International Science and Technology Journal
المجلة الدولية للعلوم والتقنية

Volume 35 العدد
Part 1 المجلد
October 2024 اكتوبر

تم استلام الورقة بتاريخ: 6 / 2024/10م وتم نشرها على الموقع بتاريخ: 30/ 2024/10م

Urban Applications of Deep Learning: Potholes Detection Using YOLOv10 Model in Al-Bayda, Libya

http://www.doi.org/10.62341/ghra3523

Galal H. Senussi[1], Radwan A. Hiba[2]

[1]Department of Mechanical Engineering, Faculty Engineering, Omar Al Mukhtar University, Al Bayda-Libya.
[2]Department of Civil Engineering, Faculty Engineering, Omar Al Mukhtar University, Al Bayda-Libya.

galal.senussi@omu.edu.ly, radwan.bohiba@omu.edu.ly

Abstract

The city of Al Bayda is a regional center with many agricultural villages that lie in the northeast of Libya. Poor road maintenance, reaching the end of their service life, and high traffic loads have led to a marked deterioration in the condition of the city's roads. This research used open-source data from Kartaview and OSM (OpenStreetMap) integrated with YOLO (You Only Look Once) object detection model to detect potholes in the city's road network. Therefore, this study aims on improving road safety and efficiency related to maintenance, using volunteer-generated geospatial data along with the latest YOLOv10 model, to create a scalable and reliable method of pothole detection. Results Confirm image diversity and accurate annotation are crucial in model training to achieve mAP (Mean Average Precision) above 50%. This identifies an effective tool for pothole detection that may be scalable in the context of urban infrastructure management and is a cost-effective approach that can ensure traffic safety and early maintenance actions while reducing the environmental impact of vehicle downtime.

Keywords: YOLO, Deep Learning, Pothole Detection, Infrastructure Management, Kartaview, OpenStreetMap (OSM).

تم استلام الورقة بتاريخ: 6 / 2024/10م وتم نشرها على الموقع بتاريخ: 30/ 2024/10م

التطبيقات الحضرية للتعلم العميق: اكتشاف الحفر باستخدام نموذج YOLOv10 في البيضاء، ليبيا

[1]جلال السنوسي، [2]رضوان هيبه
[1]قسم الهندسة الميكانيكية جامعة عمر المختار، البيضاء–ليبيا
[2]قسم الهندسة المدنية جامعة عمر المختار، البيضاء–ليبيا

الملخص

مدينة البيضاء هي مركز إقليمي يضم العديد من القرى الزراعية التي تقع في شمال شرق ليبيا. وقد أدى سوء صيانة الطرق، ووصولها إلى نهاية عمرها الافتراضي، والأحمال المرورية العالية إلى تدهور ملحوظ في حالة طرق المدينة. استخدم هذا البحث بيانات مفتوحة المصدر من Kartaview و(OpenStreetMap) OSM متكاملة مع نموذج اكتشاف الكائنات (You Only Look Once) YOLO لاكتشاف الحفر في شبكة طرق المدينة. لذلك، تهدف هذه الدراسة إلى تحسين سلامة الطرق والكفاءة المتعلقة بالصيانة، باستخدام البيانات الجغرافية المكانية التي تم إنشاؤها بواسطة المتطوعين جنبًا إلى جنب مع أحدث نموذج YOLOv10، لإنشاء طريقة قابلة للتطوير وموثوقة لاكتشاف الحفر. تؤكد النتائج أن تنوع الصور والتعليق الدقيق أمران حاسمان في تدريب النموذج لتحقيق متوسط الدقة المتوسط (mAP) أعلى من 50%. وهذا يحدد أداة فعالة لاكتشاف الحفر والتي قد تكون قابلة للتطوير في سياق إدارة البنية التحتية الحضرية وهي نهج فعال من حيث التكلفة يمكنه ضمان سلامة المرور وإجراءات الصيانة المبكرة مع تقليل التأثير البيئي الناجم عن توقف المركبات.

الكلمات المفتاحية: YOLO، التعلم العميق، اكتشاف الحفر، إدارة البنية التحتية، Kartaview، OpenStreetMap (OSM).

Introduction

Al Bayda City is a significant regional hub that connects various agricultural villages in northeastern Libya **Fig. 1**. However, urban planning and infrastructure have seen little development in recent

تم استلام الورقة بتاريخ: 6 / 2024/10م وتم نشرها على الموقع بتاريخ: 30/ 2024/10م

decades. The city's urban planning is divided into three generations of 20-year spans, with the first generation implemented fully in 1968, while the second and third generations remain incomplete [1]. Social and economic activity depend on the road network, yet many roads have decadent as a result of inadequate maintenance and repairs as well as high traffic volumes. The study aims to develop an efficacious method for detection potholes in city roads using open-source data from Kartaview, under the OpenStreetMap platform for address this problem. A volunteer team, ODAT (OSM Ly and Digital Twin Al Bayda Team), has been visualizing the city, and this data can be analyzed using the YOLO (You Only Look Once) object detection model. The goal is creating a methodology approach for detect road potholes, promote traffic safety and improving maintenance efficiency. By providing a reliable pothole detection method, the study aims to reduce repair costs, environmental impact from vehicle stops and accident risks, while also serving as a model for other cities facing similar infrastructure challenges.

Fig. 2 Al Bayda city in northeastern part of Libya

Literature survey

Joseph Redmon et al. [2] suggested the YOLO (You Only Look Once) algorithm, which converted object detection into a single regression problem and transformed the field. YOLO predictions bounding boxes and class probability straight from an image in a single pass, in contrast to earlier techniques that depended on classifiers. With Fast YOLO processing at 155 FPS and the base model reaching 45 FPS, this integrated model achieves real-time processing speeds. YOLO has demonstrated better generalization across domains, which reduces the likelihood of false positives on background objects, even in the face of certain localization mistakes.

Arjun Paramarthalingam et al. [3] developed a pothole detecting software that gives visually challenged users tactile or audio feedback using the YOLO technique. Although their model had trouble recognizing away potholes, it managed to run at 30 FPS and attain an accuracy of 82.7%. Future developments will focus on increasing accuracy and extending detecting abilities in a variety of road situations.

Satish Kumar Satti et al. [4] achieved 98.27% mAP for pothole detection by introducing a cascade classifier with a visual transformer for realize potholes and traffic mark. Their technique improved road safety for high-risk locations even in the face of difficult circumstances, such as water-filled potholes and oscillate lighting.

Yashar Safyari et al. [5] performed a thorough analysis of pothole detection strategies, classifying them into machine learning, 3D points clouds, 2D image processing, and hybrid approaches. They found that deep learning techniques, especially U-Net, were quite accurate, get to 97% accuracy and showed a great deal of promise for reliable road condition monitoring.

Cuthbert Ruseruka et al. [6] created a vehicle-based YOLO-based pothole discovery algorithm that achieved 96.3% mAP and 93% accuracy. This method provides a cost-effective way to monitor roads in real time and has the prospects to be widely embrace by traffic authorities

Zakwan Syukri Mohd Shah et al. [7] YOLOv5-based pothole detection system was executed in real-time on a Raspberry Pi, achieve high accuracy in a range of weather and lighting conditions. It works well, but only at 2-3 frames per second, which emphasizes the necessity of performance optimization in high-speed situations.

Eldor Ibragimov et al. [8] described a deep learning-based Pavement Condition Index (PCI) counting that achieved a 95% crack detection accuracy. This method provides a reliable way for manage road infrastructure while organize pavement rating and reduce manual labor.

Sungan Yoon et al. [9] Improved Pothole Detection by Fusion in RGB and Transformed Disparity Map (TDM) Data in YOLO with 10.7% Performance Gain. This approach has an advantage over the 3D reconstruction methods, hence much faster and more feasible for real-time applications.

Nachuan Ma et al. [10] determined that deep convolutional neural networks (CNNs), especially in mixed 3D approaches, offer the most prospect after reviewing cutting-edge computer vision algorithms for pothole identification. In order to overcome data limitations in road surface analysis, they confirm the significance of unsupervised learning.

Mohan Prakash B et al. [11] Trained the YOLOX model for pothole detection with an average precision of 85.6% at small computational cost, its low power consumption and small size make it suitable to deploy on resource-constrained devices.

Sung-Sik Park et al. [12] compared models found that YOLOv4-tiny was the most successful YOLO model for pothole identification, with 78.7% mAP, when compared to YOLOv4, YOLOv4-tiny, and YOLOv5s. Future research will focus on enhancing correspondence in low-light and bad weather state of affairs.

Methodology

The methodology described in this paper is predicated upon the utilization of open-source data. The data in question was contributed to the Kartaview platform by the ODAT team, a voluntary organization operating within the city, thus ensuring a high level of reliability. Each image provided is correlated with its geographical coordinates, in addition to pertinent information such as the image number and path. This correlation aids in accurately identifying the locations of potholes, as illustrated in **Fig. 2** [13].

تم استلام الورقة بتاريخ: 6 / 2024/10م وتم نشرها على الموقع بتاريخ: 30/ 10/ 2024م

Table 1 Details of the performance of YOLOv10 types

Model	Test Size	#Params	FLOPs	APval	Latency
YOLOv10-N	640	2.3M	6.7G	38.50%	1.84ms
YOLOv10-S	640	7.2M	21.6G	46.30%	2.49ms
YOLOv10-M	640	15.4M	59.1G	51.10%	4.74ms
YOLOv10-B	640	19.1M	92.0G	52.50%	5.74ms
YOLOv10-L	640	24.4M	120.3G	53.20%	7.28ms
YOLOv10-X	640	29.5M	160.4G	54.40%	10.70ms

Fig. 2 Important information for dataset

Furthermore, the YOLO model is also open-source, which enhances the effectiveness of combining these two features for early pothole detection. This approach helps reduce traffic risks, enhance vehicle safety, and inform relevant authorities, especially since the city lacks regular maintenance and follow-up on road conditions.

To train the YOLO model, image labeling and location identification are necessary.

Roboflow was utilized for this purpose, offering a faster and more efficient process. The YOLO model is implemented in Python, and Google Colab Notebook was used as the working environment.

Fig. 3 illustrates the proposed methodology.

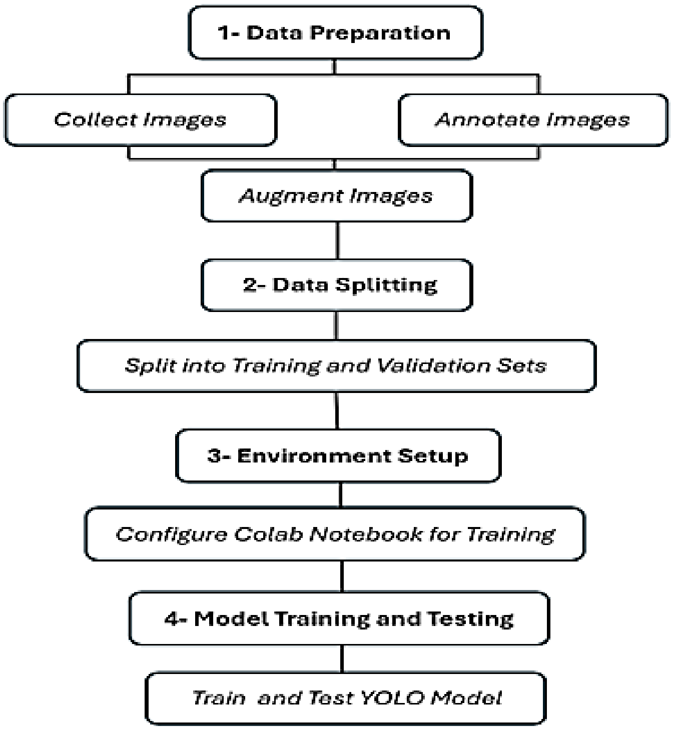

Fig. 3 flow-chart of the proposed methodology

YOLO v10

It is an algorithm used to detect objects in images based on CNN. Ultralytics has presented many models, the latest of which is YOLO v10. It is preferable to use the latest due to its continuous development to obtain the best accuracy with the least processing time. A comparison of YOLO models based on the latency and the number of parameters with COCO *AP*% (Common Objects In Context) (Average Precision) is show in **Fig. 4** [14]

YOLO v10 is divided into several models (N-S-M-B-L-X) where the efficiency increases gradually depending on the number of parameters, computing power requirements, average accuracy on the validation set, and the time taken to make a single pass through

تم استلام الورقة بتاريخ: 6 / 10/2024م وتم نشرها على الموقع بتاريخ: 30/ 10/2024م

the model. **Table 1** Shows details of the performance of YOLOv10 types. where YOLOv10-x was adopted in this study [14]

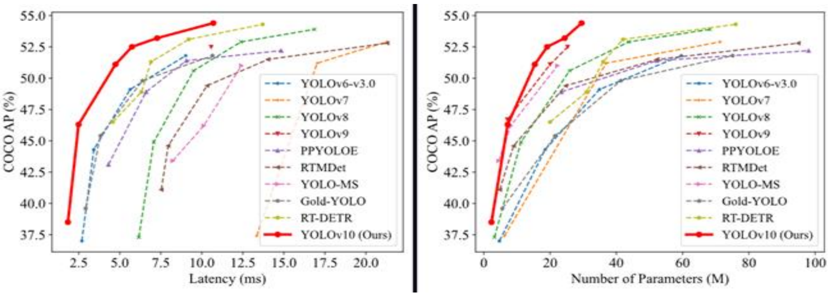

Fig. 3 Comparisons with others model of latency and size accuracy

Bounding Box Calculation and Loss Measurement in YOLO algorithm

Box of bounding is a form that represents an object in image; it has properties including height bh, width bw, class c, and center coordinates bx, by [15]. The location and dimensions of the bounding box are located based on the following equations

Center Coordinates of the bounding box:

$$bx = (2 \cdot \sigma(tx) - 0.5) + cx$$
$$by = (2 \cdot \sigma(ty) - 0.5) + cy$$

(cx) and (cy) represent the grid cell coordinates, and (σ) is the sigmoid function.

- **Width and Height of the bounding box:**

$$bw = pw \cdot (2 \cdot \sigma(tw))^2$$
$$bh = ph \cdot (2 \cdot \sigma(th))^2$$

(pw) and (ph) are the base dimensions of the bounding box.

- **Intersection Over Union (IOU)**

IOU is used to measure the accuracy of the bounding box by calculating the overlap between the predicted box and the actual box.

- **Loss Function**

After passing the image through the model, YOLO outputs several values, including object classes, box of bounding, and confidence levels. The total loss is calculated using:

Loss = $\lambda_1 L_{cls} + \lambda_2 L_{obj} + \lambda_3 L_{loc}$

تم استلام الورقة بتاريخ: 6 / 10/2024م وتم نشرها على الموقع بتاريخ: 30/ 10/2024م

(L_{cls}): Classification loss.
(L_{obj}): Objectness loss, that punishes wrong predictions of object presence.
(L_{loc}): Localization loss.
($\lambda 1$, $\lambda 2$, and $\lambda 3$): Weight

Experiment analysis

The Yolo model training process is carried out through the collected data to reach the best result where the accuracy depends on the mAP (Mean Average Precision) measure and then a test is conducted for new data at different confidence levels. The training process depends on the quality and diversity of the images as well as the Annotation process for drilling. Therefore, the training process requires several attempts at the beginning to avoid some deep learning problems such as Overfitting and Underfitting, and making changes in important parameters such as learning rate, batch size and early stopping mechanism to obtain the best results.

Data collection

Data is collected from the Kartaview platform, taking into account the diversity of images in terms of lighting conditions, shapes of holes and their locations within the images.
Images are named based on track number, image number, street name, and annotation of holes and augmentation to increase the data size, then divided into a training set and verified. **Table 2** shows the details of the dataset for training processing.

Table 2 Details of the dataset for training processing

Data	Number of images
Training	491
Validation	122
Total	613

Processing of data

The benefit of this procedure is to increase the data size while giving more diversity and different conditions to the images is show in **Fig.**

5 Explain of Augmentations to improve the efficiency of the model for the training process. **Table 3** shows the details of Augmentation for the data set.

Table 3 Details of Augmentation for dataset

Augmentations	Description
90° Rotate	Clockwise, Counter-Clockwise, Upside Down
Crop	0% Minimum Zoom, 20% Maximum Zoom
Rotation	Between -2° and +2°
Noise	Up to 1.96% of pixels

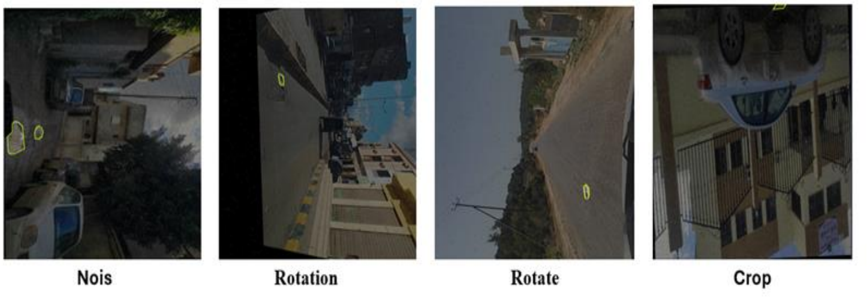

Fig. 2 Explain of Augmentations

Colab Notebook

Google offers the Colab Notebook work environment and provides multiple types of high-efficiency and high-speed graphics cards, as it relied on the **A 1000 GPU NVIDIA**, then added the Ultralytics library to train the YOLO model.

Model Training and Testing

The YOLOv10 model consists of three main parts: the backbone, the neck, and the head. The backbone captures characteristics from the input images, the neck functions and enhances those features, and the head generates the final boundary boxes and predicts object classes. The loss function measures how much the predicted boxes differ from the actual boxes. In YOLOv10x, the binary cross entropy focal loss, and GIoU loss are used together for this purpose. Tuning the hyperparameters, such as learning rate, batch size, and

تم استلام الورقة بتاريخ: 6 / 2024/10م وتم نشرها على الموقع بتاريخ: 30/ 2024/10م

number of epochs, is vital to make the model more effective is show **Table 4**. The model is trained and then tested on multiple images selected with different diversity and conditions to determine the training efficiency The model is trained to achieve the best accuracy of mAP 50% and then tested on multiple images selected with different diversity and conditions to determine the training efficiency.

Table 4 Parameter used in training YOLOv10x

Parameter	Type or Value
Model	YOLO v10x
Type of GPU	A 1000 GPU
Batch size	14
Image size	640 x 640
Learning rate	0.001
Optimizer	Adam optimizer
Loss function	focal and GIOU loss function

Results

The evaluation of results depends on objectness loss, box loss and classification loss during training and validation, along with the mean Average Precision (mAP) measured against epoch numbers **Fig. 6**. The goal is to achieve an mAP greater than 50% for reliable results.

Used precision and recall to assess the accuracy of the YOLOv10 model on the potholes images dataset. Recall calculates the proportion of true positive detections (TP) compared to the total number of actual positives ($TP + FN$), where FN represents false negatives. Recall is computed using the following formula:

$$Recall = TP + FN/TP$$

On the other hand, precision measures the proportion of TP against total predicted positives ($TP + FP$), with FP representing false positives. Precision is calculated as follows:

$$Precision = TP + FP/TP$$

mAP values across different recall levels. It is calculated using the area under the precision-recall curve, with the AUC averages for each class to obtain the final mAP.

تم استلام الورقة بتاريخ: 6 / 2024/10م وتم نشرها على الموقع بتاريخ: 30/ 2024/10م

$$mAP = \frac{1}{N} = \sum_{i=1}^{N} AP_i$$

N is the number of classes, and AP_i is the average precision for class i. **Table 5** shows the result mAP 50% obtained and the number of Epoch.

Table 5 Result of Precision, Recall and *mAP*

Metrics	Result
Epoch	230
Precision	0.76
Recall	0.28
mAP @ 0.5	0.53

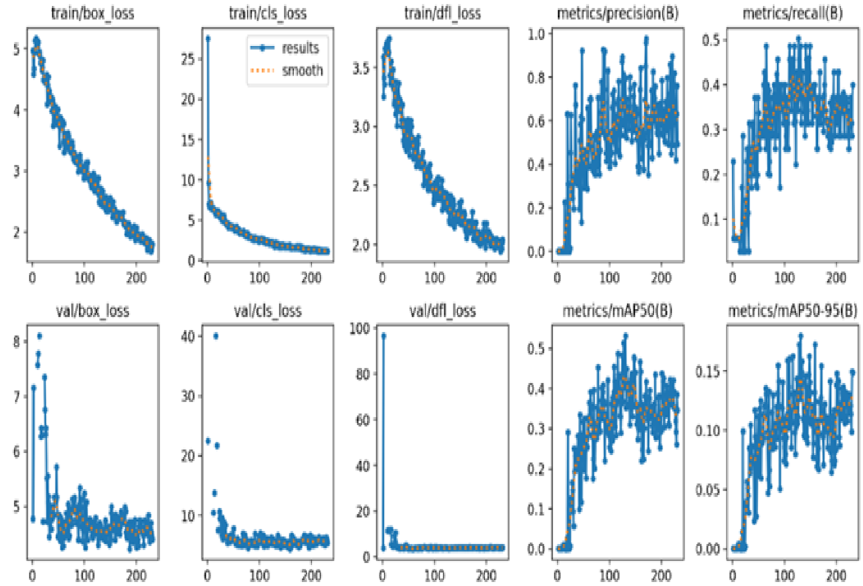

Fig. 6 Ploting of mAP, precision, recall measurements, and losses

After testing several images at different confidence levels, some images had holes detected and others did not. **Table 6** shows the number of detections and confidence levels. **Fig. 7** shows the images that were tested on at different Confidence.

Table 6 Number of detections and confidence levels

	Number of images	Number of deductions	True deductions
	1	1	1
	2	0	0
0.1	3	3	2
	4	2	2
	5	5	3
	6	6	6
	1	0	0
	2	0	0
0.3	3	3	2
	4	2	2
	5	2	2
	6	5	5
	1	0	0
	2	0	0
0.6	3	2	2
	4	0	0
	5	1	1
	6	3	3

Confidence 0.1 Confidence 0.3 Confidence 0.6

Fig. 5 Detections at different confidence levels

Discussion

Results of this study are showing the great promise that the YOLOv10 model holds in pothole detection, with *mAP* over 50%. This indicates good performance in pothole detection under

تم استلام الورقة بتاريخ: 6 / 2024/10م وتم نشرها على الموقع بتاريخ: 30/ 2024/10م

different image conditions and could be further improved upon. This set of Kartaview geospatial imagery and object detection capabilities within YOLOv10 provided the model with the ability to balance both detection and geolocation. The precision was 0.76, which indicates accuracy in detecting true pothole locations. However, the recall was relatively lower at 0.29, indicating that while the model is accurate in detecting potholes, some of them may be lost. This may be owing to a number of factors caused by the complexity of the diverse image backgrounds and smaller sizes of potholes. The analysis also indicated that confidence levels play an essential role in the performance of the model. More potholes were detected at the lower levels of confidence, regardless of false detections such as **Fig. 7**, while higher levels of confidence provided fewer but more accurate detections, at the risk of missing smaller ones. This shows the need for training with larger, better-quality data and to adjust the levels of confidence based on achieving the greatest detection and accuracy. The model was improved with data augmentation techniques applied, such as rotation and zoom. From this result, it is shown that image conditions and the size of the dataset are very critical to train the model effectively. This result proved the possibility of using open-source data and deep learning models for monitoring urban infrastructure.

Recommendations and future scope

- Increasing dataset, this could provide generally increased robustness of the YOLO model for generally changing environmental conditions, like lighting or seasonal changes, which might best generalize the model.
- Proposing an Automatic Road and Street Path Checking Mechanism and Defect Detection.
- Community sharing of data gathering Volunteers from groups like ODAT increase data entry speed and provide updated images of high quality to train the models more precisely.
- Collaboration with local governments to repair the detected road defects within the shortest time, scheduling/planning for maintenance accordingly.

تم استلام الورقة بتاريخ: 6 / 10/2024م وتم نشرها على الموقع بتاريخ: 30/ 10/2024م

Conclusion

This study explains how geographic coordinate-related data collection, open-source images of roads and streets, can be achieved with the help of deep learning techniques for road defects detection. The exploitation of Kartaview image data and the YOLOv10 model offers a scalable approach to the assessment of road damage that can aid in early intervention by reducing the cost of repair works while improving the safety of road users. The high accuracy of the YOLO model confirms its possibility for broader adoption in infrastructure management. Future research could focus on the integration of real-time detection ability and the adaptation of this scope to other cities facing similar infrastructure challenges.

References

[1] D. Associates, "Final Master Plan Report: Al-Bayda - Shahat, Benghazi District, Al-Jabal Al-Akhdar Municipality," Consultants in Development and Housing Affairs, Report No. 33, 1984.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 27-30 June 2016 2016, pp. 779-788.

[3] A. Paramarthalingam, J. Sivaraman, P. Theerthagiri, B. Vijayakumar, and V. Baskaran, "A deep learning model to assist visually impaired in pothole detection using computer vision," Decision Analytics Journal, vol. 12, p. 100507, 2024/09/01/ 2024.

[4] S. K. Satti, G. N. V. Rajareddy, K. Mishra, and A. H. Gandomi, "Potholes and traffic signs detection by classifier with vision transformers," Scientific Reports, vol. 14, no. 1, p. 2215, 2024/01/26 2024.

[5] Y. Safyari, M. Mahdianpari, and H. Shiri, "A Review of Vision-Based Pothole Detection Methods Using Computer Vision and Machine Learning," Sensors, vol. 24, no. 17.

[6] C. Ruseruka, J. Mwakalonge, G. Comert, S. Siuhi, F. Ngeni, and Q. Anderson, "Augmenting roadway safety with machine

تم استلام الورقة بتاريخ: 6 / 2024/10م وتم نشرها على الموقع بتاريخ: 30/ 2024/10م

learning and deep learning: Pothole detection and dimension estimation using in-vehicle technologies," Machine Learning with Applications, vol. 16, p. 100547, 2024/06/01/ 2024.

[7] Z. S. Mohd Shah and M. N. Haji Mohd, "Real-time Pothole Detection using Deep Learning," Evolution in Electrical and Electronic Engineering, vol. 5, no. 1, pp. 459-469, 04/21 2024.

[8] E. Ibragimov, Y. Kim, J. H. Lee, J. Cho, and J.-J. Lee, "Automated Pavement Condition Index Assessment with Deep Learning and Image Analysis: An End-to-End Approach," Sensors, vol. 24, no. 7.

[9] S. Yoon and J. Cho, "Convergence of Stereo Vision-Based Multimodal YOLOs for Faster Detection of Potholes," Computers, Materials and Continua, vol. 73, no. 2, pp. 2821-2834, 2022/06/14/ 2022.

[10] N. Ma et al., "Computer vision for road imaging and pothole detection: a state-of-the-art review of systems and algorithms," Transportation Safety and Environment, vol. 4, no. 4, p. tdac026, 2022.

[11] M. P. B and S. K.C, "Enhanced pothole detection system using YOLOX algorithm," Autonomous Intelligent Systems, vol. 2, no. 1, p. 22, 2022/08/31 2022.

[12] S.-S. Park, V.-T. Tran, and D.-E. Lee, "Application of Various YOLO Models for Computer Vision-Based Real-Time Pothole Detection," Applied Sciences, vol. 11, no. 23.

[13] Kartaview. "Kartaview: Explore the World with Open Street Level Imagery." https://kartaview.org/landing (accessed.

[14] THU-MIG. "YOLOv10: Object Detection Algorithm." https://github.com/THU-MIG/yolov10 (accessed.

[15] C.-Y. Wang and H.-y. Liao, YOLOv1 to YOLOv10: The fastest and most accurate real-time object detection systems. 2024.