International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

Road Damage Detection

Alekhya N[1], Dr. Madhumitha K[2]

[1,2]Department of Computing Technologies
SRM Institute of Science and Technology
Chennai, India
[1]an0610@srmist.edu.in, [2]madhumik1@srmist.edu.in

Abstract

Damages to road pavements have massive impacts on driving comfort, endanger safety, and can result in accidents. Traditional approaches to detecting damage, including hand inspection and mounted sensors, are not suitable for monitoring large areas. An alternative solution is to use street-view as a comparatively cheap option that offers current road data in the city. The proposed paper suggests a better system of pavement damage detection using YOLOv5 and street-view images. The model incorporates a Generalized Feature Pyramid Network (Generalized-FPN) to allow cross-layer and cross-scale feature fusion to improve the accuracy in detecting large distress targets. More accurate bounding box regression is done by a diagonal Intersection over Union (IoU) loss function, and the predictions and regressions are decoupled by a Head structure. The experimental findings prove that the given methodology is effective in strengthening the weak feature fusion on the spatial levels and shows higher results in detecting pavement distress in multi-scale and multi-level street-view images.

Keywords—Generalized Feature Pyramid Network (Generalized-FPN), Deep learning, Diagonal IoU Loss, Object Detection, Pavement Damage Detection, Street-View images, YOLOv5.

1. Introduction

Road damage detection has emerged as a massive issue in the management of urban infrastructure. The condition of roads worsens to pose a significant threat to people and increase mobility barriers. The intersection of computer vision, civil engineering, and machine learning has resulted in high-quality solutions to detect and classify classes of road damage, such as cracks, potholes, rutting, and surface wear. Early detection and categorization of such damage allow a more cost-effective maintenance program, which minimizes the cost of repair and maximizes the lifespan of road networks. As urbanization and the growth of traffic congestion occur, manual inspection by trained people is no longer adequate. These conventional approaches, which tend to be ground-based or multi-sensor vehicle systems like laser scanners or ultrasonic testers, are time-consuming, labor-intensive, and may at times be disruptive to traffic. Moreover, they tend to bring subjectivity to the process, which results in irregularity in the categorization of damages and severity evaluation. The recent developments in the field of AI and deep learning have caused a revolution in road damage detection. With the aid of algorithms that can process vast numbers of images, researchers and engineers are automating the detection process in a faster and more accurate manner. Two models of deep learning, which have become two of the most popular frameworks in this area, are YOLOv5 and Detectron2, and there are certain differences in their benefits. YOLOv5 is commonly known as extremely fast in processing images, so it can be used in real-time in the work of on-vehicle or drone-based surveillance. Its single-stage detection architecture has enabled it to provide fast inference with minimal loss of accuracy. On the other hand, Detectron2 (a Faster R-CNN model) has the ability to localize and robustly segment, which is particularly useful in the perception of details in a scene and complex object classification. Detectron2 is a valuable asset in object detection owing to its modular and configurable framework and the generation of region-based convolutional neural networks (R-CNN). The networks produce region proposals that are likely to hold objects, which require less area to be searched than exhaustive image scanning techniques. This area-wise method enables Detectron2 to achieve more efficient utilization of computing resources to enhance speed and accuracy. It also uses Feature Pyramid Networks (FPN) to allow it to identify objects at varying scales. Its anchor-based detection approach guarantees a higher performance in detecting features of different sizes- a necessary feature in the analysis of different patterns of pavement damage. Also, Detectron2 can use a range of backbone networks: ResNet and MobileNet, among others, which enables users to customize models to their needs to balance speed, accuracy, and resource usage. This is most effective in the ability to adapt models to various environments, such as urban, rural, and variable terrain with changing road conditions. Detectron2 can additionally be trained with multiple GPUs and advanced data augmentation methods, allowing the generation of effective models that perform well in generalizing on unknown data. These aspects render Detectron2 applicable to the construction of large and high-performance infrastructure monitoring systems. Compared to Detectron2, YOLOv5 might not have a fine-grained segmentation and localization performance because of its simpler architecture. Nevertheless, this opposition is an opportunity: a hybrid model that would integrate the advantages of both models. Under such a system, YOLOv5 might be able to rapidly locate areas of interest in real time, whereas Detectron2 might be able to analyze such areas in detail, estimating the nature and extent of damage more accurately. This complementary approach not only enhances the performance of the system but also makes sure that essential information is not missed. It is also possible to fine-tune Detectron2 on particular datasets, and this increases its accuracy over different kinds of road damage, even in diverse conditions, such as low lighting, shadows, or different textures on the surface. Combining the two YOLOv5 and Detectron2 with the municipalities and infrastructure authorities can enjoy a more efficient monitoring pipeline. YOLOv5 has coverage over a wide area and in a short amount of time, whereas Detectron2 can be focused where specific attention is required. The combination enables smarter decision-making to maintain the roads, allocating the resources better, and minimizing the response time. The real-life application of Detectron2 in road damage surveillance brings out the transformative nature of the technology in the current infrastructure management. With cities getting larger and larger, the need to have smarter systems that are capable of managing even greater complexity in road conditions emerges with greater urgency. Damage needs to be detected and classified accurately to plan how to carry out maintenance and prioritize repairs. The ability of Detectron2 to withstand the changing environmental factors, including the light conditions, different textures, and types of roads, further entrenches its position in contemporary road maintenance plans. Combined with fast-performing models such as YOLOv5, it creates the basis of an effective, efficient, and scalable automated pavement condition measurement. Finally, the deep learning systems (YOLOv5 and Detectron2) have strong potential in detecting road damage automatically. Both frameworks have their own advantages: YOLOv5 would be better at real-time monitoring, whereas Detectron2 would be better at analysis and flexibility. An integrated solution based on a combination of the two models can contribute to the performance of road inspection systems so that they can be more responsive, accurate, and in accordance with the demands of the intelligent infrastructure management in a rapidly urbanizing environment.

2. RELATED WORK

The importance of detecting road damage has attracted a lot of attention over the past years because of the role of road damage detection in infrastructure maintenance and road safety. The initial studies in this field were based mainly on the classical procedures of image processing. Crack and pothole detection methods like edge detection, texture analysis, and contour-based segmentation [10][9] were also predominantly used. Although these methods proved to be computationally efficient and easy to execute, they failed to hold strong in different lighting conditions, shadows, and intricate road textures. With the deep learning method, road damage detection has transformed and has made it possible to extract features automatically and enhance detection quality. This capability to learn hierarchical representations directly from images has made Convolutional Neural Networks (CNNs) the standard method [8]. YOLO (You Only Look Once) has become one of the popular CNN-based models that are used in real-time detection. YOLOv3, introduced by Redmon and Farhadi [3], uses one neural network over the image, which makes it fast at detecting objects. Subsequent versions, such as YOLOv4 [4] and YOLOv5 [5], have increased speed and detection accuracy, and can be used in embedded systems in human applications, e.g., vehicle-mounted cameras and drones. Deep learning frameworks such as Detectron2 have been extensively used when dealing with tasks that demand accuracy in segmentation and localization. Detectron2 [7] is a Facebook AI Research developed computer-assisted model that supports models like Faster R-CNN and Mask R-CNN [6], providing pixel-based instance segmentation and high detection accuracy. The usefulness of the Mask R-CNN in detecting pavement cracks was demonstrated by Yang et al. [2], and high accuracy was achieved in the problem of complex scenes in urban settings. Detectron2 itself, however, is computationally heavy, and this may also be a constraint on its use in edge-case or non-real-time applications. The recent studies have also considered the hybrid solutions that can unite the advantages of YOLO and Detectron2. These models make use of YOLO real-time detection and Detectron2 precision in segmentation, a balance between accuracy and speed [1][8]. These developments notwithstanding, a number of challenges have been identified, such as generalization of various types of roads, weather conditions, and light situations, and classification of multi-class damages and combination with Geographic Information Systems (GIS) to aid in maintenance planning. In conclusion, despite the fact that up-to-date literature illustrates that the existing state on the matter of automated road damage detection has made a lot of headway, there are still significant gaps in research. These comprise enhancing the robustness of the model in different environmental factors, higher efficiency in deployment of edges, the possibility of multi-class detection of damage with a measure of the severity of the damage, and spatial mapping to perform strategic maintenance. These gaps are the main aim of the proposed research.

3. PROBLEM DEFINITION

Modern transportation systems rely solely on road infrastructure, which is important in the process of economic growth and development. Nevertheless, roads are exposed to constant wear and tear with time, depending on certain factors like unfavorable weather conditions, traffic load, heavy vehicles, and natural aging of materials. This deterioration results in the establishment of surface defects such as potholes, cracks, rutting, and similar anomalies that not only lower driving comfort but also cause serious damage to road safety. Unless such problems are timely identified and fixed, they may lead to high-cost maintenance, higher accident rates, and vehicle damage, thus leading to significant social and economic costs. At present, the detection and evaluation of road damage is done primarily by manual inspection of roads by maintenance staff. These inspections are lengthy, tedious, and costly. Furthermore, manual processes may be bad in terms of inconsistency as a result of human factors and subjectivity, and thus, detecting vital damages may not be accurate or in time. Consequently, road defects can take a long time before they are detected, thus increasing the risk to road users and ineffective use of maintenance resources. To overcome such problems, automated, reliable, and scalable road damage detection systems are increasingly demanded. The future of computer vision and deep learning technologies has promising opportunities to come up with intelligent systems that can monitor and classify pavement distress in real-time. These automated solutions can greatly contribute towards improving the accuracy and efficiency of infrastructure maintenance programs to enable proactive interventions, which would lead to improved road safety, lower cost of maintenance, and also increase the life of the road networks.

4. Dataset Generation

In this research, a filtered dataset was used, which was compiled based on the Road Damage Dataset (RDD) and surrogate CFD road image sets to produce more than 10,000 annotated images that are representative of a variety of pavements, lighting, and traffic scenarios. This was done to have a representative sample of common roadway distresses, such as potholes, longitudinal cracks, transverse cracks, and rutting, that would be useful for object detection and instance segmentation.

A. Data Sources and Acquisition

Images were sourced from:

1. RDD: Smartphone-captured roadway scenes from multiple regions, providing varied asphalt textures and environmental conditions.
2. CFD collections: Additional urban/rural scenes to increase diversity in background clutter, lane markings, and occlusions.

B. Annotation Protocol

- Object-level annotations: Bounding boxes generated in LabelImg to facilitate training of fast detection (YOLOv5).
- Instance-level annotations: Polygons annotated in CVAT to pixel-level masks used by Detectron2.
- Class labels included pothole, longitudinal crack, transverse crack, and rutting (they can be extended to several more in further work). In order to minimize noise, annotators were guided by a short guideline (minimum pixel size, occlusion handling, and treatment of ambiguous textures). A spot-check procedure (about 10 percent of images) confirmed label consistency; discrepancies were resolved by a second pass (TM note briefly what your actual QA rate is, when possible).

C. Preprocessing and Balancing

Preprocessing ensured consistent model input, improved generalization: Normalization and resizing to standard input dimensions and channel statistics. Controlled rotations, flips, and contrast adjustment are used to augment to simulate viewpoint and variation in illumination without label drift. Balancing of classes through oversampling of the underrepresented classes (e.g., thin cracks) to reduce bias toward easily spotted, high-contrast potholes. Data division: 70 percent training, 15 percent validation, 15 percent testing (where feasible, stratified to maintain class distribution between splits).

D. Quality Control and Filtering

Extremely over-/under-exposed pictures, pictures featuring motion, or pictures without a road were filtered. In order to be consistent, duplicate or similar frames of burst captures were pruned. Label drift and class imbalance were checked by periodic validation set runs; targeted augmentation or CFD-supplied supplementation of fine cracks and shadowed scenes was provided when required.

E. Dataset Characteristics

The data set was a collection of more than 10,000 annotated images obtained on the RDD and CFD repositories. Smartphones, CCTV cameras, and drones were used as sources of images, and their variety guaranteed versatile perspectives and solutions. The data captured was taken under diverse conditions, which included daylight, overcast skies, shadows, glare, and complicated scenes that included lane markings, vehicles, and pedestrians. The images were bound with labeling polygonal masks and bounding boxes with the help of the tools of LabelImg and CVAT, and allowed both detecting and segmenting the images. The data was separated into 70 percent training, 15 percent validation, and 15 percent testing sets with all images resized to 1280 x 720 to ensure consistency in model training.

F. Ethical, Privacy, and Compliance Considerations

Where geolocation metadata is stored in a form suitable for mapping, it is stored in coordinates and presented at a fit-for-purpose granularity (e.g., road segment level) to limit the privacy exposure. The processing of datasets follows the local data-protection standards; the data are exported in reports displayed in the dashboard, and the personally identifiable features are anonymized (faces/plates are not the subject of study). Any subsequent public release would come with a usage license and a brief data card recording sources, planned use, and restrictions.

G. Limitations and Mitigation

Even in such a wide dataset, there are still difficulties with low-light, glare, and micro-crack scenarios. This was partially counteracted through controlled augmentation and class balancing; in future extensions, night/infrared imagery, bad weather (rain/ fog), and other classes like edge erosion and subsurface indicators will be added.

5. METHODOLOGY

To develop the Road Damage Detection System, the design was completed on the basis of the Agile Scrum methodology that enabled extensive functional modules to be refined and integrated throughout consecutive sprints. Every sprint was aligned with the set goals, user stories, and deliverables that were quantifiable, thus providing a step forward towards a system that was ready to production. The workflow as a whole is shown in Fig. 1.

H. Agile Development

The system was designed to have 2 large sprints:

- In the Sprint I focused on preparation of the datasets, annotation of the datasets, the development of a baseline model, and the prototyping of the architecture.
- Sprint II focused on the integration of hybrid models, edge optimization, GIS mapping, and dashboard implementation to address the real world. The Scrum framework allowed for planning sprints, conducting daily stand-ups, and retrospectives, which supported the flexibility and continuous improvement during the development process.

I. Data Preparation

The dataset was assembled on the basis of RDD and CFD repositories, totaling more than 10,000 annotated images of potholes, cracks, and rutting. Preprocessing of data consisted of: Augmentation: Rotations, flips, and contrast variation to improve generalization. Annotation: Bounding boxes and polygons created with LabelImg and CVAT to label potholes, longitudinal cracks, transverse cracks, and rutting. Balancing: Minority classes are oversampled to enhance stability within the categories. Splitting: Training, validation, and testing (15, 15, and 70 percent, respectively). This was to guarantee rich, well-organized data that can be used in object detection as well as segmentation.

J. Model Development

A hybrid pipeline was implemented to balance speed and accuracy: YOLOv5 served as the main detection model, which gives fast predictions of bounding boxes. Detectron2 improved these detections by instance segmentation and severity estimation, which provided pixel-level crack and pothole detail. This two-step workflow minimized false positives to a large extent and improved the quality of segmentation. Baseline YOLOv5 had a precision of 90% and recall of 88% and mAP@0.5 of 85%, and Detectron2 had an IoU of 84% on segmentation tasks

K. System Architecture

The general architecture of the proposed road damage detection system is illustrated in Fig. 1. The framework is based on a two-phase hybrid pipeline, in which images captured from CCTV cameras, drones, or smartphones are first processed by YOLOv5 for fast bounding-box detection. A Filter and Forward module then sends high-confidence regions to Detectron2, which does the pixel-level segmentation and severity classification. Damages are geotagged using a GIS engine, and the results are made available to stakeholders through a web-based dashboard and API services. This modular architecture means that real-time performance is achieved on edge devices while still achieving high accuracy through granular segmentation on cloud or hybrid deployments.

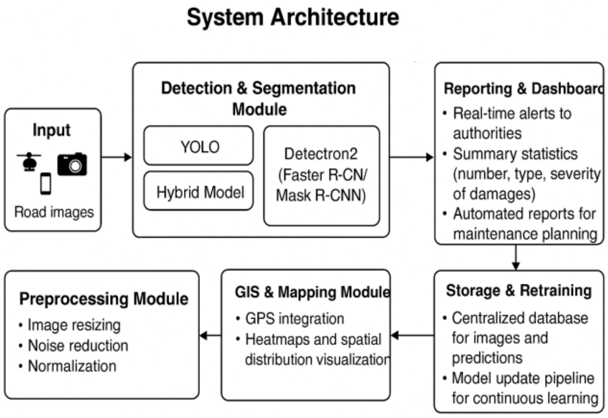

Fig. 1. System architecture for road damage detection using YOLOv5 and Detectron2.

- Input Layer: Accepted single images or streams from smartphones, drones, or CCTV cameras.
- Preprocessing Layer: Applied augmentation and batch normalization before inference.
- YOLOv5 Module: Performed bounding box detection with confidence thresholding.
- Filter and Forward Module: Passed high-confidence detections to the segmentation stage.
- Detectron2 Module: Conducted pixel-wise segmentation and assigned severity scores (minor, moderate, critical).
- Geospatial Engine: Tagged detections with GPS metadata and plotted them using Leaflet.js and Google Maps API.
- Dashboard and API Layer: Provided role-based access, analytics, real-time alerts, and weekly reports in PDF/CSV formats.

L. Edge Delpoyment

To enable real-time, field-ready operation, the models were optimized for deployment on NVIDIA Jetson Nano devices:

- TensorRT conversion accelerated inference.
- INT8 quantization reduced computational load.
- Weight pruning minimized the memory footprint.
- These optimizations achieved 30 FPS on 720p video streams, validating the feasibility of lightweight edge deployment in smart city infrastructure.

M. GIS Integration and Dashboard

Geospatial visualization was integrated to assist maintenance authorities in damage prioritization. The dashboard supported:

- Heatmaps of critical damage zones.
- Real-time alerts with severity scoring.
- Interactive filtering by location, damage type, and time.
- Exportable analytical reports (PDF, CSV).

- In addition, it was essential to make sure that non-technical stakeholders such as planners and inspectors could gain sensible insights from the detection pipeline.

6. EXPERIMENTAL SETUP

The proposed road damage detection system was deployed on high-performance servers as well as on low-power edge devices to justify practicality. Model training was performed on a GPU-based workstation (NVIDIA RTX series GPU, 16 GB RAM, Ubuntu 20.04), and deployment experiments were performed on a low-power NVIDIA Jetson Nano device. The software stack consisted of Python 3.9, PyTorch 1.12, and CUDA/cuDNN libraries. A pre-trained YOLOv5 weights were used for preliminary object detection, while Detectron2 (Faster R-CNN backbone) was used to perform fine-grained segmentation and severity categorization. Edge device optimization was performed by TensorRT conversion, INT8 quantization, and weight pruning. The dataset, which was developed by collecting from the RDD and CFD repositories, comprised around 10,000 annotated images. Images were resized, normalized, and augmented (rotation, flips, contrast). The dataset was divided into training (70%), validation (15%), and testing (15%) sets in a balanced way in terms of potholes, longitudinal cracks, transverse cracks, and rutting. Training was done for 50 epochs with a batch size of 16 using the Adam optimizer and an initial learning rate of 0.001. Performance was evaluated with precision, recall, mean Average Precision (mAP), and Intersection over Union (IoU).

7. PERFORMANCE ANALYSIS

The hybrid pipeline showed high results in detection and segmentation. YOLOv5 reported a precision of 90, a recall of 88, and mAP of 0.5, whereas Detectron2 offered pixel-level accuracy at up to 92 IoU on large, well-defined damages. YOLOv5 with detectron2 minimized false positives and increased segmentation quality when compared to single-model baselines. On the Jetson Nano, 720p video streams at 30 FPS could be sustained with edge deployment, which validates vehicle-mounted or drone-based monitoring applications. The integration of GIS also facilitated geospatial visualization of the damage on the roads, and this facilitated actionable knowledge on the bench of maintenance priority. Nevertheless, some restrictions were noticed. Under low-light conditions, performance was worse, and fine cracks and micro-damages were sometimes missed. Segmentation accuracy was also lower in strong glare or occlusions. Nevertheless, with these difficulties, the system proved capable of detecting major road damage safely and also exhibited scalability due to its modular nature in terms of microservices architecture. In sum, combining the hybrid deep learning models with edge optimization and GIS mapping emphasizes the potential of the system to serve as a practical tool to monitor the smart city infrastructure.

8. RESULT

N. Quantitative Evaluation

The models were assessed using Precision, Recall, F1 Score, mAP, and IoU. YOLOv5 served as the baseline detection model, while Detectron2 (Faster R-CNN backbone) provided enhanced pixel-level segmentation.

TABLE I: QUANTITATIVE EVALUATION OF YOLOV5 AND DETECTRON2

Metric	YOLOv5(%)	Detectron2(%)
Precision	90	94
Recall	88	91
mAP@0.5	85	93
IoU	84	92

These results demonstrate that Detectron2 has superior performance to YOLOv5 in both detection and segmentation tasks, especially in detecting large, well-defined damages like potholes and rutting. However, fine cracks and shadowed damages were more challenging, indicating the need for further dataset expansion and fine-tuning.

O. Deployment Performance

The system was experimented with the help of both GPU servers and NVIDIA Jetson Nano edge devices: Within GPGP servers, the pipeline was efficient with large batches, which is appropriate in large-scale road surveys. Inference on Jetson Nano, inference on 720p streams reached 30 FPS, which is sufficient to support real-time vehicle-mounted or drone-based monitoring.

P. GIS and Dashboard Integration

The ability to connect the system to GIS allowed geospatial mapping of road damages, severity-based heatmaps, and exportable reports (PDF/CSV). Inspectors and city planners tested the dashboard and confirmed that it enabled them to plan and decide on maintenance.

Q. Qualitative Observations

- **Strengths**: High segmentation accuracy for large, visible damages; effective real-time deployment.
- **Limitations**: Micro-crack performance was negatively affected by light/glare conditions
- **Future Potential**: Predictive maintenance through temporal data analysis, expanded classification (e.g., edge erosion, subsurface damage), and feedback-based retraining.

9. Conclusion

This paper presented a road damage detection system that combines the rapid object detection proposed by YOLOv5 and the accurate instance segmentation proposed by Detectron2. The tested approach demonstrated good performance based on standard assessment measures, such as precision, recall, mAP, and IoU, and delivered real-time inference at 30 FPS on 720p video streams on a microservices-based deployment with NVIDIA Jetson Nano. These findings confirm that it is viable to implement the system in resource-constrained settings like vehicles and drones. In addition to detection, GIS mapping combined with a web-based dashboard facilitated usability through real-time alerts, prioritization (based on severity), and analytics insights in the hands of city planners and maintenance authorities. This points to the fact that the system can potentially be used to decrease inspection expenses, enhance road safety, and aid data-driven repair planning. Further research will center on extending the system to support multi-spectral sensing (to operate at night) and more forms of damage, lightweight systems to use on the edges, and predictive analytics to allow proactive maintenance. Adding the feedback control mechanisms and crowdsourced annotations could also be useful to add accuracy and scalability on a wide range of geographies. Altogether, the suggested system shows how computer vision and deep learning can evolve automated road condition monitoring and become part of the more efficient and sustainable urban infrastructure management.

References

1. Maeda, H., Sekimoto, Y., Seto, T., Kashiyama, T., & Omata, H. (2018). Road Damage Detection Using Deep Neural Networks with Images Captured Through a Smartphone. arXiv preprint arXiv:1801.09454.
2. Yang, F., Zhou, S., & Guo, Y. (2019). Automatic Pavement Crack Detection and Segmentation Based on Improved Mask R-CNN Model. Measurement, 145, 657-664.
3. Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. arXiv preprint arXiv:1804.02767.
4. Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y.M. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv preprint arXiv:2004.10934.
5. Jocher, G. (2020). YOLOv5 by Ultralytics. GitHub repository.
6. He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. Proceedings of the IEEE International Conference on Computer Vision, 2961-2969.
7. Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., & Girshick, R. (2019). Detectron2. Facebook AI Research.
8. Zhang, K., Cheng, H. D., Jiang, X., & Yang, Y. (2020). Unified Vision-Based Method for Automated Road Crack Detection Using Deep Convolutional Neural Network. Computer Aided Civil and Infrastructure Engineering, 35(1), 39-54.
9. Mohan, D., & Poobal, S. (2018). Crack Detection Using Image Processing: A Critical Review and Analysis. Alexandria Engineering Journal, 57(2), 787-798.
10. Amhaz, R., Chambon, S., Idier, J., & Baltazart, V. (2016). Automatic Crack Detection on Two-Dimensional Pavement Images: An Algorithm Based on Minimal Path Selection. IEEE Transactions on Intelligent Transportation Systems, 17(10), 2718-2729.
11. Kang, D., Koch, C., Tran, H. D., & Kim, C. (2018). Automated Pavement Crack Segmentation Using U-Net-Based Convolutional Neural Network. Construction and Building Materials, 201, 12-25