The Scientific Temper
Vol. 17, No. 11, November 2026: pp. 7221-7227
ISSN: 0976-8653, E-ISSN: 2231-6396
DOI: 10.58414/SCIENTIFICTEMPER.2026.17.11.2592
A Web Of Science Journal

RESEARCH ARTICLE

Head-Only Training and End-to-End Fine-Tuning of YOLOv5n and YOLOv5s for Pothole Detection on the RDD2022 India Subset

Prajina V K

Department of Computer Science, Krupanidhi Degree College, Bengaluru, Karnataka, India

Corresponding author: *prajinavk@gmail.com*

ABSTRACT

This study compared head-only training and end-to-end fine-tuning of YOLOv5n and YOLOv5s for pothole detection on the India subset of the Road Damage Dataset 2022 (RDD2022). The 7,705 eligible images were divided into training (5,382), validation (1,161) and internal test (1,162) sets. Each of the four configurations was trained once for 12 epochs using 256 × 256-pixel inputs. End-to-end fine-tuning increased test average precision at an intersection over union of 0.50 (AP@0.50) from 1.36% to 6.84% for YOLOv5n and from 1.45% to 3.22% for YOLOv5s. At confidence thresholds selected separately on the validation set, the number of annotated potholes detected increased from 33 to 87 of 478 for YOLOv5n and from 13 to 53 for YOLOv5s. False-positive detections decreased from 1,206 to 502 for YOLOv5n and increased from 278 to 623 for YOLOv5s. Fine-tuned YOLOv5n achieved the highest observed recall (18.20%) and detected 41 of 321 small potholes. Both models benefited from end-to-end fine-tuning under this protocol, although most annotated potholes remained undetected.

Keywords: *Pothole detection, Object detection, RDD2022, YOLOv5, Transfer learning, Road imagery*

INTRODUCTION

Road images can aid in making pavement inspections by providing information about the location and type of damage observed on the surface. Maeda et al. (2018) demonstrated the feasibility of training convolutional object detectors on bounding boxes annotated in images captured with vehicle-mounted smartphones. The Road Damage Dataset 2022 (RDD2022) extended this approach to 47,420 road images from six countries, including India, and identified potholes as a separate damage class (D40) alongside three crack classes (Arya et al., 2024). An annotated benchmark can be used to evaluate a detector for its localisation ability within an image, which is relevant to screening road sections for inspection.

Road-damage detectors may yield different results if used on images from a new region. Arya et al. (2021) studied the transfer of a detector trained on Japanese images to Indian and Czech data and evaluated models trained with data from multiple countries. The subsequent Crowdsensing-based Road Damage Detection Challenge (CRDDC'2022) evaluated submissions on separate country and combined leaderboards (Arya et al., 2022). This distinction matters for Indian road imagery. A score combining all countries

Received: 5th November, 2026 **Accepted: 7th November, 2026** **Published: 28th November, 2026**

and damage types hides the performance of the class of pothole in one country.

Several studies have examined computational cost alongside detection performance. B and K.C. (2022) evaluated an enhanced YOLOX model using a public pothole dataset. Wan et al. (2022) modified YOLOv5s to reduce model size and improve speed while detecting four types of road damage on an expanded RDD2020 dataset. Babawarun et al. (2025) studied YOLOv8s using RDD2022 India and a South African road dataset. Their within-country and cross-country experiments illustrate how a detector's reported performance depends on the images, classes and protocol used for evaluation.

Image-only bounding-box detection is one of several ways to locate pavement defects. Fan et al. (2020) used stereo disparity and a fitted road surface to identify depressions from road geometry. The current study uses monocular still images and assesses localisation against annotated boxes. Different measurements would need to be used for depth and physical severity.

Potholes can occupy only a small part of a forward-facing road image. In a complex visual scene, a high-resolution dataset was created with small and very small pothole annotations by Chen et al. (2026). In general, Lin, Dollár, et al. (2017) demonstrated that multi-scale feature fusion can boost object detection. These studies inspire reporting the recall by box size in addition to the overall average precision. Higher input resolution may improve the visibility of small potholes, although it may also increase inference time.

This study examined whether allowing the pretrained backbone and neck to adapt to Indian road imagery improves pothole detection at a fixed input resolution. Head-only training and end-to-end fine-tuning were compared for YOLOv5n and YOLOv5s (Ultralytics, 2022) using the same RDD2022 India splits, 256 × 256-pixel inputs and 12-epoch training schedule. The primary comparison was the change in AP@0.50 within each architecture. Precision, recall, false-positive detections and recall by annotated pothole size were also examined.

METHODS

Study design and dataset

This study compared YOLOv5n and YOLOv5s for single-class pothole detection in still road images. The India subset of the RDD2022 dataset (Arya et al., 2024) provided 9,665 images, of which 7,706 had XML annotations. The 1,959 challenge test images lacked public annotations and were excluded. One additional image containing an unrecognised class code was set aside. D40 was mapped as class 0, and the bounding boxes were also normalised to YOLO format. Images whose annotation files contained no D40 bounding boxes were retained as background examples for pothole detection.

The 7,705 retained images were assigned to fixed training, validation and internal test splits (Table 1). Consecutive blocks of 50 image identifiers were grouped, and strongly similar image pairs were kept together using a perceptual hash Hamming distance of ≤6 and an RGB thumbnail root mean square error of ≤10. Groups were assigned using seed 20261112. The source lacked verified route identifiers, so this procedure could reduce image overlap while leaving possible overlap between road scenes.

Table 1. Dataset composition by split

Split	Images	Images with D40	Images without D40	D40 boxes
Training	5,382	1,069	4,313	2,233
Validation	1,161	229	932	476
Test	1,162	232	930	478

Model training

Both models were initialised from YOLOv5 v7.0 checkpoints pretrained on the Common Objects in Context dataset (COCO; Lin et al., 2014) using the official Ultralytics (2022) release. The original 720 × 720 images were downsampled to 256 × 256 pixels by area interpolation. Each architecture was evaluated under two training strategies. For head-only training, the backbone and neck were frozen. Their outputs were cached in 16-bit floating point and converted to 32-bit floating point to train the three output convolutions. For end-to-end fine-tuning, the resized images were passed through the full network in 32-bit floating point, and the backbone, neck and detection head were updated jointly. Both strategies started from the corresponding COCO-pretrained checkpoint, reused the pretrained box and objectness weights, and used a newly initialised single-class output. Both used the same default anchors and YOLOv5 box and objectness losses.

Each of the four configurations was trained once for 12 epochs using random seed 42, AdamW, a batch size of 16, weight decay of 0.0005 and a cosine learning-rate schedule from 0.002 to 0.0002. The runs used the same training split and image order, without image

augmentation. The final-epoch checkpoint from each run was evaluated. The software environment used Python 3.12, PyTorch 2.5.1 and torchvision 0.20.1.

Evaluation

For each of the four configurations, the confidence threshold that maximised F1 on the validation split was fixed before test evaluation. Precision, recall and F1 used one-to-one matching of detections to reference boxes in descending confidence order at intersection over union (IoU) ≥0.50. Average precision (AP) was calculated separately from detections with confidence ≥0.001, non-maximum suppression IoU of 0.60 and a limit of 300 detections per image. AP@0.50 used an IoU threshold of 0.50, and AP@0.50–0.95 averaged AP across ten thresholds in 0.05 increments. The descriptive precision–recall curves use the one-to-one matching rule at IoU ≥ 0.50 and sweep the retained confidence scores.

Converted YOLO annotations were transformed back to original-image pixel coordinates and compared with the source D40 XML annotations for all 7,705 retained images. The matching procedure was checked using synthetic examples with known detection outcomes. For the two head-only runs, test-set detection counts were independently recalculated from the saved predictions and original XML annotations using the validation-selected thresholds and an IoU threshold of 0.50.

Recall was also calculated by annotated box area at the 256 × 256 input scale. Small boxes had areas below 32^2 pixels, medium boxes ranged from 32^2 to below 96^2 pixels, and large boxes had areas of at least 96^2 pixels.

For the original head-only benchmark, mean forward-pass time per image was measured on an Intel Xeon Platinum 8573C CPU using six PyTorch threads, 32-bit inference and batches of eight. Timing excluded image loading, preprocessing and non-maximum suppression. Model comparisons were descriptive.

RESULTS

Detection performance on the held-out test set

The internal test set contained 1,162 images and 478 annotated potholes. End-to-end fine-tuning increased AP@0.50 by 5.47 percentage points for YOLOv5n and 1.77 percentage points for YOLOv5s (Table 2). At the respective validation-selected thresholds, fine-tuned YOLOv5n detected 54 additional potholes and produced 704 fewer false-positive detections than its head-only baseline. Fine-tuned YOLOv5s detected 40 additional potholes and produced 345 more false-positive detections. Figures 1 and 2 show the precision–recall curves and detection counts for the original head-only runs.

Table 2. Detection performance and CPU forward-pass time on the internal test set.

Model	Training strategy	Confidence threshold	Precision (%)	Recall (%)	F1 (%)	AP@0.50 (%)	AP@0.50–0.95 (%)
YOLOv5n	Head only	0.030	2.66	6.90	3.84	1.36	0.38
YOLOv5n	End-to-end	0.105	14.77	18.20	16.31	6.84	1.85
YOLOv5s	Head only	0.090	4.47	2.72	3.38	1.45	0.35
YOLOv5s	End-to-end	0.055	7.84	11.09	9.19	3.22	0.87

Note. Precision, recall and F1 are reported at the validation-selected threshold for each configuration. AP was calculated separately across confidence scores.

All 3,187 converted pothole boxes agreed with the source annotations to within 0.001 pixels. Independent matching against the original test annotations reproduced the true-positive, false-positive and false-negative counts for both head-only runs. In the original head-only CPU benchmark, mean forward-pass time was 9.56 ms per image for YOLOv5n and 17.21 ms per image for YOLOv5s at a batch size of eight.

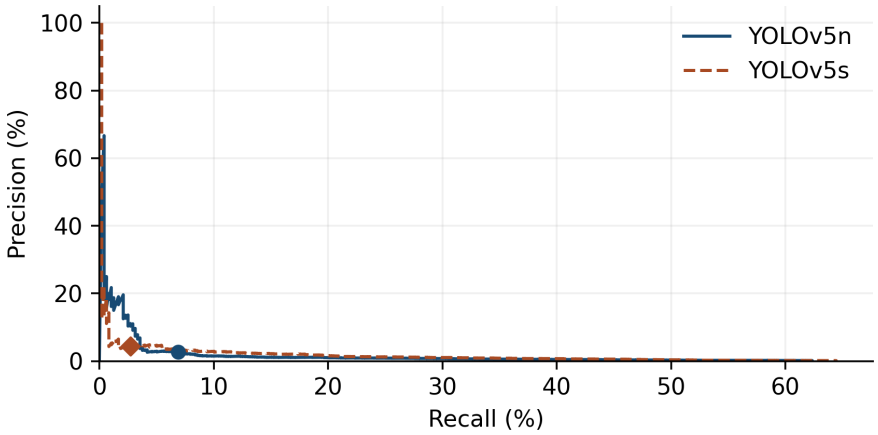

Figure 1. Precision–recall curves for head-only YOLOv5n and YOLOv5s on the internal test set

Note. Matching used IoU ≥ 0.50. Markers indicate the test operating points at the validation-selected confidence thresholds of 0.030 for YOLOv5n and 0.090 for YOLOv5s.

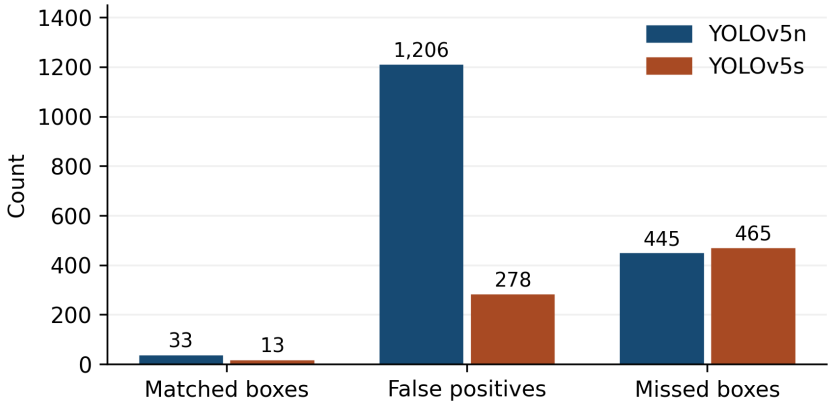

Figure 2. Detection outcomes for head-only YOLOv5n and YOLOv5s at their validation-selected confidence thresholds

Note. The 478 reference boxes were used for evaluation of both models. A false positive is a detection that did not match with a reference box according to the evaluation rule.

Detection by annotated box size

Small potholes accounted for 321 of 478 test annotations (67.2%). End-to-end fine-tuning increased small-pothole detections from 13 to 41 for YOLOv5n and from 5 to 23 for YOLOv5s. For the 141 medium potholes, detections increased from 12 to 37 and from 5 to 24, respectively. For the 16 large potholes, detections increased from 8 to 9 and from 3 to 6, respectively. Figure 3 shows recall within each size category.

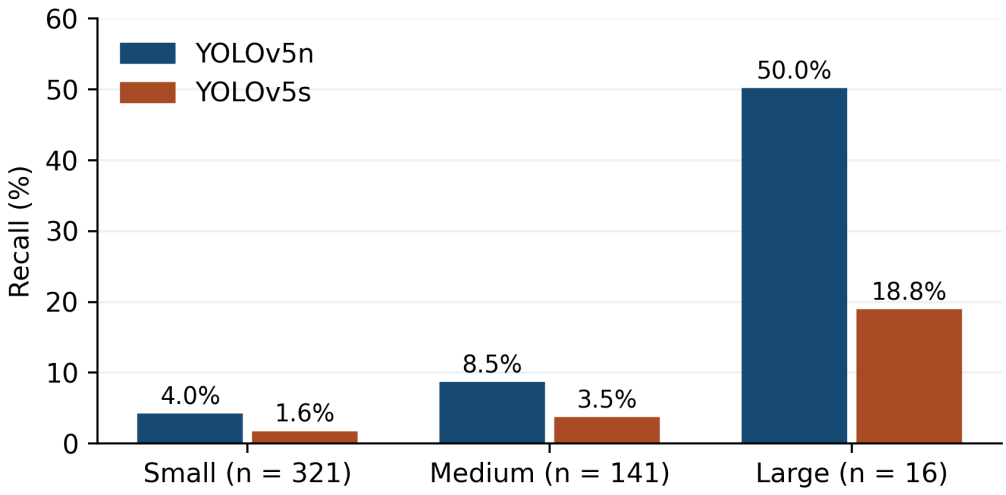

Figure 3. Pothole recall by annotated box size, model and training strategy

Note: Size categories were defined by bounding-box area at the 256 × 256 input scale. The large-box category contained 16 annotations.

DISCUSSION

Detection performance and operating thresholds

The effect of training strategy was larger than the difference between the two head-only models. Fine-tuning increased AP@0.50 from 1.36% to 6.84% for YOLOv5n and from 1.45% to 3.22% for YOLOv5s. At the validation-selected thresholds, YOLOv5n gained recall while producing fewer false positives, whereas YOLOv5s gained recall while producing more false positives. The threshold-specific counts describe the operating point selected for each configuration. Each configuration was trained once, so the variability of these differences across runs remains unknown.

The detection counts quantify errors against the reference boxes, but they do not identify the source of each error. For example, a predicted box can fail the matching criterion because it is poorly aligned, duplicates another prediction or lies on the background. An error taxonomy such as TIDE could help distinguish these patterns in the saved detections (Bolya et al., 2020). Such an audit would show whether subsequent work should prioritise localisation, background rejection or recovery of missed objects.

Comparison with prior road damage studies

The strongest CRDDC'2022 submission reported an F1 score of approximately 76% on the combined six-country test set using an ensemble of YOLO and Faster R-CNN models (Arya et al., 2022). For YOLOv5n and YOLOv5s, respectively, F1 scores were 3.84% and 3.38% after head-only training and 16.31% and 9.19% after end-to-end fine-tuning. The challenge score reflects several damage classes, different test images and an ensemble, whereas this study evaluated individual detectors on D40 alone using an internal India split. The low recall and precision establish weak performance on its own test set. The difference between the reported F1 scores cannot isolate the effect of architecture or training. Babawarun et al. (2025) also reported an AP@0.50 of 73.2% for their 'pothole/separation/rutting' category. Differences in category definition, data split and training protocol limit direct comparison with the present D40-only result.

Babawarun et al. (2025) provide a more detailed geo-comparison. YOLOv8s achieved a reported mAP@0.50 of 77.6% across four remaining damage classes after retraining, whereas they achieved 15.6% in their 6-class configuration on their RDD2022 India split. Both figures are average (mean) scores by class. The present AP values are for a different split, detector and training schedule and only apply to D40. A comparison of their two configurations will demonstrate the need for class definitions and training design when interpreting scores even within the same source data set. Similarly, the YOLOX study used a separate public pothole dataset (B & K.C., 2022), while YOLO-LRDD changed the backbone, feature fusion and loss function and used an expanded RDD2020 dataset at 640 × 640 (Wan et al., 2022). Neither study isolates the effect of freezing YOLOv5 features on the current split.

Small potholes and possible causes of missed detections

Small boxes accounted for 321 of 478 test annotations (67.2%). Head-only YOLOv5n and YOLOv5s detected 13 and 5 small potholes, respectively. End-to-end fine-tuning increased these counts to 41 and 23, although both

models still missed most small potholes. Both models also had higher recall for the 16 large boxes, although the small denominator limits interpretation of that stratum. The pattern is consistent with the challenge of representing small road defects described by Chen et al. (2026) and with the importance of feature scale shown by Lin, Dollár, et al. (2017). Downsampling from 720 × 720 to 256 × 256 may have removed useful detail. In addition, Yosinski et al. (2014) found that transferability can fall when pretrained features are specialised to a different task. The higher small-pothole recall observed after fine-tuning is consistent with a benefit from adapting pretrained features to pavement imagery. Input resolution was fixed at 256 × 256 pixels, so the effect of downsampling could not be estimated separately.

Of the 5,382 training images, 4,313 (80.1%) contained no D40 annotations. This proportion describes the image set and does not quantify the imbalance between foreground and background candidate locations. Easy background examples can dominate the aggregate loss in dense detection (Lin, Goyal, et al., 2017). Future experiments could evaluate positive-image sampling or loss weighting while retaining the same validation protocol. Higher-resolution inputs and tiled inference also warrant investigation for small potholes, with their additional computational cost measured explicitly (Akyon et al., 2022).

Generalisability and practical implications

The previous studies on cross-country transfer (Arya et al., 2021; Babawarun et al., 2025) inspire testing on routes and recording conditions that are different from the training data. Grouping nearby image identifiers and visually similar frames was intended to reduce overlap across splits. Verified route identifiers were unavailable, so some overlap between road scenes remains possible. The internal test set can then only offer limited evidence of performance when measured on roads that are recorded separately. Repeated training runs using different random seeds would assess run-to-run variability. Evaluation on images from independently recorded routes would assess geographic generalisation.

In the original head-only CPU benchmark, YOLOv5n had a shorter mean forward-pass time than YOLOv5s (9.56 versus 17.21 ms per image). Maeda et al. (2018) compared the runtime of a GPU server and a smartphone, while Wan et al. (2022) compared the single image recognition time under their setup. There are variations in hardware, image size, and image timing limits between these studies. The current times do not include loading, preprocessing and suppression of overlapping predictions. Batch-one video test, end-to-end latency, false alerts per distance travelled, and warning lead time are all required for a driver-warning application. The current results are most useful to provide a baseline for this development work, given the high miss counts.

CONCLUSION

On the internal RDD2022 India test set, end-to-end fine-tuning at 256 × 256 pixels increased AP@0.50 from 1.36% to 6.84% for YOLOv5n and from 1.45% to 3.22% for YOLOv5s. Fine-tuned YOLOv5n achieved the highest observed recall, detecting 87 of 478 annotated potholes, including 41 of 321 small potholes. At its validation-selected threshold, it also produced fewer false-positive detections than its head-only baseline. Nevertheless, it missed 391 annotated potholes. Repeated training across random seeds and evaluation on independently recorded routes are needed to assess the reproducibility and generalisability of these findings.

On the internal RDD2022 India test set, end-to-end fine-tuning at 256 × 256 pixels improved average precision and small-pothole detection for both YOLOv5 variants under the 12-epoch training protocol. YOLOv5n showed the larger observed gain and detected 41 of 321 small potholes, compared with 13 under head-only training. Overall recall remained low. Each configuration was trained with one random seed, so the reproducibility of these gains across runs remains to be established.

Acknowledgement

The author thanks the Management of Krupanidhi Group of Institutions for support through the Krupanidhi Research Incubation Centre (K-RIC), and the Research Mentors for their guidance.

Conflict of interest: None

Source of support: None

REFERENCES

Akyon, F. C., Altinuc, S. O., & Temizel, A. (2022). Slicing aided hyper inference and fine-tuning for small object detection. In *2022 IEEE International Conference on Image Processing (ICIP)* (pp. 966–970). IEEE. https://doi.org/10.1109/ICIP46576.2022.9897990

Arya, D., Maeda, H., Ghosh, S. K., Toshniwal, D., Mraz, A., Kashiyama, T., & Sekimoto, Y. (2021). Deep learning-based road damage detection and classification for multiple countries. *Automation in Construction, 132*, Article 103935. https://doi.org/10.1016/j.autcon.2021.103935

Arya, D., Maeda, H., Ghosh, S. K., Toshniwal, D., Omata, H., Kashiyama, T., & Sekimoto, Y. (2022). Crowdsensing-based road damage detection challenge (CRDDC'2022). In *2022 IEEE International Conference on Big Data (Big Data)*. IEEE. https://doi.org/10.1109/BigData55660.2022.10021040

Arya, D., Maeda, H., Ghosh, S. K., Toshniwal, D., & Sekimoto, Y. (2024). RDD2022: A multi-national image dataset for automatic road damage detection. *Geoscience Data Journal, 11*(4), 846–862. https://doi.org/10.1002/gdj3.260

B, M. P., & K.C., S. (2022). Enhanced pothole detection system using YOLOX algorithm. *Autonomous Intelligent Systems, 2*, Article 22. https://doi.org/10.1007/s43684-022-00037-z

Babawarun, T., Pandelani, T., & Ngwangwa, H. M. (2025). Investigation of the transferability of measured data for application of YOLOv8s in the identification of road defects: An SA-Indian case study. *Sustainability, 17*(23), Article 10641. https://doi.org/10.3390/su172310641

Bolya, D., Foley, S., Hays, J., & Hoffman, J. (2020). TIDE: A general toolbox for identifying object detection errors. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), *Computer vision – ECCV 2020* (pp. 558–573). Springer. https://doi.org/10.1007/978-3-030-58580-8_33

Chen, H., Tu, Z., Zhao, Y., & Ye, J. (2026). A high-resolution perspective-view road image dataset for pothole detection. *Scientific Data, 13*, Article 961. https://doi.org/10.1038/s41597-026-07317-w

Fan, R., Ozgunalp, U., Hosking, B., Liu, M., & Pitas, I. (2020). Pothole detection based on disparity transformation and road surface modeling. *IEEE Transactions on Image Processing, 29*, 897–908. https://doi.org/10.1109/TIP.2019.2933750

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 936–944). IEEE. https://doi.org/10.1109/CVPR.2017.106

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)* (pp. 2999–3007). IEEE. https://doi.org/10.1109/ICCV.2017.324

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Computer vision – ECCV 2014* (pp. 740–755). Springer. https://doi.org/10.1007/978-3-319-10602-1_48

Maeda, H., Sekimoto, Y., Seto, T., Kashiyama, T., & Omata, H. (2018). Road damage detection and classification using deep neural networks with smartphone images. *Computer-Aided Civil and Infrastructure Engineering, 33*(12), 1127–1141. https://doi.org/10.1111/mice.12387

Ultralytics. (2022). *YOLOv5* (Version 7.0) [Computer software]. GitHub. https://github.com/ultralytics/yolov5/releases/tag/v7.0

Wan, F., Sun, C., He, H., Lei, G., Xu, L., & Xiao, T. (2022). YOLO-LRDD: A lightweight method for road damage detection based on improved YOLOv5s. *EURASIP Journal on Advances in Signal Processing, 2022*, Article 98. https://doi.org/10.1186/s13634-022-00931-x

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems 27* (pp. 3320–3328). Curran Associates. https://proceedings.neurips.cc/paper_files/paper/2014/file/532a2f85b6977104bc93f8580abbb330-Paper.pdf