Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

AI Based Visual Pollution Detection in Urban Environments

Chandanasiri K, Hemavathi M P
Pranathi S, Rakshitha C
Dept. of Computer Science and Engineering
G Madegowda Institute of Technology
Mandya, Karnataka, India

Ms. Madhushree
Assistant Professor (Project Guide)
Dept. of Computer Science and Engineering
G Madegowda Institute of Technology
Mandya, Karnataka, India

Abstract **- Visual pollution such as uncollected garbage, advertising clutter, wall markings, exposed wires and construction debris degrades the appearance and usability of urban spaces, yet it is still monitored mainly through manual field inspection that is slow and inconsistently documented. This paper presents UrbanLens AI, a web based application that detects visual pollution in street imagery using the YOLOv8n object detection model. The system identifies six trained categories, namely garbage, billboards, posters, graffiti and wall markings, wires, and construction debris, and summarizes each scene with an Urban Visual Pollution Index (UVPI) from 0 to 100 that is mapped to Low, Medium, or High severity. The application accepts uploaded images, browser camera frames and selected video frames, and provides annotated results, rule based cleanup recommendations, analysis history, dashboards, location based maps, before and after cleanup comparisons, downloadable PDF reports, and on demand Grad-CAM and LIME explanations. The model was trained on 2,365 images from public Peshawar and Dhaka street datasets. On 353 held out test images it recorded a precision of 54.46 percent, a recall of 46.99 percent and a mean average precision at an overlap threshold of 0.5 of 47.51 percent, with construction debris detected best and wires worst. The results show that the prototype is a workable monitoring aid but needs local data and further training before dependable civic use.**

Keywords - visual pollution; urban environments; object detection; YOLOv8n; explainable AI; urban visual pollution index; smart cities

I. INTRODUCTION

Rapid urban growth has made the visual quality of streets a public concern. Visual pollution covers visible elements that make an urban scene untidy or unpleasant, including garbage, excessive advertising, unauthorized posters, graffiti and wall markings, tangled or exposed wires, and construction debris. Municipal teams usually find such problems through repeated field visits and complaints. The resulting photographs and notes are often stored without consistent categories or location details, which makes it hard to compare locations, judge severity, or confirm that a cleanup has taken place.

Computer vision offers a way to automate part of this work. Recent studies have applied object detectors to visual pollution and waste monitoring using street view imagery, explainable models, low cost IoT hardware, and decision support for cleaning teams [1]–[5]. Most of these works, however, concentrate on a single task such as detection accuracy, a single category such as garbage, or a standalone prototype. Fewer describe an integrated workflow that runs from detection to scoring, mapping, reporting and review.

This paper presents UrbanLens AI, a web based application that detects visual pollution in urban images and organizes the results for review. The main contributions are as follows.

- Detection of six visual pollution categories with the lightweight YOLOv8n model, with bounding boxes and confidence values.
- An Urban Visual Pollution Index (UVPI) from 0 to 100 and a Low, Medium, or High severity class for each analyzed scene.
- A single web application with image, camera and video frame analysis, history, dashboards, geotagged maps, PDF reports, cleanup comparisons and rule based recommendations.
- On demand Grad-CAM and LIME explanations and an optional TensorFlow Lite edge runtime.

The project addresses visual pollution only. It does not measure air quality or noise pollution. The rest of the paper is organized as follows: Section II reviews related work, Section III states the problem and objectives, Section IV describes the proposed system, Section V lists the tools used, Section VI explains the implementation, Section VII reports the results, and Sections VIII and IX give the conclusion and future scope.

II. LITERATURE SURVEY

Table I summarizes five related studies. Hossain et al. [2] detected visual pollution in Google Street View images with YOLOv5 to help locate polluted areas, but found that street litter and tangled wires were harder to detect because of their varied shapes, colours and textures, and that manual image collection and annotation took considerable effort. Al Mazroa et al. [1] combined computer vision with explainable artificial intelligence to detect and classify visual pollution in Saudi Arabia, at the cost of several deep learning components that raise implementation complexity and computing needs.

Alsolami et al. [3] proposed a low cost monitoring system in which a webcam on a Raspberry Pi 4 captures images, a trained neural network detects recyclable waste, and the results reach a cloud dashboard over Wi-Fi. Its coverage is limited to the camera field of view and it depends on hardware and connectivity. Reddy et al. [4] compared deep learning architectures for waste detection and sorting, including lightweight YOLOv8 and YOLOv11 variants with attention

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

and transformer based designs, and noted that heavier models need more computing resources and that lighting, occlusion and cluttered backgrounds affect performance. Ayman et al. [5] joined YOLO based detection with vehicle routing optimization, turning detections into structured location data and maintenance plans for municipal cleaning teams, with maps and dashboards. Their system concentrates on garbage accumulation, and the authors list multi vehicle scheduling, citizen feedback, traffic aware routing and wider pollution coverage as future work.

TABLE I. SUMMARY OF RELATED WORK

Ref. / Year	Technique	Limitation
[2] 2021	Google Street View with YOLOv5	Litter and wires hard to detect; manual collection and annotation
[1] 2024	Computer vision with explainable AI	Many deep learning components; high complexity and compute
[3] 2025	CNN, Raspberry Pi, IoT dashboard	Limited to camera view; needs hardware and network
[4] 2026	Lightweight YOLO, attention and transformer models	Heavier models need more compute; lighting and occlusion effects
[5] 2026	YOLO detection with vehicle routing	Focus on garbage; broader categories and scheduling pending

These studies show that YOLO family detectors suit visual pollution work and that explainability, deployment and decision support each add value. None of them, as reported, offers one lightweight application that combines multi category detection, a transparent scene score, geotagged mapping, cleanup comparison and on demand explanations. UrbanLens AI is designed to fill this gap at prototype level.

III. PROBLEM STATEMENT AND OBJECTIVES

A. Existing Practice and Problem Statement

Current monitoring relies on manual field inspection, which needs repeated visits and staff effort. Photographs and complaints may be stored without consistent categories or location details, visual assessments can differ between observers, separate tools are often needed for detection, mapping and reporting, and comparing conditions before and after cleanup requires well organized records.

The problem addressed here is to design and develop an AI based web application that detects visual pollution related objects in urban imagery, estimates scene severity with a transparent scoring method, and organizes the results through location based maps, analytics and downloadable reports. The aim is to support consistent documentation and help users prioritize locations for further inspection and cleanup.

B. Objectives

- Detect and localize six trained visual pollution categories.
- Calculate a UVPI score between 0 and 100.
- Classify analyzed scenes into Low, Medium, or High severity.
- Provide category specific corrective recommendations.
- Maintain analysis history, dashboards and geographical maps.
- Support PDF reporting, cleanup comparisons and model explanations.

IV. PROPOSED SYSTEM AND METHODOLOGY

A. System Overview

UrbanLens AI is a unified web application for visual pollution detection and reporting. The YOLOv8n model [6] identifies the supported objects and returns bounding boxes with confidence values. A scoring service then computes the UVPI, location tagged analyses appear on an interactive map, and users can review earlier results and compare before and after images of a location.

B. Methodology and Workflow

Fig. 1 shows the processing pipeline. The user supplies an image, a video or a browser camera frame, and the input is validated and prepared for inference. YOLOv8n predicts categories, confidence scores and bounding boxes. The scoring service calculates the UVPI and assigns a severity level, after which rule based corrective suggestions are produced. The annotated image, scores and detected categories are displayed, and the analysis is saved for history, maps, dashboards and PDF downloads.

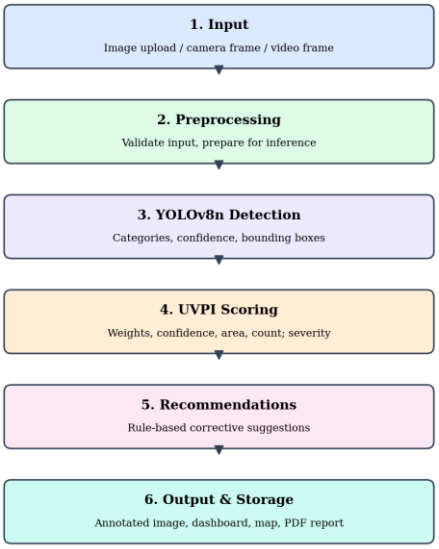


Fig. 1. Processing workflow of UrbanLens AI.

C. Urban Visual Pollution Index

The UVPI summarizes the detected pollution level of a scene on a scale of 0 to 100. The score considers category weight, detection confidence, occupied image area and object count, so a higher score indicates a larger weighted contribution from the detected objects. The score is then mapped to a severity class using the thresholds in Table II. The UVPI is a project defined heuristic and requires local calibration before it is used for decisions.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

TABLE II. UVPI SEVERITY LEVELS

UVPI score	Severity
0 to below 34	Low
34 to below 67	Medium
67 to 100	High

D. Model Explanations

Gradient based class activation mapping (Grad-CAM) [7] and local interpretable model agnostic explanations (LIME) [8] are offered for compatible saved analyses. They run on demand rather than for every image because LIME needs many inference passes, and they help a reviewer see which image regions influenced a detection.

V. TOOLS AND TECHNOLOGIES

The technologies used to build the system are listed in Table III.

TABLE III. TOOLS AND TECHNOLOGIES USED

Component	Technology
Programming language	Python
Backend framework	Flask
Frontend	HTML, CSS, JavaScript, Bootstrap
Object detection	Ultralytics YOLOv8n, PyTorch
Image processing	OpenCV, Pillow, NumPy
Database	SQLite with SQLAlchemy
Charts and maps	Chart.js, Leaflet
Explainable AI	Grad-CAM, LIME
PDF reports	ReportLab
Optional edge inference	TensorFlow Lite

VI. IMPLEMENTATION

A. Dataset Preparation

Public street image datasets from Peshawar [9] and Dhaka [10] were combined to give 2,365 images, split as shown in Table IV. The model is trained on six categories: garbage, billboard, poster, graffiti or wall markings, wires, and construction debris.

TABLE IV. DATASET SPLIT

Subset	Images
Training	1,673
Validation	339
Test	353
Total	2,365

B. Model Training

YOLOv8n was trained for 25 epochs with an input size of 384 pixels and a batch size of 8. The small model and input size keep training and inference light enough for modest hardware.

C. Web Application

The trained model is integrated into a Flask application. The application provides registration, login, user profiles and administrator access. Users can upload images, run continuous browser camera analysis, process selected video frames, and save analyses to a history stored in SQLite. Weighted UVPI scoring, severity classification and rule based recommendations run on every analysis.

D. Visualization and Reporting

Dashboards present analysis totals, category distribution, severity statistics and score trends using Chart.js. Saved geotagged analyses appear on a Leaflet map as severity markers or as a heatmap, and the map refreshes every 15 seconds. Downloadable PDF reports are generated with ReportLab, and before and after images of a location can be compared with the change in score.

E. Explainability and Edge Support

Grad-CAM and LIME explanations are generated on demand for compatible saved PyTorch detections. For edge use, the model can be exported to FP16 TensorFlow Lite and run with a standalone interpreter based detector. Performance on a device depends on its hardware.

VII. RESULTS AND DISCUSSION

A. Detection Performance

The installed YOLOv8n model was evaluated on the 353 held out test images. The recorded metrics are given in Table V. Construction debris achieved the highest class level mean average precision at 0.5 overlap (mAP@50) of 78.44 percent, while wires achieved the lowest at 18.70 percent.

TABLE V. DETECTION RESULTS ON THE TEST SET

Evaluation metric	Recorded result
Precision	54.46%
Recall	46.99%
mAP@50	47.51%
mAP@50–95	27.25%

B. Application Output

Fig. 2 shows the application landing page with a detection result, in which an object is labelled construction debris with a confidence of 0.77. Fig. 3 shows the monitoring dashboard, which reports total analyses, average UVPI and the number of low and high risk scenes. Fig. 4 shows the pollution map, where each saved geotagged analysis appears as a marker and a heatmap view is available. The analysis history lists each saved result with its identifier, location, object count and UVPI.

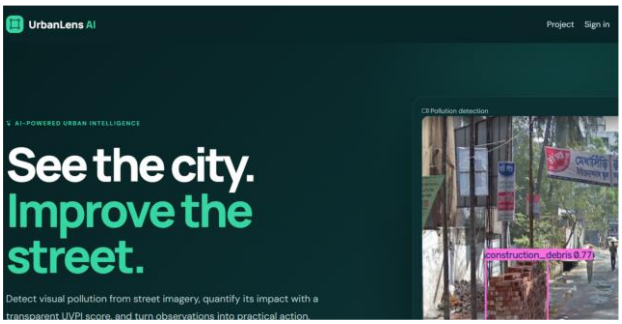


Fig. 2. Detection result with bounding box, class label and confidence.

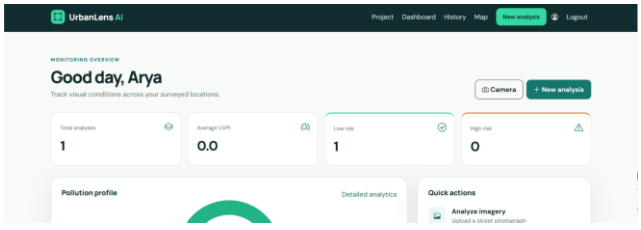


Fig. 3. Monitoring dashboard of UrbanLens AI.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

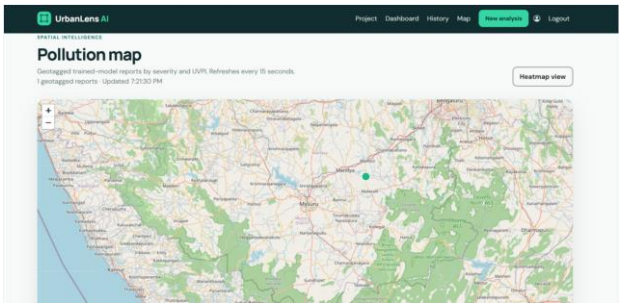


Fig. 4. Pollution map of geotagged analyses.

C. Discussion and Limitations

The metrics show that the model finds many objects but still misses some and produces false detections, so more training data and model improvement are needed, particularly for wires. Thin and irregular objects such as wires were also reported as difficult in earlier work [2]. The figures describe performance on the public test dataset, and the training images come from Peshawar and Dhaka, which may differ from Indian streets, so local performance needs separate evaluation. The UVPI is a heuristic that has not yet been compared with expert assessments. The system is therefore an academic monitoring prototype and not a replacement for inspection by trained staff.

VIII. CONCLUSION

UrbanLens AI combines object detection, severity scoring, mapping and reporting in one web application. YOLOv8n supports detection of six visual pollution categories, and the UVPI gives a transparent way to summarize a scene. History, recommendations, explanations and cleanup comparisons support structured review of results. On the public test set the model recorded 54.46 percent precision, 46.99 percent recall and 47.51 percent mAP@50, which shows that local validation and improved detection are needed before dependable civic use.

IX. FUTURE SCOPE

- Expand the dataset with locally collected and annotated street images from Indian cities.
- Improve detection of wires and other difficult objects, and train additional visual pollution categories.
- Calibrate the UVPI against expert assessments.
- Develop mobile citizen reporting and integrate municipal complaint and cleanup workflows.
- Add privacy protection through face and number plate redaction.
- Evaluate performance on edge devices and larger video workloads.

ACKNOWLEDGMENT

The authors thank Ms. Madhushree, Assistant Professor, Department of Computer Science and Engineering, G Madegowda Institute of Technology, Mandya, for her guidance throughout this project.

REFERENCES

[1] A. Al Mazroa, M. Maray, A. M. Alashjaee, F. A. Alotaibi, A. A. Alzahrani, A. Alkharashi, S. D. Alotaibi, and M. M. Alnfiai, “Computer vision with explainable artificial intelligence for visual pollution detection in the Kingdom of Saudi Arabia,” IEEE Access, 2024.

[2] M. Y. Hossain, I. R. Nijhum, A. A. Sadi, M. T. M. Shad, and R. M. Rahman, “Visual pollution detection using Google Street View and YOLO,” in Proc. 2021 IEEE 12th Annu. Ubiquitous Computing, Electronics & Mobile Communication Conf. (UEMCON), 2021.

[3] R. Alsolami, R. Atwah, R. Alajlani, and S. Alhajaili, “Smart visual pollution detection system,” in Proc. 2025 2nd Int. Conf. Advanced Innovations in Smart Cities (ICAISC), 2025.

[4] A. V. Reddy, B. Hima Sagar, S. P. Kumar, and B. S. Devi, “AI-powered waste detection and sorting using object detection,” in Proc. 4th Int. Conf. Self Sustainable Artificial Intelligence Systems (ICSSAS), 2026.

[5] Y. M. Ayman, E. A. Elgendi, and I. A. F. El-Khodary, “An intelligent decision support system for visual pollution mitigation in smart cities,” in Proc. 2026 IEEE Int. Conf. Smart Sustainable Systems for Computer and Engineering Applications (3SCEA), 2026.

[6] G. Jocher, A. Chaurasia, and J. Qiu, Ultralytics YOLOv8, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[7] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual explanations from deep networks via gradient-based localization,” in Proc. IEEE Int. Conf. Computer Vision (ICCV), 2017, pp. 618-626.

[8] M. T. Ribeiro, S. Singh, and C. Guestrin, “Why should I trust you?: Explaining the predictions of any classifier,” in Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, 2016, pp. 1135-1144.

[9] B. Tariq, Visual Pollution Dataset Peshawar Pakistan, Kaggle dataset.

[10] M. Y. Hossain et al., Visual Pollution Dhaka Streets, Kaggle dataset.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)