[1.]Ishak KAZIC, [1.]Becir ISAKOVIC

DEVELOPMENT OF AN APPLICATION FOR DATA COLLECTION AND ANALYSIS FOR EMERGENCY RESPONSE IN ECOLOGICAL CRISIS AREAS OF BOSNIA AND HERZEGOVINA

[1.]Dept. of Information Technologies, Faculty of Engineering, Natural & Medical Sciences, International Burch University, Sarajevo, BOSNIA & HERZEGOVINA

Abstract: Inadequate maintenance of garbage disposal sites and containers is one of the biggest challenges facing Bosnia and Herzegovina. The problem of overcrowded containers is a problem that happens year after year, and over time it seems that there is simply no solution. The traditional method, which involves the use of a fixed garbage collection schedule, is no longer efficient. In this, one of the most pressing problems arises, which implies inadequate management of the system that is in the service of controlling it. Overcrowded containers are an excellent place for rodents and therefore represent a source of a large number of infections, this fact is one of the major environmental problems. Therein lies the inspiration of this paper, which deals with researching the implementation of automation in conditions aimed at preventing environmental disasters, thus providing better analytics and insight into the state of the field. The system is designed so that it does not only consist of an AI model, but also provides an entire pipeline that aims to teach the detection of environmental problems that exist in all major cities in the territory of Bosnia and Herzegovina. The presented system includes the integration of the entire pipeline consisting of Kafka, automatic garbage recognition with the help of a trained YOLOv8 model, analytics, distance calculation with the help of visual presentation of problems with the help of Haversine formula, and visualization of metadata on a map that represents an interactive custom dashboard. The main part of this pipeline includes an AI model that is trained to detect garbage, but also an implementation that includes determining and assessing the severity of the problem, which also includes the distance of the camera from the problem itself in order to more accurately determine the size of the problem related to the environmental problem in the form of an overloaded container. The research deals with the results obtained from the trained model as well as insights into the ability of the AI model to determine and prioritize environmental problems. The model was trained on approximately 4000 images and achieved an Average Precision (mAP@0.5) of 0.82 and an image–level waste–presence classification F1–score of 98.3 percent. In addition, the model works in near real time, which consists of 24 ms per image, roughly 41 frames per second, on a Tesla T4 GPU. The entire pipeline was tested on the basis of images collected from the city of Sarajevo, which includes 258 images, of which 273 recognized objects are marked as garbage and 24 images are marked as environmentally acceptable. One of the important items in this paper is the position of the image and the container. Metadata about the location of the container was downloaded from the website of the utility service, while the distance of the photo was determined based on GPS coordinates. The paper provides an insight into the results obtained from this research, but also the limitations that this type of implementation faces.

Keywords: object detection, YOLOv8, geospatial analysis, environmental monitoring, urban waste management, Apache Kafka, severity assessment, smart city

INTRODUCTION

Advances in computer vision technology have significantly enabled better, faster, and simpler automation and processing of images collected by mobile devices. Various algorithms related to object recognition in various systems have improved significantly and are already widely used, some examples are environmental monitoring, traffic analysis, and urban management [1], [2]. In environmental protection situations, one of the basic things for preventing pollution is early detection of problems, where early detection of problems gives authorities the space to adequately develop a strategy and act appropriately.

However, visual detection alone cannot reveal the real problem, and the scale and magnitude of the crisis. For example, there are several situations in which we have a picture of an environmental problem from close up and a picture of an environmental problem from a distance. So, environmental problems photographed from close up may seem larger than they actually are. In addition, additional data such as whether the environmental problem is next to a container or an illegal waste dump cannot be clearly determined, and this is a limitation of a system that relies solely on visual evidence. To address these shortcomings, this paper offers solutions that integrate geospatial information into the analysis.

The data combined in this paper represent data that includes GPS coordinates and container coordinates. Such data allows for distance-based problem analysis, spatial hotspot analysis, and context-based understanding that cannot be achieved from images alone. An additional problem affecting Bosnia and Herzegovina is poor analysis and integration of digitalization, which results in poor knowledge of the terrain and poor implementation of any measures for better conservation. This paper presents a system based on an AI model that combines object detection with geospatial analysis to enable better monitoring of environmental problems. Images taken by users are used as the main input, and these images are processed through our researched AI model, and metadata data such as geodata about containers and image location provide us with a clearer insight into the scale of the problem. The main contributions of this work are (i) the design of a comprehensive process that includes detection and analysis, not just a model, (ii) the integration of distance-based geospatial analysis to improve reliability in assessing the severity of eco-problems, and (iii) experimental testing and evaluation of the detection performance and behavior of the complete system in a real urban environment through the use of locally collected data.

METHODOLOGY

The system described in the research paper deals with the integration of computer vision, geospatial analysis, and data visualization into one complete pipeline that is used for the purpose of pipeline monitoring. The architecture consists of several phases and it includes data entry in real time, image detection, geo analysis and visualization on an interactive dashboard. Below are the detailed steps in this pipeline, as well as the logic and way of working.

Data Ingestion and Streaming Pipeline

At the very beginning, Apache Kafka [3] was implemented as support for real time operation. In this part of the pipeline, Kafka behaves as a middleware used for message exchange that separates data input from downstream processing, thus enabling a scalable solution and much more resistant to errors that occur in reports from the handling side. The images that arrive at the Kafka input contain various metadata data, including time stamp, geolocation, and information about the report. Kafka services recognized as consumers subscribe to these messages and perform downstream processing, where these processes include object detection and geospatial analysis. This design enables more data to be processed at the same time in real time, where scalability and consistency in receiving and forwarding data is enabled, and reduces potential data losses caused by problems in the network. The database used for this is Firebase, and this database enables the safe storage of these data and processed results from where these data are used for the final display of visual data as well as analysis.

Image-Based Detection

Data captured by a mobile phone in the form of images serve as the primary input. After that, the images are processed with the help of a trained deep learning object detection model based on YOLOv8. For each possible detection that the trained model finds, the model provides a frame, object class and confidence level. Given that there are variations in the size of the object and the distance of the camera, reliability thresholds and calculations of the surface of the object are used for a more accurate assessment of the seriousness of the eco-problem, and for this reason, several categories have been implemented, such as the category of low, medium, and high priority, where the priorities are based on the number of objects and spatial coverage within the image.

Model Training and Configuration

The YOLOv8 detection model is trained on a set of 4000 labeled images. It is divided into 70-20-10 sets where 70 percent is training then we have 20 percent for validation and after that 10 percent for testing to ensure a balanced presentation of waste instances. The training was performed during 50 epochs. Convergence is indicated by the stabilization of the validation metric. Also in addition, the AdamW optimizer was used with a learning rate of 0.001 and a batch size of 16. All input images were modified to 640x640 pixels, and this is the recommended value for YOLOv8. Standard augmentations, including flip, rotate, and color change, were applied to improve generalization as much as possible. Google Colab is used to train this type of model where a Tesla T4 series GPU is used. The confidence threshold is set at 0.5, and was chosen based on precision and recall analysis to balance false positives and recall.

Geospatial Analysis and Severity Assessment

Each incoming image is associated with GPS coordinates representing the location of the recording. The received GPS coordinates are compared with the database of coordinates of garbage containers, which you download from the official website of the municipal service. The shortest distance between the received image with the location and the coordinates of the nearest container is calculated using Haversine formula [10], the purpose of which is to determine the distance on a great circle between two points on the Earth whose base is represented in the form of their geographic longitudes and latitudes. Using this distance-sensed approach reduces the bias obtained from the camera's perspective. As an example, one can cite more containers photographed from a distance and fewer containers photographed from close-up, where containers from close-up can take up more space in the image and thus deceive the system. By combining the estimated coverage with the distance to the nearest container, the implemented pipeline (system) qualifies incidents into low, medium and high severity categories of eco-problems. Waste located at a considerable distance from the container is treated as a serious incident because it is very likely an illegal waste disposal site, instead of a regularly maintained garbage container.

Data Aggregation and Visualization

Detection outputs and spatial data are stored in a structured format that includes image identifiers, followed by timestamps, number of objects detected, coverage, severity, and geolocation. Data preprocessing, cleaning and statistical analysis are performed in Python using the Pandas library [4], which supports scalable aggregation and reporting. For visualization, the processed results are exported in GeoJSON format [11] and displayed using the Leaflet mapping library [5]. Each incident that has occurred is displayed as a point on the interactive map, with a color and size that imply its severity or the number of detections, which has the ultimate goal of enabling users to explore spatial patterns, identify hot spots and prioritize the cleaning of marked locations.

RESULTS

The system was evaluated across three complementary dimensions:

- detection performance,
- operational speed, and
- end-to-end behavior on a locally collected dataset.

Two datasets were used in the testing. The first consisted of approximately 4000 labeled images for training and validating the detection model.

The second was a separate validation dataset of 258 images captured in Sarajevo, containing 234 images with visible litter (a total of 273 labeled litter examples) and 24 clean images used as negative controls to assess false positive behavior. All images in the second dataset were associated with geolocation metadata corresponding to the actual locations of the containers.

Table 3. Local Sarajevo dataset composition

CATEGORY	COUNT
Total images	258
Images with waste	234
Annotated waste instances	273
Clean control images	24

Detection Performance

On the held-out test split, the YOLOv8 model achieved a precision of 0.8154, recall of 0.7604, mean average precision (mAP@0.5) of 0.8194, and mAP@0.5:0.95 of 0.5080. These values indicate reliable detection of debris objects at the standard overlap threshold, with the lower mAP@0.5:0.95 reflecting the more stringent localization requirement at higher overlap thresholds. In addition, a complementary image-level evaluation was performed to assess the system's ability to determine whether an image contains debris at all, which is often the most important issue in real-time monitoring. On the validation subset of 100 images, the model correctly identified 88 waste images and 9 clean images, with only 2 false positives and 1 false negative, yielding a precision of 97.78 percent, a recall of 98.9 percent, and an F1-score of 98.3 percent. A threshold-independent analysis confirmed these results, with a precision and recall area under the curve of 0.99 and a receiver operating characteristic area under the curve of 0.993.

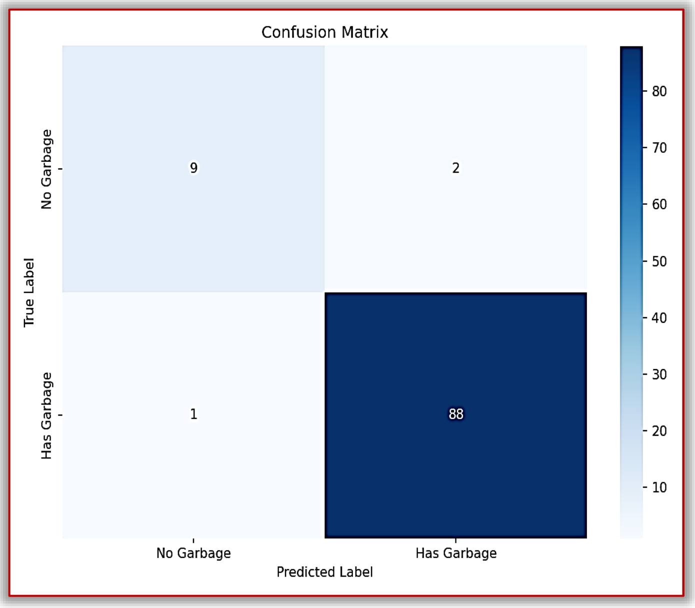

Figure 1. Confusion Matrix

Table 2. Detection model performance

METRIC	VALUE
Precision	0.8154
Recall	0.7604
mAP@0.5	0.8194
mAP@0.5:0.95	0.5080
Mean inference time	~24 ms (~41 FPS)

Inference Speed

Inference performance was measured to assess suitability for near real-time deployment. The model achieved a mean inference time of approximately 24 ms per image on a Tesla T4 GPU, corresponding to roughly 41 frames per second. This confirms that detection is fast enough to support live, mobile, and web-based ecological monitoring.

Coverage, Severity, and Geospatial Results

Coverage was estimated as the proportion of image area covered by detected bounding boxes, and combined with the Haversine distance to the nearest container to assign severity. Across the evaluated reports, detections were georeferenced and related to real container locations, with nearest-container distances typically ranging from about 21 m to 79 m. Incidents were distributed across moderate and distant proximity bands, and severity labels were concentrated in the higher categories when waste was present, reflecting the combined influence of visual extent and spatial context. All processed detections were successfully visualized on the interactive map, where locations requiring urgent intervention could be clearly distinguished from those requiring little or none.

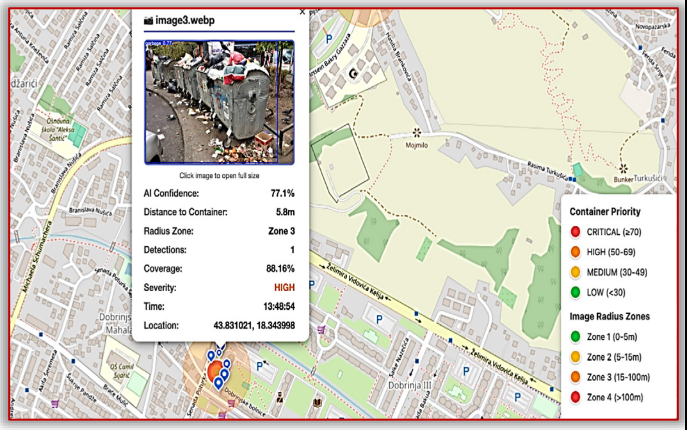

Figure 2. Dashboard / map

System Reliability

At the system level, the pipeline processed the full dataset and produced one detection record per input image, with a verified one-to-one correspondence between inputs and stored results. The decoupled, persistent design of the Kafka-based architecture allowed any items missed because of transient processing conditions to be re-ingested and recovered rather than lost, confirming the fault-tolerance rationale for the streaming design.

DISCUSSION

The results demonstrate that combining AI-based detection with distance-aware geospatial analysis produces a more reliable assessment of environmental incidents than visual analysis alone. The detection model reliably recognizes the presence of waste, as shown by the high image-level F1-score, while the bounding-box metrics indicate strong but stricter performance at the localization level. The coexistence of these two figures is expected rather than contradictory: determining whether an image contains waste is an inherently easier task than precisely localizing every instance, and the system performs strongly on the decision that matters most for rapid response.

The central methodological insight is that pixel-based coverage alone is insufficient because of camera-perspective effects. By incorporating the Haversine distance between each report and the nearest known container, the system corrects for this bias and adds genuinely decision-relevant context. This enables a distinction between overfilled containers, which are part of normal service, and waste in locations without nearby infrastructure, which is more likely to represent illegal dumping. Compared with prior approaches that rely primarily on visual coverage or object counts [6], [7], or that focus on waste-type classification [8] or sensor integration without explicit distance-aware severity [9], the proposed framework explicitly relates detections to infrastructure, which is especially valuable in underrepresented regions with sparse ecological data.

From an operational perspective, the streaming architecture supports continuous ingestion of new reports and transforms detection outputs into actionable geospatial alerts. High-severity incidents, identified through a combination of object count, coverage, and spatial distance, can be prioritized and visualized on the interactive map, allowing municipal services to locate critical hotspots quickly, plan cleanup operations, and monitor recurring problems over time. In practice, this replaces a fixed weekly inspection routine, in which crews may visit points that do not require attention while neglecting others, with a data-driven process that directs effort where it is most needed.

Several limitations should be acknowledged. First, the geolocation metadata used for local validation was simulated and matched to real container locations rather than collected through field GPS, the geospatial results should therefore be interpreted as a controlled proof-of-concept rather than a calibrated field measurement. Second, the severity thresholds are expert-defined and rule-based, providing reproducibility and transparency but not yet statistical validation against ground-truth severity judgments. Third, the model currently detects waste as a single class and does not distinguish specific incident types, such as overfilled containers versus illegal dumping, beyond what the distance heuristic implies. Finally, the temporal evaluation was conducted over a controlled batch rather than a long longitudinal period, so seasonal or long-term trend claims are outside the scope of the present study. These limitations directly motivate the future work described below.

CONCLUSIONS

This paper presented the development of an end-to-end, AI-based application for the collection and analysis of environmental incident data, aimed at supporting emergency response in ecological crisis areas of Bosnia and Herzegovina. The system integrates real-time data ingestion based on Apache Kafka, automated waste detection using a YOLOv8 model, distance-aware geospatial analysis based on the Haversine formula and real container locations, persistent storage in a real-time database, and interactive visualization through a GeoJSON and Leaflet map interface. Together, these components form a clear pipeline rather than an isolated model.

Experimental evaluation confirmed the feasibility and value of the approach. The detection model achieved an mAP@0.5 of 0.82 and an image-level waste-presence F1-score of 98.3 percent, while operating at near real-time speed of about 24 ms per image (approximately 41 frames per second). The complete pipeline was validated on a locally collected dataset of 258 images from Sarajevo, and all detections were successfully georeferenced and visualized, with priority locations clearly distinguished from low-priority ones. The distance-aware severity assessment was shown to mitigate the camera-perspective bias inherent in purely visual metrics, providing a more faithful representation of real-world conditions.

The principal significance of this work lies in demonstrating that affordable, open and widely available technologies can be combined into a practical environmental-monitoring tool for a region where such tools have been largely absent. By converting unstructured, user-generated imagery into prioritized, geographically contextualized information, the system offers municipalities a path toward evidence-based decision-making: it can reduce unnecessary field visits, accelerate response to genuine hazards, and provide objective data on where container capacity should be increased or reduced.

Future work will focus on replacing simulated geolocation with real field-collected GPS data, statistically validating the severity model against expert assessments, extending the detector to distinguish specific incident types such as overfilled containers and illegal dumping, and deploying a dedicated mobile application for citizen reporting. Longitudinal data collection over multiple seasons will further enable temporal trend analysis and predictive hotspot modeling. Overall, the proposed framework provides a scalable and reproducible foundation for urban waste monitoring and ecological incident response in Bosnia and Herzegovina and in similar urban environments.

References

[1] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv8: State–of–the–Art Object Detection," Ultralytics, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics
[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real–Time Object Detection," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788.
[3] J. Kreps, N. Narkhede, and J. Rao, "Kafka: A Distributed Messaging System for Log Processing," in Proc. NetDB Workshop, 2011, pp. 1–7.
[4] W. McKinney, "Data Structures for Statistical Computing in Python," in Proc. 9th Python in Science Conf. (SciPy), 2010, pp. 56–61.
[5] Leaflet Contributors, "Leaflet: An Open–Source JavaScript Library for Interactive Maps," 2023. [Online]. Available: https://leafletjs.com/
[6] Y. Zhang, L. Chen, and H. Wang, "Real–Time Urban Waste Detection Using YOLO–Based Deep Learning Models," IEEE Access, vol. 12, pp. 33421–33434, 2024.
[7] J. Park, S. Kim, and H. Lee, "Transformer–Assisted Object Detection for Illegal Dumping Recognition in Urban Environments," Pattern Recognition Letters, vol. 178, pp. 14–23, 2025.
[8] A. Kumar, R. Patel, and S. Verma, "Multi–Class Waste Detection and Segmentation Using Deep Convolutional Networks," Sustainable Cities and Society, vol. 105, 2025.
[9] D. Silva, M. Costa, and J. Ribeiro, "Real–Time Smart Waste Monitoring Using Deep Learning and IoT Integration," Sensors, vol. 25, no. 3, 2025.
[10] R. W. Sinnott, "Virtues of the Haversine," Sky and Telescope, vol. 68, no. 2, p. 159, 1984.
[11] H. Butler, M. Daly, A. Doyle, S. Gillies, S. Hagen, and T. Schaub, "The GeoJSON Format," RFC 7946, Internet Engineering Task Force (IETF), 2016.
[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.
[13] T.–Y. Lin et al., "Microsoft COCO: Common Objects in Context," in Proc. European Conf. Computer Vision (ECCV), 2014, pp. 740–755.
[14] F. Pedregosa et al., "Scikit–learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
[15] S. M. Abdullahi, H. Yusuf, and A. Musa, "A Review of Smart Waste Management Systems for Smart Cities: Technologies, Challenges, and Future Directions," Journal of Environmental Management, vol. 351, 2024.

ISSN: 2067–3809