measure for object detection models, to gain comprehensive insights into the detector's capabilities. We compute the Precision-Recall curve respectively for small, medium and large objects, the scale of which has been defined by COCO Metrics. The figure below explains the three kinds of lane markings in a single picture (Figure 16).

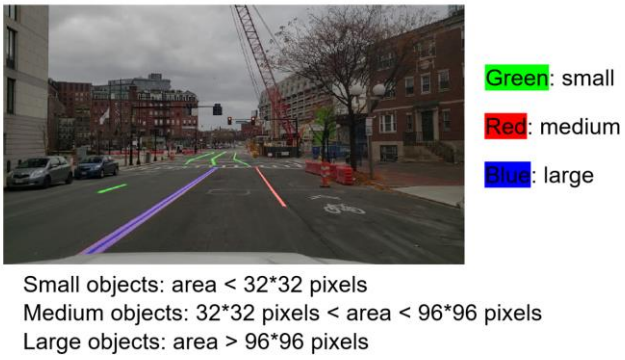

Figure 16: Visualization for Small, Medium, Large objects.

Performance on the Pseudo Labeled Test Set: We present the performance evaluation of the new instance segmentation model on the lane markings category across various object scales, namely small,

[32] Tsung-Yi Lin et al. "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.

medium, and large, based on the pseudo-labeled test set. The precision-recall curves serve as a robust tool to gauge the model's detection capabilities for each scale category. The findings reveal that the overall performance of the new model on lane markings is moderate, yet intriguing trends emerge when considering different object scales. Remarkably, the new model demonstrates a noticeable improvement in detecting large lane markings compared to medium and small ones. Although its performance on small and medium lane markings may not be as satisfactory, the substantial gain in accuracy for large instances is an encouraging outcome (Figure 17). The evaluation of the new detector on lane markings of different scales yields significant insights. When assessing the detector's performance at an Intersection over Union (IoU) threshold of 0.75, notable disparities are observed among small, medium, and large lane markings. These discrepancies imply that the detector's precision and recall are sensitive to the IoU threshold for objects of varying sizes, possibly due to the diverse complexities and appearances of lane markings at different scales.

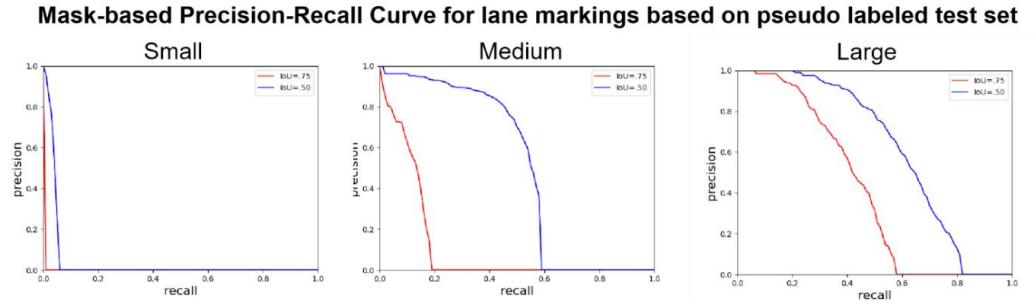

Figure 17: The unified detector performs best on large road markings with the pseudo labeled test set.

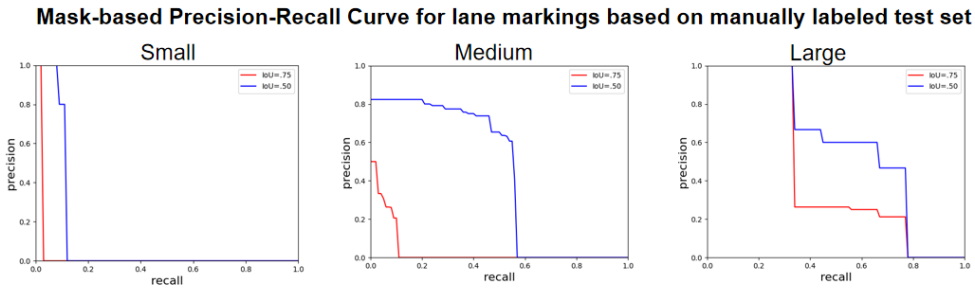

Figure 18: The unified detector performs best on large road markings with the manually labeled test set.

However, a compelling pattern emerges when the IoU threshold is set to 0.50. The new detector achieves a consistent overall average precision and recall of 60.0% for both medium and large lane markings. This suggests a robust performance for these two size categories under a lower IoU threshold. Interestingly, the precision for medium lane markings experiences a sharp decline, whereas the performance for large lane markings remains more stable. This finding implies that the detector excels in accurately detecting and localizing larger lane markings but faces challenges in precisely identifying and delineating medium-sized lane markings.

These results provide valuable insights into the detector's scale-dependent behavior and its sensitivity to the IoU threshold. They underscore the importance of carefully calibrating the IoU threshold based on the specific application requirements and characteristics of the target objects. Furthermore, the observed variations prompt further exploration into the factors influencing the detector's performance

on different scales. Addressing the challenges associated with medium lane markings through targeted fine-tuning may lead to overall performance enhancements and bolster its practical applicability in real-world autonomous driving scenarios.

Performance on the Manually Labeled Test Set: In the absence of manually labeled instances of lane markings in our original Roadbotics dataset, the introduction of pseudo labeled test set may result in increased noise. To address this limitation, we manually annotated visible lane markings in 50 randomly selected images from the Roadbotics test set. This manually labeled test set serves as a reliable ground truth, enabling a direct comparison with our new model's predictions. Subsequently, we generated a Precision-Recall curve based solely on the lane marking category using the manually labeled test set (Figure 18). This curve allows us to evaluate the new detector's performance in comparison to human-perceived lane markings and provides a basis for comparison with the results obtained from the pseudo labeled test set.

Although the Precision-Recall curve obtained from the manually labeled test set follows a similar trend to the one derived from the pseudo labeled test set, the jaggedness of the curve could be attributed to the limited number of test images. Despite this, the analysis reveals significant distinctions in the performance of the new detector across small, medium, and large lane markings when evaluated with an IoU threshold of 0.75. When the IoU threshold is set to 0.50, it becomes evident that the detector demonstrates an overall average precision and recall of 60.0% for both medium and large lane markings. Notably, the precision for medium lane markings experiences a sharp decline, while the performance on large lane markings remains comparatively more stable.

Figure 19: Detection model performance as illustrated by the receiver operating characteristic curve and confusion matrix. The results on the left column are for the scenario where images from all cities form the training and testing data sets, while the right column has cities split into training and testing.

4.4. Roadwork Detection and Localization (Michael Cardei)

For the Mask R-CNN instance segmentation model, we conduct an analysis across multiple intersection-over-union (IoU) thresholds. We computed the Average Precision (AP) at specified IoU values of 0.50 and 0.75, as well as across the range of 0.50 - 0.95. These metrics are also obtained for objects of different

sizes: small, medium, and large depending on the area. Additionally, for each class, we calculate the mean Average Precision (mAP) overall, and across the different size categories. To measure the performance of work zone localization we use a variety of metrics including precision, recall, F1 score, dice coefficient (DSC), and IoU. Precision is the ratio of true positives (TP) to the sum of true positives and false positives (FP). Recall is the ratio of true positives to the sum of true positives and false negatives (FN). The F1 score considers both precision (P) and recall (R), providing a single metric that unifies both. The F1 metric is defined as $F1 = \frac{2 \times P \times R}{P + R}$. The dice coefficient metric measures the overlap between the predicted and ground truth masks. For sets X (prediction) and Y (ground truth), the DSC is computed as $DSC = 2 \times \frac{|X \cap Y|}{|X| + |Y|}$.

Detection: The detection models' performance in same city and different-city settings is shown in Figure 19. When all images from all cities are split between training and testing the AUC is 0.99. This demonstrates very effective work zone identification abilities. When tested on images from cities not included in training the AUC decreased a negligible amount of 0.01 demonstrating the generalization proficiency. For reference, an AUC of 1.0 implies the model has perfect classification abilities, and an AUC of 0.5 means that the model has no discriminative abilities between positive and negative classes. The confusion matrix compares the predicted classification with the ground truth. We observe comparable classification tendencies across both settings with a minimal increase in false positives when testing on unseen cities. However, there is an overall accurate classification ability by the detection model.

Localization: For the task of instance segmentation the Mask R-CNN achieves an AP of 0.363 when averaged over all classes, object sizes, and IoU thresholds from 0.50 to 0.95. With an IoU threshold of 0.5 the AP is 0.562, and 0.403 with a threshold of 0.75 (Figure 20).

IoU	Area	MaxDets	AP	Value
0.50:0.95	all	100	AP	0.363
0.50	all	1000	AP	0.562
0.75	all	1000	AP	0.403
0.50:0.95	small	1000	AP	0.169
0.50:0.95	medium	1000	AP	0.371
0.50:0.95	large	1000	AP	0.523

Figure 20: Average Precision across all objects with vary areas and IoU thresholds.

The performance for segmentation across different classes is shown in Figure 21. The metrics include mean Average Precision (mAP) overall, as well as by size (small, medium, large). The results vary across classes with the highest-performing classes being arrow-board, drum, and road all having an mAP above 0.5. The rest of the classes' mAP range from 0.2 - 0.5 with only Non-Work person, Guide Sign, Tubular marker, and work equipment having a mAP less than 0.2. The results vary potentially due to the common size of the object or when difficult contexts are required such as with workers and nonworkers.

Category	mAP	mAP_s	mAP_m	mAP_l
Cone	0.353	0.261	0.542	0.56
Fence	0.289	0.139	0.146	0.345
Drum	0.501	0.321	0.657	0.746
Barricade	0.268	0.113	0.296	0.396
Barrier	0.379	0.144	0.295	0.614
Work Vehicle	0.361	0.041	0.282	0.44
Vertical Panel	0.457	0.282	0.671	0.81
Tubular Marker	0.4	0.237	0.568	0.666
Arrow Board	0.618	0.288	0.505	0.729
TTC Message Board	0.411	0	0.421	0.442
Guide Sign	0.177	0.07	0.277	0.248
Road	0.605	N/A	0	0.609
Non-Work Vehicle	0.325	0.19	0.394	0.444
TTC Sign	0.349	0.186	0.468	0.617
Work Equipment	0.113	0	0.065	0.189
Worker	0.296	0.154	0.381	0.497
Non-Work Person	0.09	0.055	0.131	0.329
Lane Ends Sign	0.533	0.396	0.572	0.733

Figure 21: Mean Average Precision overall, and per size category for individual class categories.

Segmentation Metric	Model Output	Convex Hull Transformed
IoU	0.382	0.400
Dice Coefficient	0.503	0.520
Precision	0.483	0.504
Recall	0.612	0.652
F1 score	0.503	0.520

Figure 22: Zone localization performance for instance segmentation mask output and Convex Hull transformed masks.

In order to evaluate the method's ability to localize diverse representations of work zones a new class called 'work zone' was created and manually annotated. This class contains samples over 130 images. We evaluate the direct instance segmentation output after applying a binary filter alongside the same ouptut with the Convex Hull algorithm applied to it. We observe an increase in every metric when the Convex Hull algorithm is applied as can be observed in Figure 22.

5. Recommendations

Automatic detection of roadwork zones from camera data is an extremely challenging problem. A challenge of developing computer vision and machine learning methods is the need for a comprehensive dataset of roadwork zones. For generalization, the dataset should have many hundreds of thousands of images of roadwork zones due to their irregular appearance, but also because roadwork objects (e.g., work vehicles, work equipment, etc.) appear differently in many states and even cities across the country. The dataset also needs to encompass different road environments, weather conditions, and light conditions. Roadwork vehicles equipped with cameras would be great candidates for capturing roadwork images. So would other public and service vehicles that are routinely on the road such as public works vehicles, buses, waste disposal trucks, etc.

> *Recommendation 1: DOTs actively pursue the capture of imaging data in roadwork zones.*

Another aspect needed for a roadwork dataset is annotations (labels and segmentations). Although we demonstrated some success with a unified detector, annotations should be performed manually to establish a reliable ground-truth for model training. This is a labor-intensive and expensive task to perform.

> *Recommendation 2: DOT sponsors grants or challenges to specifically have the dataset images manually annotated.*

In this work, it was demonstrated that computer vision and machine learning can be utilized to automatically detect roadwork zones to report information about roadwork zones to a driver, driver-assist system, or autonomous vehicle navigation system. Other drivers, especially those without a detection system, would also benefit from the information if it could be easily shared.

> *Recommendation 3: Support and invest in infrastructure that wirelessly relays and transmits information about roadwork zones.*

6. Conclusion and Future Work

This study addressed mobility issues caused by roadwork zones. The approach taken was to automatically detect roadwork zones from a vehicle's camera data so that the driver or vehicle can take appropriate actions. We found that there is not much work done in this area of computer vision and machine learning. Consequently, a novel dataset was created to train machine learning models and algorithms were developed to supplement the dataset and perform roadwork detection and localization. The dataset consists of 8,556 real world images where each image has associated manual object segmentations, object labels, and image descriptors. Approximately 20 roadwork object classes were identified for annotation.

The current state of the roadwork dataset has been made available. The dataset will be expanded by adding synthetically created roadwork images using the GeometryPaste method that was developed as part of this study. This method copies manual segmented objects into an image with correct placement (on the road) and with the correct size. It was shown that training with a mix of real-world images and synthetic images can improve detection results. Additionally, we plan on inviting researchers to add their own images or annotations to the dataset. Rather than relying solely on manual annotations, we investigated the use of a unified detector to merge object classes from multiple datasets. Results were promising with road markings. Finally, a method was developed for roadwork detection and localization, which showed very promising results.

To further improve automatic detection and localization of roadwork zones, the below areas should be considered for future work.

- Expand the roadwork dataset by:
    - Including images from different cities, captured during different weather conditions and light conditions,
    - Including augmented with pasted manually annotated objects with geometric scaling and context (GeometryPaste method),
    - Capturing images with work vehicles,
    - Enabling people to contribute manual annotations.
- Further develop the GeometryPaste method to also alter the appearance of objects to match the conditions (e.g., weather and light) in the destination image.
- Further develop the detection and localization method for improved accuracy.
- Explore the use of labels and text descriptions with natural language processing to understand the context of the road work zone.
- Develop a universal wireless infrastructure for transmitting and receiving roadwork zone information in a standardized manner.