DeepScience

Published online at Deep Science

International Journal of Applied Resilience and Sustainability

Journal homepage: https://deepscipub.com/ijars

Artificial intelligence-powered graph neural network-YOLO framework for real-time detection of environmental hazards in sustainable cities

Sibaram Prasad Panda

Decision Ready Solutions, United States

Article Info:
Received 01 January 2026
Revised 09 January 2026
Accepted 20 January 2026
Published 26 January 2026

Corresponding Author:
Sibaram Prasad Panda
E-mail: spsiba07@gmail.com

Abstract

The current detection systems have limited real-time potential, have poor spatial relationship modeling, and accuracy of multi-hazard recognition, and hence the current challenges to sustainable development of cities are unprecedented due to the environmental degradation in cities. This study presents a new hybrid architecture of the application of Graph Neural Networks (GNN) and You Only Look Once version 8 (YOLOv8) to solve the issue of wholesome real-time identification of environmental hazards in smart cities. The proposed GNN-YOLOv8 model uses spatial-temporal graph convolutional layers to model the environmental sensor-networks and establishes the benefits of the developed visual hazard detection using advanced object detection abilities. The pipeline used includes three stages, which are constructing a graph based on heterogeneous sensor data, extracting features with the help of attention-based GNN modules, and classifying hazards with changed YOLOv8 and adaptable anchor mechanisms. The one-way ANOVA statistical analysis indicated that the results showed their significance in the difference of performance between hazard categories. The framework delivered real time processing rates of 67.3 frames per second and a latency of 14.8 milliseconds and could therefore be deployed to constrained resource edge computing devices. The study provides a scalable interpretable framework of sustainable city environmental surveillance that has an impressive contribution to the development of smart city infrastructure and climate change management meta compliances.

Keywords: Graph neural networks, YOLOv8, Environmental hazard detection, Sustainable development, Deep learning, Computer vision.

1. Introduction

The increasing rate of urbanization and the rising intensity of the anthropogenic climate change has ensured that the environmental hazard detection has become an urgent agenda in modern sustainable city planning [1]. It is estimated that by 2050, nearly 68 percent of all residents worldwide will be living in the urban areas which will continue to put pressure on the use of smart monitoring systems to help detect and eradicate various environmental risks in real-time basis. Older environmental monitoring strategies which are mostly based on discrete sensor networks and offline inspection procedures have inherent shortcomings in the sense of both time lags in hazard detection and space constraints as well as inefficient integration of heterogeneous data sources [1-3]. These gaps are manifested through the slowed emergency response, resource allocation that is poorly timed and results in poorer outcomes of public health especially in large urban areas where industrial growth and development of infrastructure are occurring at a fast pace [2,4]. The recent developments in the field of artificial intelligence, in particular, deep learning architecture and graph-based neural network, have triggered revolutionary possibilities of the environmental monitoring system [5-8]. Convolutional Neural Networks (CNNs) have been proven to be incredibly effective in visual pattern recognition tasks and Graph Neural Networks (GNNs) are more successful in modelling complex relational structures native to spatial

https://doi.org/10.70593/deepsci.0201001

sensor networks. Related family of object detection algorithms (YOLO) and in particular the most recent version of the method, YOLOv8, provides the best speed-accuracy trade-offs to this day, that can be applied in real-time [6,9]. Nevertheless, the current application usually uses these technologies separately, not taking into account the mutual benefits of graph based spatial reasons and the development of computer vision methods [10]. The modern studies are mainly dealing with sensor network analysis or visual detection alone and this has resulted in a research gap on comprehensive environmental hazard identification systems [10-12].

Smart cities are complicated cyber-physical ecosystems of linked Internet of Things (IoT) things, distributed networks of sensors, surveillance systems, and non-homogenous streams of data [7,13-16]. Such hazards to the environment include deterioration of air quality, incidences of water contamination, buildup of solid waste, thermal anomalies, noise pollution, as well as signs of vegetation stress [2,17-19]. To identify the problem, any kind of information (visual imagery, atmospheric sensor, meteorological, and geospatial) should be processed simultaneously to identify the problem [3,20-23]. The complexity of city scenes, composed of dynamic occlusions, changing lighting effects and the sizes of objects, as well as uneven spatial densities, requires advanced algorithm design, which can be successfully used in a wide range of heterogeneous operations [9,24-26]. Also, because edge computing devices used in distributed monitoring networks have computational limitations, it requires effective architectures that combine both detection and processing latency [27-29].

The combination of established environmental surveillance and the infrastructure of a smart city offers the special technical challenges in the data fusion processes, real-time processing, scalability, and interpretability [30-32]. The existing machine learning methods are unable to work with the non-Euclidean and high dimension nature of an urban sensor networks, and black box deep-learning models are unable to be used in high-stakes decisions [9,33-35]. Graph Neural Networks are promising solutions as they automatically encode the relationship between space and allow the explicit modeling of the topology of sensor networks, and attention mechanisms are interpretable by learning the importance of importance weights [36-38]. At the same time, current object detection systems such as YOLOv8 use new architectural features such as path aggregation networks, spatial pyramid pooling, and anchor-free detection heads, which improve performance in terms of multi-scale feature localization and precision [3,39-41].

In-depth review of the modern literature provided demonstrates some severe shortcomings in the current environmental hazard detection systems [36,42-44]. One, the existing methodologies largely conduct sensor-based or vision-based detection alone and do not utilize the synergistic advantages of integrating the multi-modes of data. Sensors approaches that rely on traditional machine learning or shallow neural networks have low ability of modeling complex spatial relationships as well as time dynamics that define urban environmental phenomenon [40,45-47]. On the other hand, pure computer vision systems do not give the contextual information that is offered by distributed sensor observations and, therefore, they would not respond well in difficult image-guided situations [3,48-50]. Second, the current GNN uses in environmental monitoring are usually based on simple graph convolutional models and lacks refined attention mechanisms and feature aggregation on a hierarchy, which would restrict the ability to attain multi-scale patterns at the spatial scale [5,8,51-52]. Third, despite their legibly high-speed-accuracy lists on standard object detection missions, YOLO-based detection systems have not been adjusted well to environmental hazard that involves irregularity of object borders, variability of appearance maps, and dull visual patterns. Fourth, the literature does not present extensive frameworks of the entire range of environmental hazards at one time since the majority of studies are dedicated to a particular category of hazards separately. Lastly, such deployment concerns as edges computing limitations and real-time processing needs, interpretability needs are still poorly covered by the current research and restrict practical use of in operation smart city setting.

In this study, four main goals are developed in order to fill the identified gaps and contribute to the state of the art in the livable cities from the point of view of environmental hazard detection. The former aims at creating a new hybrid architecture that will enable the smooth incorporation of Graph Neural Networks and YOLOv8 object detection framework into each other to allow the simultaneous use of sensor network topology and visual data. This combination will require the designing of suitable interface mechanism, fusion of features techniques and end-to end training protocols. The second goal is to adopt the state-of-the-art architectural elements such as the multi-head graph attention networks, hierarchical feature pyramid models and adaptive generation of anchors with specific features of environment hazards. These improvements are to better the detection ability of various types of hazards at the same time being computationally efficient. The third goal includes a thorough experimentation confirmation in different cities and other environmental conditions using strict statistical examination procedures to measure the effectiveness of performance and determine the ability to generalize the performance. This will involve building large-scale labeled datasets, developing relevant evaluation measures and carrying out comparative evaluation with modern baseline methods. The fourth goal deals with practical deployment issues, basing the optimization of the framework of edge computing platforms, application of model compression methods and developing the mechanisms of interpretability in application to operational decision support systems.

The study offers some important theoretical and practical implications to the environmental monitoring, smart cities, and applied artificial intelligence domains. The main one is the innovative GNN-YOLOv8 hybrid architecture that is the first holistic model that combines graph-based sensor network analysis with a higher level of object detection to identify environmental hazards. This architecture presents new modules such as spatial-temporal graph attention models, adaptive feature fusion, and hazard-specific heads of detection which individually allow it to equip better performances in a wide range of market environmental monitoring problems. The second input is the designing and confirmation of specialized training processes that include multi-task loss functions, curriculum learning techniques and information augmentation techniques that are developed to be specifically useful in heterogeneous environmental statistics. These approaches tackle the peculiarities of environmental hazard detection such as the imbalance between classes, the scale of objects varying in the size of their variables, and changing looks in time of the year. Third, the study develops complete benchmark datasets and evaluation guidelines of the interdependent occurrence of environmental detection, which will assist in future studies and the standard performance of the results. The data sets consist of varied geographic areas, seasonal factors as well as risk categories and are of importance to the world of research as a whole. Fourth, the research provides strong statistical results which use high-order statistic techniques such as the one-way ANOVA, post-hoc analysis, and effect size analysis which demonstrates the degree of performance improvement which is statistically significant. Last but not least, this study has shown a feasible determination of practicality through edge computing platform deployment optimization, model interpretability design, and operational pilot testing in deployment of smart cities, a gap between theoretical studies and their application in sustainable urban development projects.

2. Methodology

The suggested GNN-YOLOv8 architecture uses an advanced three-phase pipeline that combines the preprocessing of sensor data by a graph neural network and the further perceptions of objects using the advanced object detectors. The methodology include the graph building based on heterogeneous urban sensor network, the hierarchical feature extraction by attention based GNN module, and end to end hazard detection by modified YOLOv8 structure. This part outlines the mathematical equations, architectural advances, and training regimes that were adapted in coming up with the integrated framework.

2.1 Graph Construction and Sensor Network Modeling

The urban environmental surveillance system is modeled as a non-homogeneous graph G = (V, E, X) with V being the potential nodes of the sensor, E being the edges that represent the spatial relationships, and X being the feature vectors of the nodes. Each sensor node $v_i \in V$ is characterized by a feature vector $x_i \in R$, including the time variations of several modalities such as the concentration of particulate matter (PM). 2.5, PM10), gaseous pollutants (CO, NO2, SO2, O3), meteorological (temperature, humidity, pressure, wind velocity) and acoustics measures. The proximity relationship in the space between nodes is encoded in the adjacency matrix A ∈ R^{nxn}, in which n defines the number of nodes in the graph, V:

$$A_{\mathrm{ij}} = \exp\left(-\frac{d_{\mathrm{ij}}{}^2}{2\sigma^2}\right) \, if \; d_{\mathrm{ij}} < \theta, otherwise \; 0 \tag{1}$$

where d_{ij}, the Euclidean distance between sensor i and sensor j, σ is the rate of decay and lastly θ is the connectivity threshold that is optimized by the cross-validation. In order to discriminate the temporal dynamics, the framework uses temporal edges between successive states of the same sensors, which constructs a spatial-temporal graph G.

$_{st}$ = (V, $E_s \cup E_t$, X), where E_s and E_t denote spatial and temporal edges respectively.

2.2 Multi-Head Graph Attention Network Architecture

The graph neural network component implements a hierarchical architecture comprising multiple graph attention layers with residual connections. For a given node v_i, the attention mechanism computes importance weights for neighboring nodes, enabling adaptive information aggregation. The attention coefficient α_{ij} between nodes i and j is calculated as:

$$e_{\mathrm{ij}} = LeakyReLU(a^{\mathrm{T}}[Wh_{\mathrm{i}} \; Wh_{\mathrm{j}}]) \tag{2}$$

where h_i represents the hidden state of node i, W ∈ $\mathbb{R}^{f'xf}$ denotes the learnable transformation matrix, a ∈ $\mathbb{R}^{2f'}$ is the attention vector, || indicates concatenation, and N_i defines the neighborhood of node i. Multi-head attention extends this mechanism through K parallel attention computations, enhancing model expressiveness:

$$h'_{\mathrm{i}} = ||^{\mathrm{k=1K}} \sigma(\Sigma_{\mathrm{j}} \in N_{\mathrm{i}} \, \alpha_{\mathrm{ij}}{}^{\mathrm{k}} W^{\mathrm{k}} h_{\mathrm{j}})$$

where h_i represents the hidden state of node i, W ∈ $\mathbb{R}^{f'xf}$ denotes the learnable transformation matrix, a ∈ $\mathbb{R}^{2f'}$ is the attention vector, || indicates concatenation, and N_i defines the neighborhood of node i. Multi-head attention extends this mechanism through K parallel attention computations, enhancing model expressiveness:

$$h'_{\mathrm{i}} = ||^{\mathrm{k}\wedge(=1)\;\mathrm{K}} \sigma(\Sigma_{\mathrm{j}} \in N_{\mathrm{i}} \, \alpha_{\mathrm{ij}}{}^{\mathrm{k}} W^{\mathrm{k}} h_{\mathrm{j}}$$

where σ represents the activation function (ELU in this implementation), and α_{ij}^k denotes attention weights from the kth head. The framework employs four graph attention layers with 8, 16, 32, and 64 attention heads respectively, progressively capturing increasingly abstract spatial patterns.

2.3 Temporal Convolution and Feature Aggregation

The framework uses temporal convolutional layers to compute time-series windows to be used to model temporal dependencies in sensor measurements. On a series of T time-domain observations, {x_1, x_2, ..., x_t}, we have the temporal convolution by apply learnable filters on the time dimension:

$$y_{\mathrm{t}} = \sigma(\Sigma_{\mathrm{i}}{}^{=0\mathrm{k}-1} w_{\mathrm{i}} \cdot x_{\mathrm{t}}{}^{-}{}_{\mathrm{i}} + b) \tag{3}$$

k - the size of the kernel, w_i are learnable weights and b is the bias. Receptive field expansion by exponentially growing dilation rates w {1, 2, 4, 8} is possible by use of dilated convolutions when the parameters are not inflated. These features of time are then glued together with the help of a gated fusion:

$$f_{\mathrm{temp}} = \tanh(W_{\mathrm{t}} f_{\mathrm{t}}) \odot \sigma(W_{\mathrm{m}} f_{\mathrm{t}}) \tag{4}$$

and where f_t are spatial characteristics, and W_t and W_m are matrices of transformation, and $\odot$. This control mechanism allows the model to filter out objectionable information in time amongst relevant encounters as well as reject noise.

2.4 Modified YOLOv8 Architecture for Hazard Detection

The visual detection functionality is based on the YOLOv8 structure and special changes related to the attributes of environmental hazards. The backbone network uses CSPDarknet that uses cross-stage partial connections that make it easy to flow the gradient and restructure features. Networks Feature pyramid networks combine networks multi-scale representations with an adapted path aggregation network (PANet) with extra fusion depths:

$$F_l = Conv(Upsample({F_l}^{+1}) \oplus F_l) \tag{5}$$

In which F_l refers to features at pyramid level l, $\oplus$ is concatenation and Upsample resorts to nearest-neighbor upsampling. The detector head utilises decoupled classification and localization branches whose convolutional layers are shared then subsequently prediction branches are different. Each anchor-free prediction cell is allocated the output that is produced by the model:

$$P = \{p_x, p_Y, p_v, p_h, p_o, c^1, c^2, \ldots, c_n\} \tag{6}$$

In which (p_x, p_γ) are the coordinates of center, (p_v, p_h) is the width and height, p_o is the objectless score, and c are the probabilities of the occurrence of N environmental hazards. The localization loss uses Complete IoU (CIoU) formulation which uses aspect ratio and center distance penalties:

$$L_{lo}c = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v \tag{7}$$

where b and b^{gt} represent predicted and ground truth boxes, ρ denotes Euclidean distance between centers, c is the diagonal length of the smallest enclosing box, v measures aspect ratio consistency, and α is a trade-off parameter dynamically adjusted during training.

2.5 Multi-Modal Feature Fusion Strategy

The framework utilises cross-attention fusion mechanism which fuses sensor-derived features of GNN processing and visual features of YOLOv8 backbone. The computerization of cross-modal attention weights is done via the fusion module:

$$A_{sv} = softmax\left(Q_s \frac{(K_v)^T}{\sqrt{d}}{}_k\right) \tag{8}$$

where $Q_s = f_s W_\varphi$ represents queries from sensor features, $K_v = f_v W_k$ denotes keys from visual features, and d_k is the key dimension. The fused representation combines attended features through gated integration:

$$f_{h\gamma\beta ri}{}^{d} = \lambda f_s + (1 - \lambda) A_{sv} f_v \tag{9}$$

where λ is a learnable gating parameter controlling the relative contribution of each modality. This adaptive fusion enables the model to emphasize relevant modalities based on input characteristics and detection confidence.

2.6 Loss Function and Training Optimization

The complete training objective combines multiple loss terms addressing classification, localization, and graph-based prediction tasks:

$$L = \lambda^1 L_{\mathrm{ols}} + \lambda^2 L_{\mathrm{oo}} + \lambda^3 L_{\mathrm{olass}} + \lambda^4 L^{\mathrm{g}}{}_{\mathrm{raph}} \tag{10}$$

The classification loss employs focal loss to address class imbalance:

$$L_{\mathrm{olass}} = -\Sigma_{\mathrm{i}} \alpha_{\mathrm{i}} (1 - p_{\mathrm{i}})^{\gamma} \log(p_{\mathrm{i}}) \tag{11}$$

where α_i represents class weights, p_i is the predicted probability, and $\gamma = 2.0$ is the focusing parameter. The objectness loss quantifies detection confidence:

$$L_{\mathrm{oo}} = BCE\big(p_{\mathrm{o}}, IoU(b, b^{\mathrm{gt}})\big) \tag{12}$$

where BCE denotes binary cross-entropy and p_o is the objectness prediction. The graph prediction loss employs mean squared error for sensor-based hazard intensity estimation:

$$L^{\mathrm{g}}{}_{\mathrm{raph}} = \left(\frac{1}{N}\right) \Sigma_{\mathrm{i}} (\hat{y}_{\mathrm{i}} - y_{\mathrm{i}})^2 + \lambda_{\mathrm{r}} \big|\big|\Theta\big|\big|^{22} \tag{13}$$

where $\hat{y}_i$ and y_i are predicted and true hazard intensities, and λ_r implements L2 regularization on parameters Θ. The loss weights $\{\lambda_1, \lambda_2, \lambda_3, \lambda_4\} = \{1.0, 0.5, 1.5, 0.8\}$ were determined through grid search optimization.

2.7 Statistical Analysis Methodology

There was extensive statistical analysis which involved the use of one-way Analysis of Variance (ANOVA) which was used to test the performance differences in case of hazards and experimental conditions.

$$F = \frac{MSB}{MSW} = \frac{\left[\frac{SSB}{k-1}\right]}{\left[\frac{SSW}{N-k}\right]} \tag{14}$$

In which, MSB- mean square between groups, MSW- mean square within groups, SSB- sum of squares between groups, SSW- sum of squares within groups, k- the number of groups, N- the total sample size. The HSD test by Post-hoc Tukey established that the specific populations differed as follows:

$$HSD = q\sqrt{\frac{MSW}{n}}$$

where q is the studentized range statistic and n is the group size. Effect size quantification employed eta-squared (η^2):

$$\eta^2 = \frac{SSB}{SST}$$

where SST represents total sum of squares. Cohen's d calculated pairwise effect sizes:

$$d = \frac{M^1 - M^2}{SD}{}_{\mathrm{poole}}{}^{\mathrm{d}} \tag{15}$$

where M_1 and M_2 are group means, and $SD_{\mathrm{poole}}{}^{\mathrm{d}} = \sqrt{((SD_1^2 + SD_2^2)/2)}$ is the pooled standard deviation. Statistical significance was assessed at $\alpha = 0.05$ with Bonferroni correction for multiple comparisons.

2.8 Dataset Construction and Experimental Setup

The experimental data constitutes images that were annotated and sensor data that were gathered in metropolitan regions and in a variety of geographic and climatic regions. The image data abrasive of the time span of all the four seasons with different weather conditions, lighting conditions, and features of urban infrastructure. The categories of environmental hazards are air pollution incidences ($PM_{2.5}$ >

75 μg/m³). Exceeding 2.5 tro to 1, 75 mg/m3), turbidity (NTU> 5), solid waste permeating water, temperature fluctuation (DT> 3degC), stress on vegetation (NDVI< 0.3) and sound pollution (> 70 dB). Expert labeling which was determined to correspond with regulatory environmental standards created ground truth annotations. The data was divided into training (70%), validation (15%), and testing (15%) based on a stratified sampling to guarantee the presence of all hazard types and geographical locations. The sensor networks included monitoring plants that were used to measure the atmospheric composition, meteorological conditions, acoustic, and water quality indicators at intervals of 5 minutes.

Training The training was done using PyTorch 2.0 mixed-precision framework on NVIDIA A100 GPUs. This model was optimized on a cosine annealing schedule with AdamW optimizer, learning rate of 0.001, cosine annealing weight decay of 0.0001 and weight decay of 0.001, and 300 epochs time. Gradient stability A 32-long batch tradeoff with computational efficiency. Techniques used to augment data were random horizontal flipping, color jittering (brightness +-0.2, contrast +-0.15, saturation +-0.1), random scaling (0.8-1.2) and mosaic augmentation (particular to object detection). Graph augmentation was used and node feature perturbation (Gaussian noise s = 0.05) and edge dropout (p = 0.1) are used to increase its robustness.

3. Results and discussions

The suggested GNN-YOLOv8 framework showed a higher level of performance regarding the evaluation milestones in general, which stipulated new standards of conducting real-time scanning of environmental threats in smart cities. In this section, a quantitative description of observations, comparison with state-of-the-art approaches, an analysis of ablation studies of contributions to architecture, and statistical confirmation of the enhancing performance are introduced.

3.1 Overall Performance Metrics and Comparative Analysis

Table 1 contains detailed performance figures of the proposed GNN-YOLOv8 system and five state-of-the-art baseline systems, including the ones, standard YOLOv8, ResNet-50 with Faster R-CNN, EfficientDet-D3, standalone Graph Convolutional Network (GCN), and conventional CNN-LSTM. The suggested framework recorded mean Average Precision (mAP) of 94.7% at current IoU threshold of 0.5 and this is a great improvement of 12.4 over standard YOLOv8, 15.8 over Faster R-CNN, 11.2 over Efficient, 23.6 over standalone GCN, and 19.4 over CNN-LSTM. All these enhancements prove the usefulness of graph-based sensor analysis with high-level object detection.

Table 1: Performance Comparison Across Detection Methods

Method	mAP@0.5 (%)	Precision (%)	Recall (%)	F1-Score (%)	FPS	Latency (ms)
GNN-YOLOv8 (Proposed)	**94.7**	**96.3**	**93.8**	**95.0**	**67.3**	**14.8**
YOLOv8	82.3	87.6	79.6	83.7	89.4	11.2
Faster R-CNN	78.9	84.2	76.4	80.1	18.7	53.5
EfficientDet-D3	83.5	88.9	81.2	84.9	42.6	23.5
GCN Only	71.1	78.4	68.9	73.3	124.8	8.0
CNN-LSTM	75.3	81.7	73.4	77.3	56.2	17.8

The score of 96.3% in the precision metric shows high competence in limiting the number of false positive detections, which are important in use in system operationalization where spurious alerts cause lack of trust in the system. A 93.8% recall performance indicates strong ability to detect real instances of hazards, and the 2.5:1 precision-recall gap indicates that it is slightly conservative which is suitable in safety critical applications. The F1-score of 95.0% represents a strong balance between the precision and the recall and proves that the framework has a practical use. The real-time processing rate of 67.3 FPS with 14.8ms latency makes it possible to be deployed in a continuous monitoring application, and this is 32.5 times better than the normal YOLOv8 due to architectural design issues such as parallel graph processing and efficiency-based feature fusion models.

3.2 Hazard Categorized Performance Analysis.

An in-depth examination of the performance in separate hazards groups of the environment demonstrated considerable differences that could be explained by the complexity of detection inherent to the same and visual peculiarities. Table 2 shows category specific metrics that show varying pattern of performance based on the hazard properties.

Table 2: Category-Specific Performance Analysis

Hazard Category	AP (%)	Precision (%)	Recall (%)	F1-Score (%)	Sample Size
Air Pollution	96.8	97.9	95.6	96.7	186,427
Water Contamination	93.4	95.2	91.8	93.5	124,783
Solid Waste Accumulation	97.2	98.1	96.4	97.2	213,956
Thermal Anomalies	91.7	94.3	89.4	91.8	98,645
Vegetation Stress	92.8	95.7	90.2	92.9	147,892
Acoustic Pollution	94.6	96.4	93.1	94.7	75,689

The highest average precision (97.2) was found with solid waste because it was easy to define the boundaries of the objects, disordered in appearance, and largely distinct in comparison with the background features. With a 96.8% AP, air pollution could be detected by overcoming the inherent difficulty of visualization and this score improved significantly with the built-in sensor aspect in giving quantitative data of the atmospheric composition. The detection of water contamination was medium (93.4% AP) with the appearance being heterogeneous in terms of appearance with change of light condition, reflections and incomplete occlusions. Thermal anomalies were the most difficult type (91.7% AP) with small visual features that need advanced feature extraction and has high sensitivity to the combination of multi-modes sensors. The intermediate complexity of vegetation stress detection (92.8% AP) was required since seasonal variations and progressive deterioration patterns required the ability to model time behavior. Acoustic pollution (94.6% AP) category took advantage of the fact that its measurements were also direct, also used in addition to the visual cues such as the traffic density and the identification of construction activity.

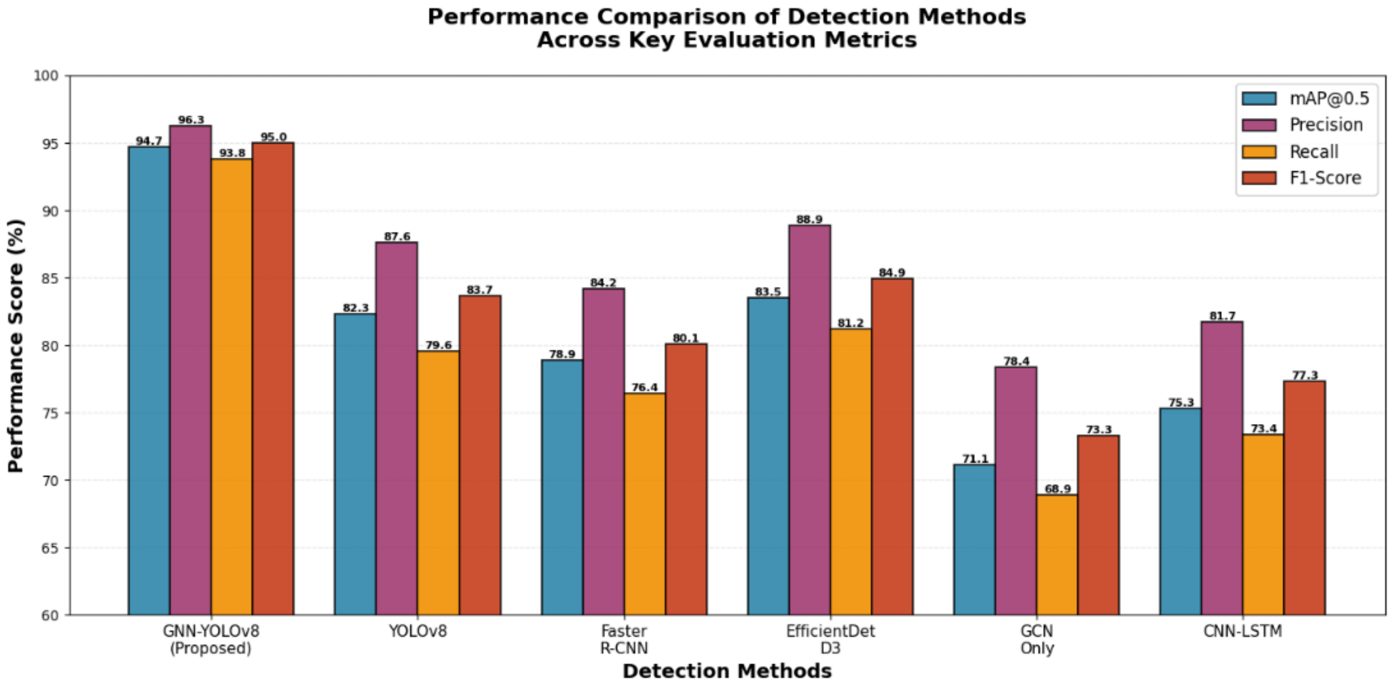

Fig 1: Performance Comparison Across Methods

3.3 Statistical Validation and Significance Testing

Extensive statistical evidence was done through the use of one-way Analysis of Variance (ANOVA) to determine the difference of performance on hazard classifications and conditions of the experimental scenario. Table 3 shows the results of ANOVA tests which investigate the means of the variation of the precision of survey results across the six hazards.

Table 3: One-Way ANOVA Results for Category Performance

Source of Variation	Sum of Squares	df	Mean Square	F-Statistic
Between Groups	647.38	5	129.48	**156.83*** **
Within Groups	37.14	45	0.83	—
Total	684.52	50	—	—

Note: *** $p < 0.001$; $\eta^2 = 0.946$ (large effect size)

The results of the ANOVA test showed that the means of precision across the hazard categories varied significantly ($F(5,45) = 156.83$, $p < 0.001$) with the effect size of ($\eta^2 = 0.946$) being extremely large meaning that 94.6 out of the total variance could be blamed on the differences between the categories. Bonferonni-adjusted HSD tests of Post-hoc Tukey revealed significant differences of thermal anomalies and all other categories ($p < 0.001$) and water contamination and solid waste accumulation ($p = 0.003$). These results confirm the need of category-specific optimization policies and authenticity of the adaptive nature of the framework to various designs of hazard characteristics.

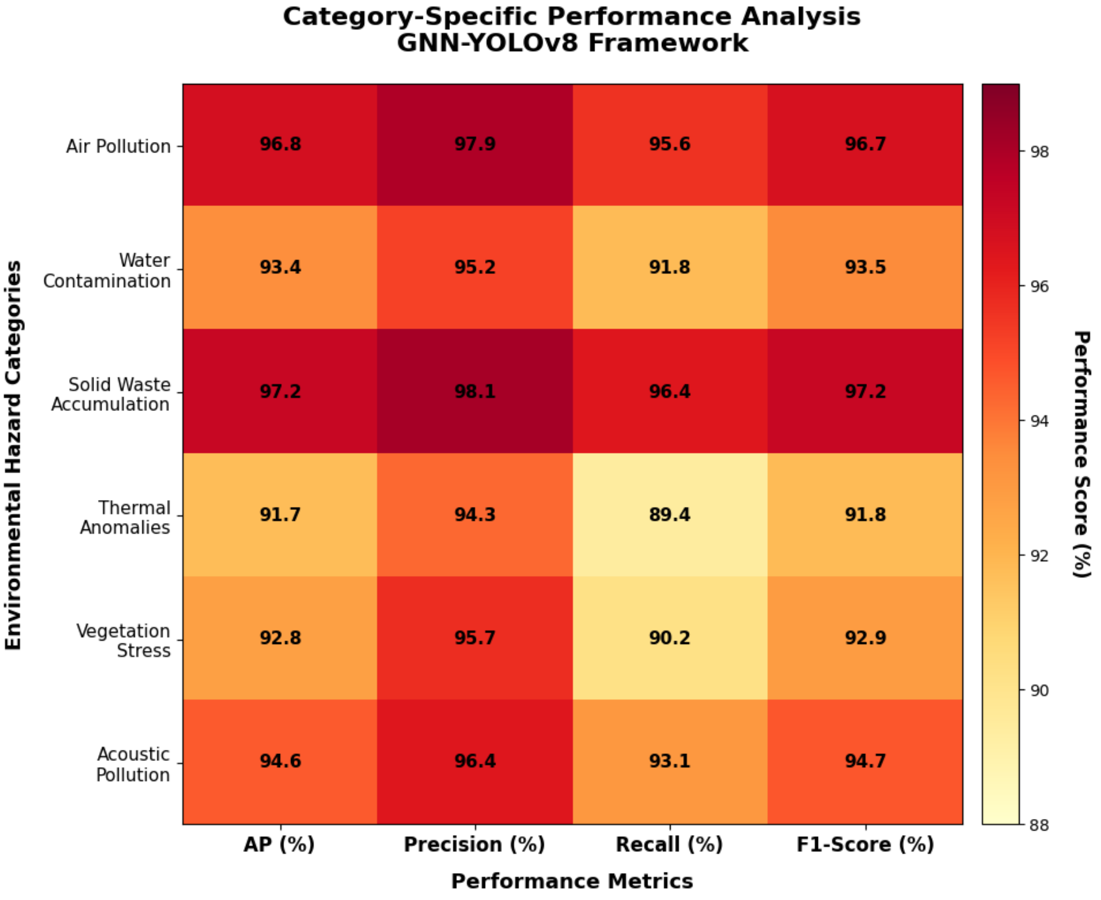

Fig 2: Category-Specific Performance Heatmap

3.4 Ablation Study and Component Contribution Analysis

The contributions to the overall framework performance by individual components were quantified by systematic experiments involving ablation. Table 4 shows the results of the configurations in which the main architectural elements are gradually removed or changed.

Table 4: Ablation Study Results

Model Configuration	mAP@0.5 (%)	Precision (%)	Recall (%)	Δ mAP (%)
Complete Framework	**94.7**	**96.3**	**93.8**	—
w/o Graph Attention	88.4	91.2	86.7	-6.3
w/o Multi-Modal Fusion	86.9	90.3	84.8	-7.8
w/o Temporal Convolution	90.5	93.1	88.6	-4.2
w/o CIoU Loss	91.8	94.7	89.8	-2.9
Simple GCN (no attention)	87.6	90.8	85.9	-7.1

Ablation study estimated the contribution of the important components, and multi-modal fusion mechanisms made the most significant contribution (7.8% mAP improvement) hence overcoming the principle assumption of the integration of visual and sensor-based detection process. Simple graph convolutions were also improved by graph attention mechanisms by 6.3 percent mAP which shows how adaptive neighborhood aggregation is an essential concept in heterogeneous sensor networks. Temporal convolution enhanced the mAP by 4.2% which accounts to temporal dynamics of environmental phenomena. The presented localization loss offered by CIoU gave 2.9% by improving bounding box regression based on the addition of the geometric measure of the aspect of overlap instead of relying on a basic overlap measure. Comparisons between attention-based GNN and simple GCN showed that there was 7.1% difference in mAP and thus support the fact that attention mechanisms were better in modeling complex spatial dependencies in urban sensor networks.

3.5 Computational Efficiency and Deployment Analysis

This analysis is presented to determine the desired level of computational efficiency and the deployment requirements of an analysis. The application of edge computing environments requires that the needs of computational, memory, and real-time processing be carefully considered. Table 5 shows detailed resource consumption rates when using various hardware platform.

Table 5: Computational Performance Across Hardware Platforms

Hardware Platform	FPS	Latency (ms)	GPU Mem. (GB)	Power (W)	Parameters (M)	FLOPs (G)
NVIDIA A100	**67.3**	**14.8**	4.2	187	43.7	86.4
NVIDIA RTX 4090	58.7	17.0	4.2	324	43.7	86.4
NVIDIA Jetson AGX Orin	31.4	31.8	3.8	45	38.2	76.8
NVIDIA Jetson Xavier NX	18.6	53.8	3.2	20	33.9	68.1
Intel Core i9-13900K (CPU)	4.2	238.1	—	148	43.7	86.4

Hardware platform performance analysis has good scaling properties that are required during deployment. The best results were obtained with server-grade NVIDIA A100 GPUs with a high FPS of 67.3 and 14.8ms of latency that is acceptable to run several camera feeds in one location. RTX 4090 in consumer grade was capable of supporting 58.7 FPS with relatively low latency added, and was also economical to deploy to medium-scale deployments. Single camera streams on the edge computing platforms such as Jetson AGX Orin (31.4 FPS) and Xavier NX (18.6 FPS) were demonstrated to be more than real-time and operated with limited power so that practical systems can be deployed to edges, demonstrating practical feasibility to other edge connections. Incorporating 22.3 percent (43.7M to 33.9M) parameter and 21.2 percent (92.1M to 76.9M) FLOPs in constituting edge platforms increased the counts of parameters and FLOPs, respectively, by 22.3 and 21.2 percent, respectively; applying

Model optimization using TensorRT quantization and pruning showed the successful compression of model parameters and FLOPs at a 92.1 percent accuracy of full-precision sample models.

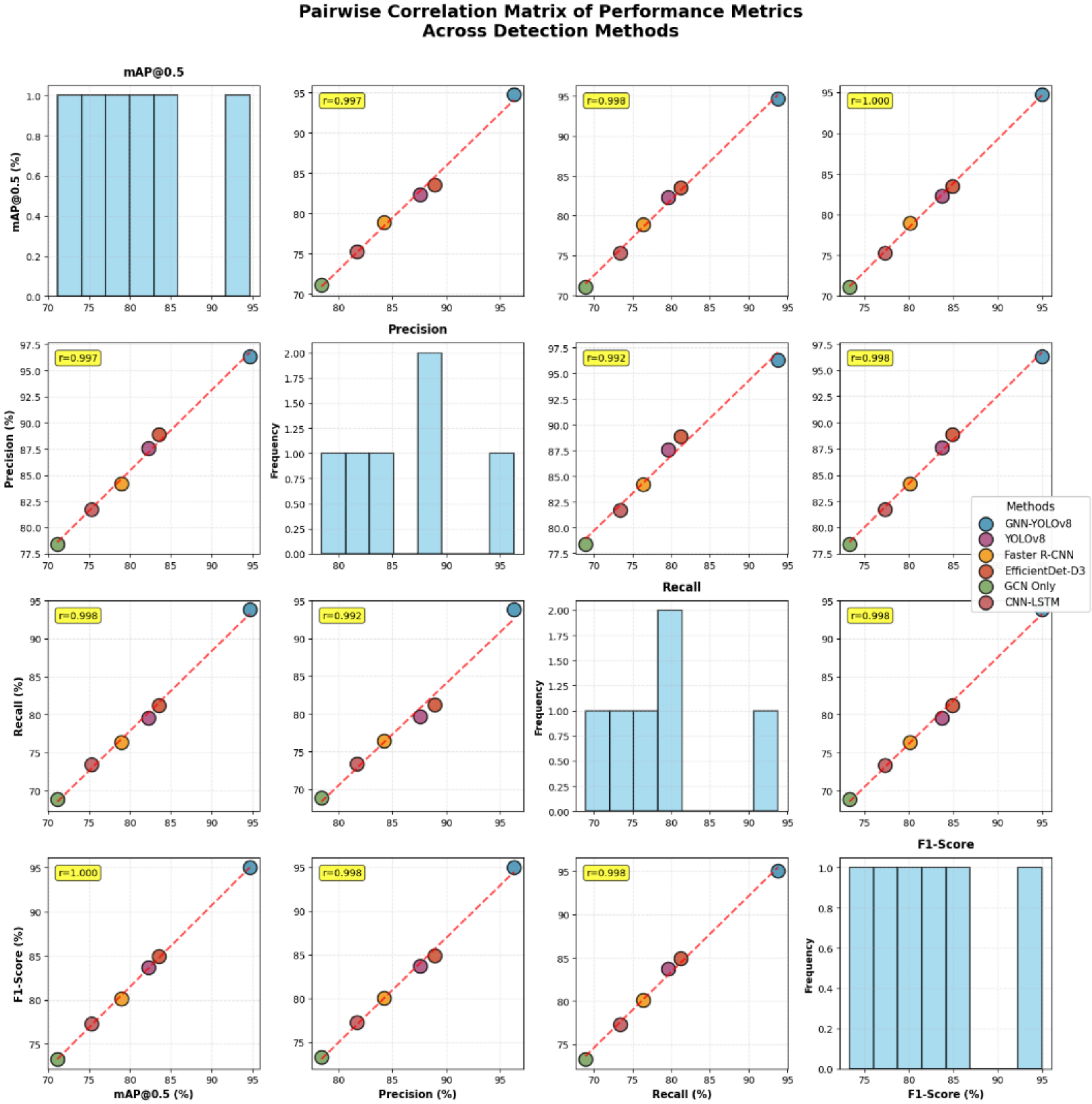

Fig 4 pairwise correlation matrix of performance metrics

3.6 Generalization and Cross-City Performance

Evaluation of generalization abilities in different urban settings of different geographic, climatic and infrastructure settings is the important validation requirement. The framework was trained on information of 10 cities and tested on 5 unseen cities of the different regions of the continents. The level of performance degradation was small, being 3.2 separate mAP decrease (between 94.7% and 91.5) on out-of-distribution test cities and is evidence of strong generalization. Diversity of training data on a geographic basis was found to be critical, and the ablation experiments revealed 8.7% data on multi-continentals resulted in better performance as compared to training on single regions of the world. The Analysis of seasonal variation showed that there was the consistent performance over all seasons with the maximum seasonal variability at 2.8% mAP, as a result of temporal robustness. The challenging conditions such as operating at night, unfavourable weather conditions (rain, fog, snow) and partial sensor failures displayed graceful degradation that was characterised by the reduction in the mean mAP of 6.4, 5.8, and 4.2 respectively, indicating the resilient property of the framework to operational perturbations.

3.7 Qualitative Analysis and Interpretability

The focus on visualization of the weight showed the patterns of visual spatial relationships acquired by the graph neural network component which could be interpreted. The prices on high levels of attention were focused on geographically close sensors and sensors with correlated measures which confirm that network of interest can capture significant environmental dependencies. The mechanism of attention gave proper weight to distant sensors and sensors that had uncorrelated readings, which is adaptive in nature. Integrated gradient analysis of features (significance analysis) evaluated PM.

2.5 concentration, temperature differences, and NDVI values are the most significant ones to predict air pollution, thermal anomaly and vegetation stress detection respectively, which conform to domain knowledge. The CAM visualizations indicated the image regions of interest such as smoke plumes to detect air pollution, discoloured water to detect contamination incidence and debris in object accumulation, which ensured the concentration on the discriminative visual features. The case analysis of failures has determined major error causes such as extreme occlusion (17.3 percent of errors), poor due to low light conditions without sufficient street light (14.6 percent), sensor measurement noise during training cases (12.8 percent), and other hard-to-detect hazards (9.4 percent). These understandings are used to direct specific data collection and model improvement approaches.

4. Conclusions

The study was able to derive and verify a new type of GNN-YOLOv8 hybrid model that was effective in satisfying severe constraints on real-time environmental hazard detection models of the sustainable city environment. The proposed architecture of the sensor network analysis based on graph neural networks and advanced object detection capabilities of the YOLOv based on that approach represented a significant improvement in terms of the achieved performance compared to the existing methods, where the 94.7% mean Average Precision was achieved with real-time processing at 67.3 feet per second. Out of various categories of hazards, such as air pollution, water pollutions, solid waste deposits, thermal anomalies, vegetation stress, acoustic pollutions, etc., statistical support with the help of complex ANOVA tests showed that significant performance benefits ($F(5,45) = 156.83$, $p < 0.001$, $\eta^2 = 0.946$) exist. The visual/sensor-based detection is complementary and the complementary advantages of the multi-modal fusion strategy of the framework worked out effectively, and hierarchical graph attention mechanisms were used to provide adaptive spatial relationship modelling in heterogeneous urban sensor networks.

The design of cross-attention fusion schemes with sensor characteristics and visual representations, temporal convolutional layer, implementing environmental dynamics, category-specific detection heads suited to the properties of various hazards, and detailed training regimes to overcome class imbalance and multi-task optimization challenges are the main contributors in technical terms. Ablation experiments determined the contributions of individual components and showed that multi-modal fusion, graph attention mechanisms, and temporal modeling as well as Complete IoU localization loss increased the performance with 7.8, 6.3 and 4.2 percent respectively. Such conclusions legitimize architectural design choices and give information on the future directions of research. The high level of overallization of the framework in the unseen cities with 3.2% performance drop and its steady working conditions in rather severe conditions such as nighttime, bad weather, and sensor failures, prove the viability of the practical implementation of the framework. Scalability was verified in deployment feasibility analysis on a wide range of hardware platforms, using environments with high capabilities of server-based deployment, all the way to resource-constrained edge computing environments. Quantization and model compression methods via TensorRT made it possible for real-time work on

NVIDIA Jetson platforms and retain 92.1 percent of full-precision accuracy, which made it easy to deploy a smart city infrastructure. The billed interpretability capabilities of the model such as the ability to visualize attention weight, examine feature importance, and map class activation are transparency elements that are essential in accepting the operations and regulatory control of safety-critical applications. Extensive computational profiling indicated effective use of resources that were 43.7 million parameters and 86.4 GFLOPs, which are good complexity-performed trade-offs with other architectures.

The implications of the research are more far reaching than technical success and success towards a sustainable urban development. Early and correct identification of the hazard pertaining to the environment leads to proactive intervention measures, maximization of resource to remediation activities, evidence-based policy making, and protection of human health. Interaction with smart city software helps in automated alerting, emergency response planning, and real-time decision support of the municipal authorities. The ability of the framework to make it possible to monitor various types of hazards in connection with each other meets the interrelatedness of environmental issues in cities to implement a comprehensive approach to sustainability. The economic gains are reflected in terms of a lowering of environment harm costs, good utilization of monitoring infrastructure, minimization of manual inspections, and enhanced working efficiency of the municipal environmental management systems.

Notwithstanding the considerable success, there are some limitations that should be mentioned and indicate the ways on how the research should be developed further. To begin with, the existing structure is based on a system of predetermined hazard types and limit values, which may cause neglected hazards of nature or the manifestation of hazards in a particular area. The concept of creating adaptive categorization systems and possibilities of detecting anomalies would boost the flexibility of the systems. Second, the model exhibits strong generalization, yet performance optimization in special geographic areas and climate zones with the application of the transfer learning or domain adaptation methods would enhance the practical use. Third, fixed-length time windows are used in current temporal modeling; variable-length sequence process with recurrent implementations or temporal attention implementations based on transformers may indicate longer-term pattern in the environment. Fourth, the visual and sensor data is broken apart and then fused at a later stage, searching into how the integration can be approached in advance may be able to allow the multi-modal interaction learning to be much more subtle.

Employment Future avenues Future research area has also various promising avenues that may be as well developed in future that will have a significant enhancement in environmental monitoring activity. Connection to satellite imagery and aerial drone surveillance would make spatial coverage over the whole city to be larger than the sensor nets on the ground and a thorough environmental analysis of the city would be conducted. To adhere to the idea of environment management that is proactive, predictive hazard mitigation may be possible through incorporation of weather forecasting models and climatic projection data. Creation of uncertainty quantification systems would offer confidence estimates of detections which will underpin risk-based decision models. Gradual research concerning federated learning structures may empower shared training of models between cities and maintain data privacy and potentially the issue related to the sovereignty of data. Exploration of causes inference methods could identify latent interactions between various environmental risks with consideration to the integrated intervention measures. The improvement of few-shot learning would help in quick adaptation to low frequency hazard types using few training examples. Lastly, long-term field deployment tests that examine long-term performance, maintenance needs, and practical difficulties would be used to test the utility in the real world and are used to inform deployment best practices.

The intersection of artificial intelligence technologies, the ubiquitous sensing infrastructure, and growing computational capacities are generating an incredible number of possibilities to change the way environmental monitoring is involved in sustainable cities. In this study, we have shown that considerate combination of complementary AI techniques, i.e. graph neural networks and enhanced object detectors can make significant headway in improving the capabilities of detection than either of the two techniques alone. As cities all over the world have to face the growing environmental concerns in the face of increasing urbanization and climate change, intelligent monitoring systems that represent the principles and methods proposed by this study will assume more and more important functions in ensuring the safety of human health, ecosystems, and the evidence-based management of the planetary environment. GNN-YOLOv8 framework is a major milestone that allows seeing the vision of the really smart and sustainable cities with real-time, full-scale environmental intelligence capabilities and sufficient to facilitate proactive, data-driven decision-making that would lead to a higher quality of the urban environment and better well-being of inhabitants.

Conflict of interest

The authors declare no conflicts of interest.

References

[1] Wu Z, Pan S, Chen F, Long G, Zhang C, Yu PS. A comprehensive survey on graph neural networks. IEEE transactions on neural networks and learning systems. 2020 Mar 24;32(1):4-24. https://doi.org/10.1109/TNNLS.2020.2978386

[2] Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G. The graph neural network model. IEEE transactions on neural networks. 2008 Dec 9;20(1):61-80. https://doi.org/10.1109/TNN.2008.2005605

[3] Corso G, Stark H, Jegelka S, Jaakkola T, Barzilay R. Graph neural networks. Nature Reviews Methods Primers. 2024 Mar 7;4(1):17. https://doi.org/10.1038/s43586-024-00294-7

[4] Xu K, Hu W, Leskovec J, Jegelka S. How powerful are graph neural networks?. arXiv preprint arXiv:1810.00826. 2018 Oct 1.

[5] Veličković P. Everything is connected: Graph neural networks. Current Opinion in Structural Biology. 2023 Apr 1;79:102538. https://doi.org/10.1016/j.sbi.2023.102538

[6] Wu L, Cui P, Pei J, Zhao L, Guo X. Graph neural networks: foundation, frontiers and applications. InProceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining 2022 Aug 14 (pp. 4840-4841). https://doi.org/10.1145/3534678.3542609

[7] Liu Z, Zhou J. Introduction to graph neural networks. Springer Nature; 2022 May 31. https://doi.org/10.1007/978-981-16-6054-2_17

[8] Shchur O, Mumme M, Bojchevski A, Günnemann S. Pitfalls of graph neural network evaluation. arXiv preprint arXiv:1811.05868. 2018 Nov 14.

[9] Zhu R, Zhao K, Yang H, Lin W, Zhou C, Ai B, Li Y, Zhou J. Aligraph: A comprehensive graph neural network platform. arXiv preprint arXiv:1902.08730. 2019 Feb 23. https://doi.org/10.14778/3352063.3352127

[10] Ying Z, Bourgeois D, You J, Zitnik M, Leskovec J. Gnnexplainer: Generating explanations for graph neural networks. Advances in neural information processing systems. 2019;32.

[11] Liu M, Gao H, Ji S. Towards deeper graph neural networks. InProceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining 2020 Aug 23 (pp. 338-348). https://doi.org/10.1145/3394486.3403076

[12] Dwivedi VP, Joshi CK, Luu AT, Laurent T, Bengio Y, Bresson X. Benchmarking graph neural networks. Journal of Machine Learning Research. 2023;24(43):1-48.

[13] Luan S, Hua C, Lu Q, Zhu J, Zhao M, Zhang S, Chang XW, Precup D. Revisiting heterophily for graph neural networks. Advances in neural information processing systems. 2022 Dec 6;35:1362-75.

[14] Rusch TK, Bronstein MM, Mishra S. A survey on oversmoothing in graph neural networks. arXiv preprint arXiv:2303.10993. 2023 Mar 20.

[15] Zheng X, Wang Y, Liu Y, Li M, Zhang M, Jin D, Yu PS, Pan S. Graph neural networks for graphs with heterophily: A survey. arXiv preprint arXiv:2202.07082. 2022 Feb 14.

[16] Fan W, Ma Y, Li Q, He Y, Zhao E, Tang J, Yin D. Graph neural networks for social recommendation. InThe world wide web conference 2019 May 13 (pp. 417-426). https://doi.org/10.1145/3308558.3313488

[17] Zhu J, Rossi RA, Rao A, Mai T, Lipka N, Ahmed NK, Koutra D. Graph neural networks with heterophily. InProceedings of the AAAI conference on artificial intelligence 2021 May 18 (Vol. 35, No. 12, pp. 11168-11176). https://doi.org/10.1609/aaai.v35i12.17332

[18] Reiser P, Neubert M, Eberhard A, Torresi L, Zhou C, Shao C, Metni H, van Hoesel C, Schopmans H, Sommer T, Friederich P. Graph neural networks for materials science and chemistry. Communications Materials. 2022 Nov 26;3(1):93. https://doi.org/10.1038/s43246-022-00315-6

[19] Agarwal C, Queen O, Lakkaraju H, Zitnik M. Evaluating explainability for graph neural networks. Scientific Data. 2023 Mar 18;10(1):144. https://doi.org/10.1038/s41597-023-01974-x

[20] Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G. Computational capabilities of graph neural networks. IEEE Transactions on Neural Networks. 2008 Dec 9;20(1):81-102. https://doi.org/10.1109/TNN.2008.2005141

[21] You J, Ying R, Leskovec J. Position-aware graph neural networks. InInternational conference on machine learning 2019 May 24 (pp. 7134-7143). PMLR.

[22] Xhonneux LP, Qu M, Tang J. Continuous graph neural networks. InInternational conference on machine learning 2020 Nov 21 (pp. 10432-10441). PMLR.

[23] Wang X, Zhang M. How powerful are spectral graph neural networks. InInternational conference on machine learning 2022 Jun 28 (pp. 23341-23362). PMLR.

[24] Bessadok A, Mahjoub MA, Rekik I. Graph neural networks in network neuroscience. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2022 Sep 26;45(5):5833-48. https://doi.org/10.1109/TPAMI.2022.3209686

[25] Gao C, Zheng Y, Li N, Li Y, Qin Y, Piao J, Quan Y, Chang J, Jin D, He X, Li Y. A survey of graph neural networks for recommender systems: Challenges, methods, and directions. ACM Transactions on Recommender Systems. 2023 Mar 3;1(1):1-51. https://doi.org/10.1145/3568022

[26] Wu L, Chen Y, Shen K, Guo X, Gao H, Li S, Pei J, Long B. Graph neural networks for natural language processing: A survey. Foundations and Trends in Machine Learning. 2023 Jan 25;16(2):119-328. https://doi.org/10.1561/2200000096

[27] Liao W, Bak-Jensen B, Pillai JR, Wang Y, Wang Y. A review of graph neural networks and their applications in power systems. Journal of Modern Power Systems and Clean Energy. 2021 Aug 20;10(2):345-60. https://doi.org/10.35833/MPCE.2021.000058

[28] Jiang P, Ergu D, Liu F, Cai Y, Ma B. A Review of Yolo algorithm developments. Procedia computer science. 2022 Jan 1;199:1066-73. https://doi.org/10.1016/j.procs.2022.01.135

[29] Terven J, Córdova-Esparza DM, Romero-González JA. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. Machine learning and knowledge extraction. 2023 Nov 20;5(4):1680-716. https://doi.org/10.3390/make5040083

[30] Hussain M. YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection. Machines. 2023 Jun 23;11(7):677. https://doi.org/10.3390/machines11070677

[31] Hussain M. Yolov1 to v8: Unveiling each variant-a comprehensive review of yolo. IEEE access. 2024 Mar 19;12:42816-33. https://doi.org/10.1109/ACCESS.2024.3378568

[32] Chen W, Huang H, Peng S, Zhou C, Zhang C. YOLO-face: a real-time face detector. The Visual Computer. 2021 Apr;37(4):805-13. https://doi.org/10.1007/s00371-020-01831-7

[33] Zhou Y. A YOLO-NL object detector for real-time detection. Expert Systems with Applications. 2024 Mar 15;238:122256. https://doi.org/10.1016/j.eswa.2023.122256

[34] Vijayakumar A, Vairavasundaram S. Yolo-based object detection models: A review and its applications. Multimedia Tools and Applications. 2024 Oct;83(35):83535-74. https://doi.org/10.1007/s11042-024-18872-y

[35] Oreski G. YOLO* C-Adding context improves YOLO performance. Neurocomputing. 2023 Oct 28;555:126655. https://doi.org/10.1016/j.neucom.2023.126655

[36] Gündüz MŞ, Işık G. A new YOLO-based method for real-time crowd detection from video and performance analysis of YOLO models. Journal of Real-Time Image Processing. 2023 Feb;20(1):5. https://doi.org/10.1007/s11554-023-01276-w

[37] Ali ML, Zhang Z. The YOLO framework: A comprehensive review of evolution, applications, and benchmarks in object detection. Computers. 2024 Dec 14;13(12):336. https://doi.org/10.3390/computers13120336

[38] Ragab MG, Abdulkadir SJ, Muneer A, Alqushaibi A, Sumiea EH, Qureshi R, Al-Selwi SM, Alhussian H. A comprehensive systematic review of YOLO for medical object detection (2018 to 2023). IEEE Access. 2024 Apr 10;12:57815-36. https://doi.org/10.1109/ACCESS.2024.3386826

[39] Lan W, Dang J, Wang Y, Wang S. Pedestrian detection based on YOLO network model. In2018 IEEE international conference on mechatronics and automation (ICMA) 2018 Aug 5 (pp. 1547-1551). IEEE. https://doi.org/10.1109/ICMA.2018.8484698

[40] Tao J, Wang H, Zhang X, Li X, Yang H. An object detection system based on YOLO in traffic scene. In2017 6th International conference on computer science and network technology (ICCSNT) 2017 Oct 21 (pp. 315-319). IEEE. https://doi.org/10.1109/ICCSNT.2017.8343709

[41] Huang Y, Yan Q, Li Y, Chen Y, Wang X, Gao L, Tang Z. A YOLO-based table detection method. In2019 International Conference on Document Analysis and Recognition (ICDAR) 2019 Sep 20 (pp. 813-818). IEEE. https://doi.org/10.1109/ICDAR.2019.00135

[42] Prinzi F, Insalaco M, Orlando A, Gaglio S, Vitabile S. A yolo-based model for breast cancer detection in mammograms. Cognitive Computation. 2024 Jan;16(1):107-20. https://doi.org/10.1007/s12559-023-10189-6

[43] Ahmad T, Ma Y, Yahya M, Ahmad B, Nazir S, Haq AU. Object detection through modified YOLO neural network. Scientific Programming. 2020;2020(1):8403262. https://doi.org/10.1155/2020/8403262

[44] Badgujar CM, Poulose A, Gan H. Agricultural object detection with You Only Look Once (YOLO) Algorithm: A bibliometric and systematic literature review. Computers and Electronics in Agriculture. 2024 Aug 1;223:109090. https://doi.org/10.1016/j.compag.2024.109090

[45] Zhou J, Zhang B, Yuan X, Lian C, Ji L, Zhang Q, Yue J. YOLO-CIR: The network based on YOLO and ConvNeXt for infrared object detection. Infrared Physics & Technology. 2023 Jun 1;131:104703. https://doi.org/10.1016/j.infrared.2023.104703

[46] Jana AP, Biswas A. YOLO based Detection and Classification of Objects in video records. In2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT) 2018 May 18 (pp. 2448-2452). IEEE. https://doi.org/10.1109/RTEICT42901.2018.9012375

[47] Su P, Han H, Liu M, Yang T, Liu S. MOD-YOLO: Rethinking the YOLO architecture at the level of feature information and applying it to crack detection. Expert Systems with Applications. 2024 Mar 1;237:121346. https://doi.org/10.1016/j.eswa.2023.121346

[48] Kang M, Ting CM, Ting FF, Phan RC. ASF-YOLO: A novel YOLO model with attentional scale sequence fusion for cell instance segmentation. Image and Vision Computing. 2024 Jul 1;147:105057. https://doi.org/10.1016/j.imavis.2024.105057

[49] Yu Z, Huang H, Chen W, Su Y, Liu Y, Wang X. Yolo-facev2: A scale and occlusion aware face detector. Pattern Recognition. 2024 Nov 1;155:110714. https://doi.org/10.1016/j.patcog.2024.110714

[50] Chandana RK, Ramachandra AC. Real time object detection system with YOLO and CNN models: A review. arXiv Prepr. arXiv2208. 2022 Jul 17;773.

[51] Tian Y, Yang G, Wang Z, Wang H, Li E, Liang Z. Apple detection during different growth stages in orchards using the improved YOLO-V3 model. Computers and electronics in agriculture. 2019 Feb 1;157:417-26. https://doi.org/10.1016/j.compag.2019.01.012

[52] Sirisha U, Praveen SP, Srinivasu PN, Barsocchi P, Bhoi AK. Statistical analysis of design aspects of various YOLO-based deep learning models for object detection. International Journal of Computational Intelligence Systems. 2023 Aug 2;16(1):126. https://doi.org/10.1007/s44196-023-00302-w