

Smart Pothole Detection and Reporting System: A Comprehensive Review of Sensing Approaches, Deep-Learning Detectors, Empirical Evidence, and Reporting Challenges

Tambe Harshal Maruti, Prof.A.P. Banger, Prof. S. B. Bhosale, Prof. V. A. Karad

Department of Computer Engineering
Jaihind College of Engineering, Kuran, Maharashtra, India

Abstract- Potholes wear out vehicles, hold up traffic and cause crashes, yet manual road inspection is slow, expensive and cannot provide a continuous, geotagged record. This paper is a focused narrative review of five IEEE and IET studies on automated pothole detection: a comparison of YOLOv3, SSD, HOG with SVM and Faster R-CNN; a study that fine-tunes four deep detectors on road images from around the world and runs the chosen detector on a real vehicle that sends alerts to an Internet of Things platform; a convolutional-network prototype for autonomous vehicles; an assessment of R-CNN and SSD MobileNet through an object detection API; and a ResNet-50 system that detects potholes and estimates their depth. The review looks at three linked themes: the sensing approaches and detection models in use, how strong and how comparable the reported evidence is, and what matters when detection is stretched into a reporting pipeline. Only one of the studies could be read in full. It reports a mean average precision of 82.02% (IoU 0.4) for Faster R-CNN ResNet101 and 5–6 frames per second for a lighter model on an embedded automotive platform. The other four were judged from their published abstracts alone; among other figures, they give 99.02% accuracy for a prototype CNN and a maximum detection confidence of 93%. Because these numbers rest on different data and metrics, they cannot be ranked against one another. Just one study carries detections onward to a cloud platform. The review offers a comparative table, a conceptual architecture for the proposed pothole detection and reporting system (not experimentally validated) and a prioritised list of research gaps.

Keywords— Pothole detection, deep learning, object detection, YOLO, GPS, Internet of Things, road safety, automated reporting.

I. INTRODUCTION

Potholes are local failures of the road surface that damage vehicles, make driving uncomfortable and lead to accidents. Ping et al. define a pothole as structural damage that leaves a hollow in the road, one that can cause serious traffic accidents and lower road efficiency [1]. Yebes et al. rank it among the most irritating and costly road hazards to maintain [2]. In practice, inspection is still often carried out by maintenance crews who drive the network, stop at a number of places and report what they see [2]. Surveys of this kind are slow, expose staff to danger and leave no continuous, structured record of where the damage is.

This review grew out of a Smart Pothole Detection and Reporting System project. As described in the seminar presentation, the system relies on vehicle-mounted cameras or vibration sensors, AI/ML processing to spot potholes, GPS tagging of every detection, uploading of the location and an image to a cloud-based authority database, and an interactive web dashboard for municipal authorities. Its stated aims are to detect potholes automatically as vehicles drive, record exact GPS coordinates, send automated complaint reports to a central web server, reduce maintenance cost and make driving safer. The difficulties it lists include accurate detection at high speed, weak light at night or in heavy rain, dust, mud or water hiding potholes, and unreliable connectivity on highways. A technology proposed for this setting therefore has two jobs: to find the pothole and to

turn that finding into a record that a road authority can act on.

Five studies are reviewed here: a multi-model comparison [1], a worldwide image dataset with on-vehicle deployment [2], a CNN prototype for autonomous vehicles [3], an evaluation of an object detection API [4] and a ResNet-50 system with depth estimation [5]. Three questions steer the review. **RQ1:** Which sensing approaches and detection models are in use? **RQ2:** What performance and trade-offs do the studies report, and how dependable is that evidence? **RQ3:** What do the studies suggest for a detection-to-reporting pipeline? The paper closes with a comparative table, a conceptual architecture and a list of research gaps.

II. REVIEW METHODOLOGY

This is a narrative review of five papers provided for the study, not a systematic database search, so it does not claim exhaustive coverage. The set consists of one IEEE magazine article [2], three IEEE conference papers [1], [4], [5] and one article from the IET journal *Electronics Letters* [3]. For every paper the aim, method, data, reported results and stated limitations were pulled out and compared.

Access to the papers was uneven, and this is stated up front so that no claim is overstated. Paper [2] was read in full through the authors' public preprint, while its bibliographic details are taken from the IEEE-published version. Papers [1], [3], [4] and [5] were assessed only from their published abstracts and bibliographic records; where an abstract does not give an item such as dataset size, hyper-parameters or limitations, it is marked "not reported in the abstract" instead of being guessed.

All figures are those reported by the original authors. Studies cited inside [2] are described only as [2] describes them and were not read again. Since only [2] links detection to onward reporting, any conclusions about reporting are inferences and are labelled as such.

III. TECHNICAL FOUNDATIONS

1. Sensing modalities

Yebe et al. sort earlier pothole detectors by sensor type: ultrasonic, accelerometer, image, and combined image-and-depth approaches [2]. An accelerometer only registers a pothole when a wheel drives over it, which drivers often try to avoid; the suspension damps the vibration; phone-based versions depend on how the phone is mounted; and braking or other disturbances can be mistaken for potholes [2]. Image-based detection is closest to how a driver notices a hazard ahead, but classical image-processing methods leaned on hand-tuned filters that favoured particular datasets [2]. Depth and stereo methods add geometry, yet [2] describes them as sensitive to errors in disparity estimation, and some require cameras pointed down at the road. Together these observations favour a camera-first design with vibration sensing as an optional extra, as in the project scope; that design choice belongs to the project and is not a finding of the reviewed papers.

2. Deep-learning object detectors

All five studies rely on convolutional neural networks. Two-stage detectors such as Faster R-CNN first propose candidate regions and then classify and refine each of them [2]. Single-stage detectors such as SSD and YOLO predict boxes and classes in a single pass, which is usually quicker [1], [2]. HOG with SVM, tested in [1], serves as a classical baseline built on hand-crafted features. The detectors in [2] come from the TensorFlow model zoo, start from weights pre-trained on a large general dataset (COCO) and are fine-tuned for one pothole class [2]; [4] works through an object detection API. The difference between *classification* (does this image contain a pothole?) and *detection* (where is it?) matters for reporting. Study [5] fine-tunes ResNet-50, a classification backbone, on a classification dataset, whereas a reporting system also needs the pothole's position in the frame, which [2] supplies through bounding boxes.

3. Datasets and evaluation metrics

The studies measure success in different ways. Yebe et al. report precision, recall and mean average

precision (mAP) under both the PASCAL and COCO definitions and, because pothole outlines are hard to annotate, add a relaxed overlap threshold of IoU 0.4 next to the usual 0.5 [2]. Dewangan and Sahu give accuracy, sensitivity and F-measure [3]; Ping et al. compare accuracy and loss [1]; Javed et al. report a maximum detection confidence [4]. A detection confidence is simply the score a model attaches to a single detection; it says nothing about how many potholes were found or missed. The metrics are therefore not interchangeable.

IV. DETECTION MODELS AND SYSTEM ELEMENTS

1. Models compared in the reviewed papers

Ping et al. train and test four models, YOLOv3, SSD, HOG with SVM and Faster R-CNN, on labelled pothole and non-pothole images [1]. Yebes et al. fine-tune three Faster R-CNN variants (Inception v2, ResNet101 and Inception-ResNet v2 with atrous convolution) together with SSD MobileNet v2 [2]. Dewangan and Sahu propose a CNN-based prototype named PotNet, which uses a vision camera and also shows intelligent driving behaviour once a pothole is detected [3]. Javed et al. evaluate R-CNN and SSD MobileNet through an object detection API on test images and videos [4]. Singh et al. build a real-time detection and depth-estimation system in PyTorch around a fine-tuned ResNet-50 [5]. The abstract of [5] does not explain how depth is obtained.

2. Data and hardware

Only [2] describes its data and hardware in any detail. Its database contains 5,874 images (5,774 for fine-tuning and 100 for validation, with 9,716 and 171 annotated potholes respectively) from Europe, America, Asia and Africa: 4,030 windscreen-camera images from a public dataset, 1,644 frames from an automotive-grade front-bumper camera near Paris after fish-eye distortion was removed, and 100 street-view images. The images were resized to 1024×800 pixels, augmented through brightness changes and horizontal flips, and trained for two million steps on Tesla P100 GPUs. The test set holds 265 difficult images from routes near Paris (tunnels, bad weather, traffic jams, bumps, stains and

manholes), 482 dash-camera frames from an Oregon forest road and 54 re-annotated images from a public smartphone dataset [2]. The detectors were then run on an NVIDIA DrivePX2 embedded platform. For [1], [3], [4] and [5], the abstracts do not report dataset size, image source or hardware.

3. Cautions

Several cautions come up repeatedly in [2]. SSD MobileNet v2, chosen for its low cost, reached only 33.62% AP at IoU 0.5 and 45.57% at 0.4. The authors put this down to a design that analyses boxes separately, losing the surrounding road context that tells a pothole from ordinary pavement, and to its weakness on small boxes. The heavier Inception-ResNet v2 model scored lower than the simpler ResNet101 model, which the authors suspect reflects over-fitting on a small database. Manholes caused most of the false positives, and detections of cracks or unlabelled potholes on badly damaged roads counted as errors even though a person might accept them. Detection becomes difficult above 60 km/h because of image blur [2]. A municipal system should expect these failure modes and should filter or review such detections rather than send them automatically as repair orders.

V. EVIDENCE OF PERFORMANCE

1. Reported results

In [2], Faster R-CNN ResNet101 gave the best result, with precision of 83.48% and recall of 75.23% at IoU 0.5, and PASCAL mAP of 77.49% at IoU 0.5 and 82.02% at IoU 0.4. Faster R-CNN Inception v2 reached mAP of 67.05% and 75.45% at the same thresholds [2]. Ping et al. conclude that YOLOv3 was the best of their four models for faster and more reliable detection, although the abstract gives no figures [1]. Dewangan and Sahu report accuracy of 99.02%, sensitivity of 99.03% and F-measure of 98.33% for PotNet, which they describe as comparable with state-of-the-art techniques [3]. Javed et al. report successful detection, with a maximum confidence of 93%, for the R-CNN and SSD MobileNet comparison [4]. The abstract of [5] contains no performance figures.

2. Accuracy versus speed

Yebes et al. measured inference time per image: Inception v2, ResNet101 and Inception-ResNet v2 needed 53.2, 94.2 and 172.1 ms on a Tesla P100, and 177.1, 432.7 and 732.9 ms on the DrivePX2 [2]. Accuracy and delay rose roughly together, with the atrous model as the exception noted above. The authors accept about 5–6 frames per second for Inception v2, since detections are sent to an IoT platform for later warnings and are not used for immediate manoeuvres, and they point out that shrinking the input would raise false detections because potholes occupy only a small part of the image [2]. Ping et al. likewise balance speed against reliability when they prefer YOLOv3 [1]. The lesson for the proposed system is that accuracy, latency and target hardware should always be reported together.

3. Comparability and quality of the evidence

The 99.02% accuracy in [3] is higher than the 82.02% mAP in [2], but that does not show PotNet to be the better detector. The two studies differ in metrics, data and test conditions: [2] tested on deliberately difficult scenes and relaxed its overlap threshold, while the abstract of [3] describes validation of a prototype “in the prepared road environment”. Whether [3] used held-out images is not stated in its abstract. In the same way, a maximum confidence of 93% [4] is the best score for a single detection, not an accuracy. Only [2] reports test data from several cities, cameras and lighting conditions, so how well the other four results generalise is unknown.

VI. FROM DETECTION TO REPORTING: THE VEHICLE-DEPLOYED STUDY

Of the five papers, [2] is the only one that carries detection all the way to a reporting channel. The work was done within the AUTOPILOT H2020 Highway Pilot use case, in which a vehicle acts as an IoT device: in-vehicle AI modules produce low-bandwidth messages that are sent wirelessly to an IoT platform, and a cloud service merges measurements from several devices to locate and characterise road hazards [2]. The set-up consisted of an automotive-grade camera on the front bumper, a GPU computer and IoT communication

modules, and the detector ran on a real vehicle at speeds below 60 km/h [2].

1. Findings

Detected potholes are monitored from control centres and broadcast as warnings to other road users once enough confidence has built up after several repeated detections at the same map location [2]. The paper also states that automated hazard detection makes it possible to report to traffic or infrastructure authorities for maintenance [2]. Among the five papers, this repeated-detection rule is the closest published counterpart to the duplicate filtering and confidence checking proposed for this project. Paper [2] presents it as a design principle; it does not say how many repeats are needed or how well the rule performs.

2. Design issues

The authors propose filtering out detections beyond the road boundary using lane or road detectors, plan lightweight or pruned networks to speed up inference and a larger dataset to raise detection rates, and name YOLOv3 as a candidate for future study [2]. They note that the frame rate is acceptable for notification but not for reactive vehicle manoeuvres [2].

3. Limits and transferability

The deployment in [2] aims at warning following vehicles as well as reporting to authorities, whereas the present project aims at a municipal dashboard with human verification. The paper reports no GPS error, upload reliability, handling of network loss or authority response time, and the abstracts of [1], [3], [4] and [5] report none either. Whether the detection results carry over to the camera positions, speeds and road types of buses or garbage trucks, which the seminar names as future deployments, has not been tested.

VII. COMPARATIVE ANALYSIS

Table 1 summarises the five reviewed studies. The findings cannot be compared directly because data, hardware and metrics differ. Entries marked “abstract only” are drawn from the published abstract.

Table 1. Comparison of the five reviewed studies

Study	Objective and design / data	Main findings and limitations
Ping et al. (2020) [1]	<i>Aim:</i> to compare detectors for street potholes <i>Design:</i> YOLOv3, SSD, HOG+SVM and Faster R-CNN trained on labelled pothole and non-pothole images (abstract only)	<i>Findings:</i> YOLOv3 judged best for faster, more reliable detection <i>Limits:</i> abstract gives no figures, dataset size or hardware
Yebe et al. (2021) [2]	<i>Aim:</i> detecting potholes in worldwide road scenes and reporting them to an IoT platform <i>Design:</i> 5,874 training and validation images; 4 DNNs fine-tuned; tested on DrivePX2 and on a vehicle (full text)	<i>Findings:</i> ResNet101 mAP 77.49% (IoU 0.5) and 82.02% (IoU 0.4); Inception v2 about 5–6 fps <i>Limits:</i> weak SSD MobileNet v2; manhole false positives; blur above 60 km/h; not real-time
Dewangan & Sahu (2021) [3]	<i>Aim:</i> pothole detection for autonomous vehicles <i>Design:</i> CNN with a vision camera in a prototype (PotNet), validated in a prepared road environment (abstract only)	<i>Findings:</i> accuracy 99.02%, sensitivity 99.03%, F-measure 98.33% <i>Limits:</i> dataset and test split missing from abstract; prepared environment
Javed et al. (2021) [4]	<i>Aim:</i> pothole detection from images and videos <i>Design:</i> R-CNN and SSD MobileNet through an object detection API (abstract only)	<i>Findings:</i> potholes found in test images; maximum confidence 93% <i>Limits:</i> confidence is not accuracy; abstract lacks recall and dataset details
Singh et al. (2024) [5]	<i>Aim:</i> real-time detection with depth estimation <i>Design:</i> PyTorch; ResNet-50 fine-tuned on a pothole classification dataset (abstract only)	<i>Findings:</i> none stated in the abstract consulted <i>Limits:</i> depth method and accuracy not stated; classification does not localise

VIII. PROPOSED CONCEPTUAL ARCHITECTURE

Figure 1 brings together the design implications of the reviewed studies as a staged architecture for the proposed pothole detection and reporting system. It is a conceptual proposal derived from the literature and has not been tested experimentally.

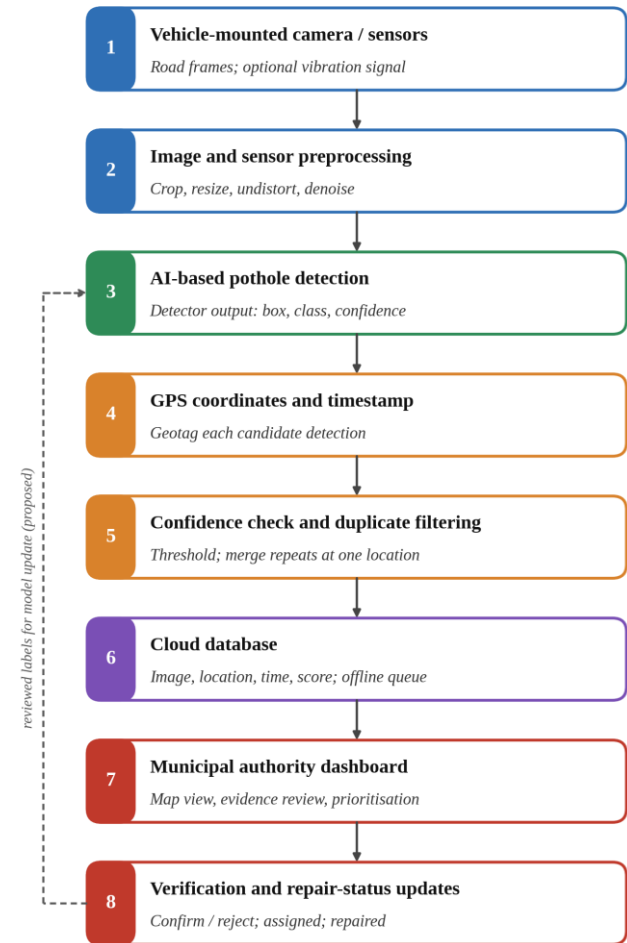


Fig. 1. Proposed conceptual architecture for the smart pothole detection and reporting system (conceptual, not experimentally validated)

Stages supported by the reviewed papers. Stage 1 captures road frames with a vehicle-mounted camera, as in [2], and an optional vibration sensor taken from the project scope. Stage 2 prepares the data; [2] removed fish-eye distortion, cropped borders and resized images to 1024×800 pixels. Stage 3 runs the detector. Faster R-CNN ResNet101 gave the best accuracy in [2], whereas YOLOv3 was

preferred for speed and reliability in [1], so the choice should be tested on the target hardware. A classification-only model, as in [5], does not locate the pothole in the frame. Stages 4 and 6 follow the IoT reporting of [2], where detections are placed on a map and sent to a cloud service.

Stages that are the author's own design recommendations. None of the five studies evaluates stages 5, 7 or 8. Stage 5 adds a confidence threshold and the merging of repeated detections at one location, motivated by the repeated-detection rule in [2]. Stage 6 should also queue records locally when the connection drops, because the project names intermittent connectivity as a challenge. Stage 7 shows the evidence, location and score to authority staff so that a person can review uncertain cases. Stage 8 records verification and repair status, and the dashed loop feeds reviewed labels back for model updating. Manholes, patches and shadows are the likely sources of false reports [2], which is why stage 7 keeps a human in the loop.

Research Gaps and Future Directions

Gaps supported by the reviewed papers. (i) Generalisation: only [2] tests across several cities, cameras and lighting conditions, and its authors suspect over-fitting caused by a limited dataset. (ii) Speed on embedded hardware: [2] names runtime as a limitation and plans lighter networks and optimisation. (iii) False positives: [2] reports manholes and ambiguous pavement damage, and annotation ambiguity led its authors to relax the IoU threshold. (iv) High-speed driving: blur above 60 km/h limits detection [2]. (v) Inconsistent evaluation: the five papers use different metrics, so their results cannot be ranked [1]–[5]. (vi) The reporting loop: only [2] reaches an IoT platform, and it reports no location error, upload reliability or repair closure (the abstracts of the other four mention none). (vii) Severity: [5] aims at depth estimation but its abstract gives no accuracy, and the other studies detect presence only.

Design recommendations (the author's own, not findings of the papers). Test under day, night, rain and wet-surface conditions against ground-truth pothole locations; measure GPS error and the

duplicate-report rate; keep offline queueing and delayed synchronisation; blur or restrict access to faces and number plates in stored frames; add depth or area estimation before any automatic severity ranking; and keep a human verification step before every repair order. The seminar's future scope, such as cameras on public buses and garbage trucks, AI priority-based tickets and crowdsourced citizen reports, depends on these tests.

Directions requiring investigation. Priorities include a small pilot with campus or municipal maintenance staff over repeated road trips; separate reporting of model metrics, location error, end-to-end latency, upload success and staff response time; head-to-head tests of a lightweight YOLO-family detector and Faster R-CNN on the same data and device; and a study of whether the repeated-detection rule in [2] reduces false reports without delaying genuine ones.

IX. CONCLUSION

Across the five reviewed studies, deep-learning detectors can find potholes in road images with encouraging accuracy. The best-documented result, from [2], shows 82.02% mAP at a relaxed IoU of 0.4 for a heavier detector and about 5–6 frames per second for a lighter one on embedded hardware, while [3] and [4] report different figures on different data. The evidence is nevertheless uneven: only one paper was read in full, the metrics differ, and only [2] extends detection to an IoT platform. For the proposed pothole detection and reporting system, this points to a modular design in which the detector can be swapped independently of geotagging, duplicate filtering, cloud storage and the authority dashboard, with human verification of uncertain detections. The system can be called dependable only after evaluation on real roads in varied conditions, with location error and upload reliability measured.

REFERENCES

1. P. Ping, X. Yang, and Z. Gao, "A deep learning approach for street pothole detection," in *Proc.*

- 2020 IEEE 6th Int. Conf. Big Data Comput. Service Appl. (BigDataService)*, Oxford, U.K., Aug. 2020, pp. 198–204, doi: 10.1109/BigDataService49289.2020.00039.
2. J. J. Yebes, D. Montero, and I. Arriola, "Learning to automatically catch potholes in worldwide road scene images," *IEEE Intell. Transp. Syst. Mag.*, vol. 13, no. 3, pp. 192–205, 2021, doi: 10.1109/MITS.2019.2926370.
 3. D. K. Dewangan and S. P. Sahu, "PotNet: Pothole detection for autonomous vehicle system using convolutional neural network," *Electron. Lett.*, vol. 57, no. 2, pp. 53–56, Jan. 2021, doi: 10.1049/ell2.12062.
 4. A. Javed, M. S. Mahmud, M. T. Alam, M. F. B. Ohab, K. R. Ali, A. A. Jobaer, and M. Uddin, "Pothole detection system using region-based convolutional neural network," in *Proc. 2021 IEEE 4th Int. Conf. Comput. Commun. Eng. Technol. (CCET)*, Beijing, China, Aug. 2021, doi: 10.1109/CCET52649.2021.9544396.
 5. A. Singh, M. Randhawa, N. Kalra, C. Seth, A. Gill, P. Chopra, and G. Kaur, "Pothole sensing and depth estimation system using deep learning technique," in *Proc. 2024 OPJU Int. Technol. Conf. (OTCON) Smart Comput. Innov. Adv. Ind. 4.0*, Jun. 2024, pp. 1–4, doi: 10.1109/OTCON60325.2024.10687877.