Svedberg Open
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Advanced Computer Vision Techniques Using Graph Neural Networks for Real-Time Object Detection and Scene Understanding

Mridul Dixit[1], Prasanth Varasala[2], Malarvizhi S[3], Regulwar Ganesh Bhaiyya[4], Dr. G. Sanjiv Rao[5], Chandrashekhar Ramesh Ramtirthkar[6], Piyush Pal[7], Mahendran Arumugam [8]

[1]Department of Computer Engineering & Applications, GLA University, Mathura, Email: mridul.dixit@gla.ac.in
[2]Associate Professor, Department of Electronics and Communication Engineering, Pragati Engineering College, ADB Road, Surampalem, Near Peddapuram, Kakinada District, Andhra Pradesh, India - 533437. Email: varasalaprasanth@gmail.com
[3]Assistant Professor, Department of Commerce, Meenakshi College of Arts and Science, Meenakshi Academy of Higher Education and Research, Email: malarvizhicom@maher.ac.in
[4]Associate Professor, Department of Information Technology, Vardhaman College of Engineering, Shamshabad, Hyderabad, India - 501 218, Email: ganeshregulwar@vardhaman.org
[5]Professor, Department of Artificial Intelligence and Machine Learning, Aditya University, Surampalem, Andhra Pradesh, Pin 533437, Email: dean_cd@adityauniversity.in
[6]Associate Professor, Mechanical Engineering, Vishwakarma Institute of Technology, Pune, Maharashtra, 411037,India. Email: chandrashekhar.ramtirthkar@vit.edu
[7]School of Engineering &Technology,Noida international University, Uttar Pradesh 203201, India, Email: piyush.pal@niu.edu.in
[8]Center for Global Health Research,Saveetha Medical College, Saveetha Institute of Medical and Technical Sciences, Chennai, India. Email: mahendrana.sdc@saveetha.com

Abstract

Real-time object recognition and scene perception are central to the higher level computer vision operationalities in autonomous driving, smart surveillance, robotics, and smart healthcare systems. Nevertheless, traditional convolution-based object detecting models are mainly concerned with single object detection and do not typically work well in obtaining contextual and spatial associations among objects in complex scenes. Such a restriction diminishes the accuracy of semantic understanding as well as the reliability of decision-making in changing real-world situations. To counter this difficulty, this paper presents a novel computer vision structure that is more advanced with the implementation of the Graph Neural Network (GNN) to detect objects and comprehend scenes in real-time. In the proposed model, the lightweight YOLO-based backbone feature extractor is paired with a Graph Attention Network (GAT) to predict inter-object relationships and context scene relationships. It has a scene graph generation mechanism to enhance semantic reasoning and spatial interaction analysis between detected objects. The framework was tested on benchmark datasets such as COCO and Visual Genome with real-time conditions. The experimental findings support that the proposed method obtained an average Precision (mAP) of 91.3, detection accuracy of 94.1, scene relationship recognition accuracy of 92.6, and a detection rate of 42 FPS, better than the traditional CNN-based and transformer-based detection frameworks and with low computational latency in real-time implementation.

Keywords: Computer Vision; Graph Neural Networks (GNN); Real-Time Object Detection; Scene Understanding; Graph Attention Networks (GAT); Scene Graph Generation; Deep Learning; Contextual Reasoning; YOLO-Based Detection; Spatial Relationship Modeling; Intelligent Vision Systems; Semantic Scene Analysis.

1. Introduction

Computer vision is now a significant field of artificial intelligence research, allowing machines to process and comprehend visual data of images and videos (Diwan et al., 2023). The latest developments of deep learning have enabled computer vision applications to be highly enhanced in autonomous vehicles, smart surveillance, robots, health monitoring, and automated transportation (Lohani et al., 2022). Two of these uses include real-

time object detection and scene understanding which are important in facilitating intelligent systems to communicate effectively with dynamic environments.

Real-time object detection is concerned with detecting and localizing objects in images or video frames and at low computational latency. Faster R-CNN, SSD, and YOLO deep learning models have high detection accuracy and speed (Mahendru and Dubey, 2021). Nevertheless, the majority of traditional convolutional neural network (CNN)-based detectors process objects in isolation and are commonly incapable of taking into account the semantic and contextual object-object relationships in more complex scenes.

Scene processing goes beyond the detection of objects to examine the spatial relationship and interaction of objects. Proper contextual reasoning is relevant in many related applications, e.g., autonomous driving and surveillance, where object relationships play a role in decision-making (Wang et al., 2019). The current detection models are insufficient in modeling these inter-object dependencies and this minimizes the performance of scene interpretation in complex environments.

Graph Neural Networks (GNNs) offer a powerful approach to the tasks of representing relational information and using them to model objects as graph nodes and object interactions as edges (Atwood and Towsley, 2016). GNNs are able to acquire contextual dependencies and enhance semantic meaning through message-passing processes (Chami et al., 2022). This paper suggests an innovative framework of GNN-based incremental real-time object detection and scene understanding to overcome the shortcomings of traditional detectors. The proposed architecture integrates a light-weight YOLO-based network of feature extraction with a Graph Attention Network (GAT) block to enhance contextualized reasoning and spatial relationship analysis with retaining the capability of real-time processing.

Assessment of the proposed framework is performed on benchmark datasets and compared with the traditional CNN-based and transformer-based methods based on parameters like the mean Average Precision (mAP), precision, TTR, and inference speed. The research will utilize an effective and smart vision system that can detect objects accurately and understand the scenery better to facilitate the creation of autonomous systems in the next generation.

2. Related Work

The recent developments in deep learning and graph based learning architecture have contributed to the development of the world in terms of object detection and scene understanding architecture. Graph Neural Networks (GNNs) have become good contextual reasoning and relational learning models. Atwood and Towsley (2016) proposed diffusion-convolutional neural networks to propagate features in the graph, and Bastings et al. (2017) suggested learning contextual representations using graph convolutional encoders. Ferludin et al. (2022) and Chami et al. (2022) also emphasized that learning graph representation and scalable GNN implementations are crucial to the current artificial intelligence applications.

Mechanisms of attention have also enhanced contextual feature extraction and semantic reasoning. The attention-based learning models proposed by Bahdanau et al. (2014) inspired the creation of Graph Attention Networks (GATs). Also, two studies by Allamanis et al. (2017) and Bank et al. (2023) showed the usefulness of learning the graph representation and feature embedding methods on complex structured data.

In real-time object detection, the use of the YOLO-based architectures has been very successful as the models are fast and accurate. Diwan et al. (2023) surveyed the history of YOLO architectures and their use in intelligent vision systems. Mahendru and Dubey (2021), Chandan and coworkers (2018), and Yu and colleagues (2018) created open-source models of real-time object detection and tracking based on YOLO and OpenCV. Moreover, Lohani et al. (2022), Masood et al. (2022), and Wang et al. (2019) showed the significance of the smart object tracking and surveillance systems in dynamic settings.

Despite the high accuracy of object detection in current approaches, the majority of conventional CNN-based models were mainly concerned with the independent recognition of objects without a strong consideration of semantic relationships between objects. Thus, this paper suggests a YOLOv8 framework augmented with GNNs to enhance the contextual reasoning and semantic scene perception of real-time intelligent vision systems.

3. Proposed Methodology

The suggested framework combines a minimal real-time object-detecting framework with a contextual reasoning based on Graph Neural Network (GNN) to understand the scene better. The system involves a single pipeline that deals with YOLOv8-based feature extraction, graph construction, graph attention learning, and semantic relationship analysis. The provided methodology is more effective at object detection and contextual scene understanding and has low computational latency, which is appropriate to be implemented in real-time.

3.1 Overall Framework Architecture

The given system comprises four key steps, including image acquisition and preprocessing, feature extraction and object detection, graph construction and GNN reasoning, and scene understanding with output generation. As shown in Figure 1, the proposed framework comprises the YOLOv8-based feature extraction and Graph Attention Network (GAT)-based contextual reasoning to effectively detect objects in real-time and make sense of the semantics of a specific scene. To begin with, the input images or video frames are obtained in benchmark datasets like COCO and Visual Genome. Preprocessing step is resizing, normalization, augmentation and noise reduction, to enhance quality of input and model generalization. To normalize pixels, the following is used:

$$I_{norm} = \frac{I - \mu}{\sigma}$$

Where I_{norm} represents normalized image pixels, I denotes original pixel values,μ is the mean pixel intensity, σ is the standard deviation.The obtained processed images are fed to the YOLOv8 based on CSPDarkNet architecture to extract features in real time and locate objects. Multi-scale feature maps are extracted by the backbone network to extract spatial and semantic features, which enhance the detection of objects of different sizes and complex visual feature. The process of extracting features can be illustrated as:

$$F = \phi(I)$$

Where I is the input image, $\phi(\cdot)$ denotes the backbone feature extraction function, F represents extracted feature maps. The detection module predicts object bounding boxes represented as:

$$B_i = (x_i, y_i, w_i, h_i, c_i)$$

Where (x_i, y_i) denote object center coordinates,w_i,h_i represent width and height,c_i is the confidence score. The quality of bounding box prediction is evaluated using the Intersection over Union (IoU) metric:

$$I_O U = \frac{Area(B_p \cap B_{gt})}{Area(B_p \cup B_{gt})}$$

Where B_p represents the predicted bounding box generated by the proposed object detection model, and B_{gt} denotes the corresponding ground-truth bounding box obtained from the annotated dataset.

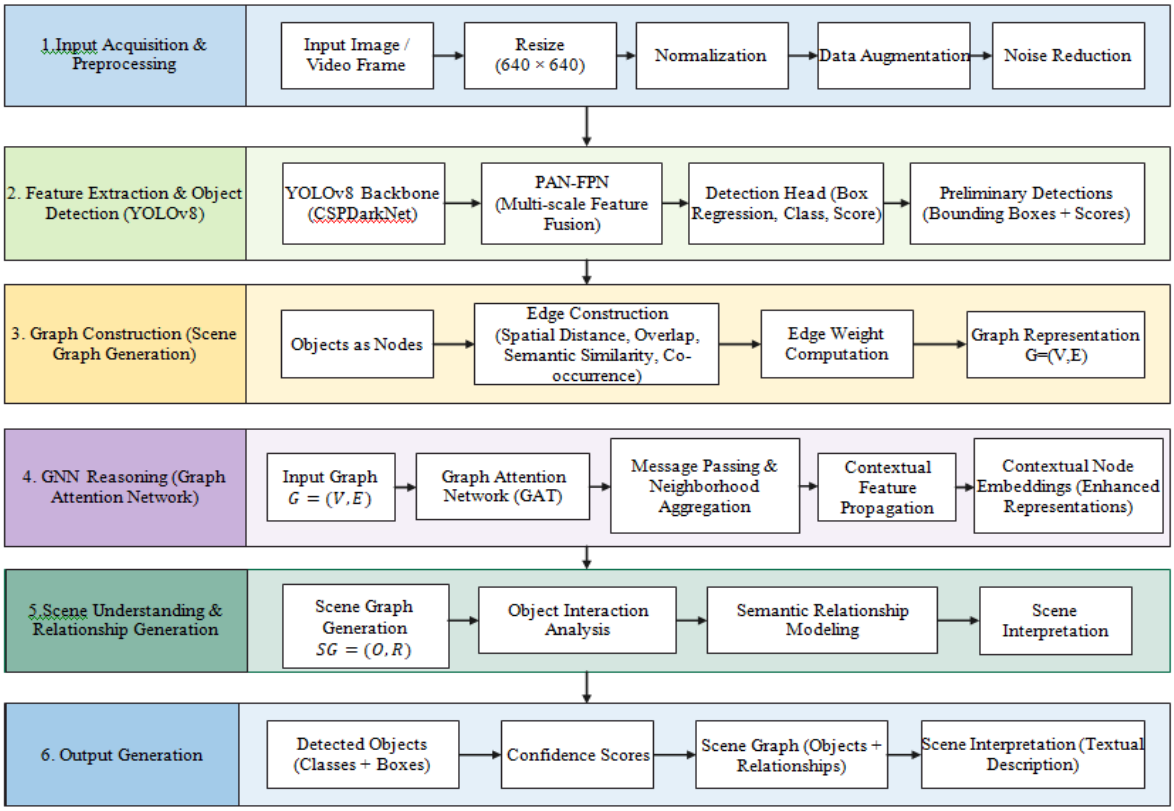

Fig 1:Framework Architecture Diagram

3.2 Graph Construction and GNN-Based Contextual Reasoning

Upon initial identification of the objects, the identified objects are converted into a graph format to facilitate the contextual reasoning and semantic interaction analysis. The graph is represented as:

$$G = (V, E)$$

where V denotes graph nodes and E denotes graph edges. Each detected object is represented as a graph node containing spatial and semantic information:

$$vi = [f_i, x_i, y_i, w_i, h_i]$$

where f_i represents object feature embedding, (x_i, y_i) denote object spatial coordinates, and w_i, h_i represent bounding box dimensions. Graph edges are established based on spatial distance, semantic similarity, object overlap, and co-occurrence relationships between detected objects. The edge weight between two nodes is computed as:

$$e_{ij} = \alpha S_{ij} + \beta D_{ij} + \gamma O_{ij}$$

where S_{ij} represents semantic similarity, D_{ij} denotes spatial distance,O_{ij} indicates overlap ratio, and α, β, γ are weighting coefficients controlling the contribution of each relational factor. The graph that is created is fed through a Graph Attention Network (GAT) to propagate contextual features and neighborhood aggregation. The updating mechanism of the node is defined as:

$$h_i^{(l+1)} = \sigma\left(\sum_{j \in N(i)} \alpha_{ij} W h_j^{(l)}\right)$$

where h_i denotes node features at layer$l$, $N(i)$ represents neighboring nodes, W is the trainable weight matrix, α_{ij} denotes attention coefficients, and $\sigma(\cdot)$represents the activation function. The attention coefficients are calculated using:

$$\alpha_{ij} = \frac{\exp\left(\text{LeakyReLU}\left(a^T\left[Wh_i \parallel Wh_j\right]\right)\right)}{\sum_{k\in N(i)} \exp\left(\text{LeakyReLU}\left(a^T\left[Wh_i \parallel Wh_j\right]\right)\right)}$$

where a represents the attention parameter vector and || denotes vector concatenation. GAT module allows contextual reasoning, semantic interaction modeling, aggregation of neighborhoods and understanding of features at scene level. The framework enhances the detection model in terms of understanding the complicated environments and contextual relationships of the objects by spreading information among similar objects.

3.3 Scene Understanding and Algorithm Flow

The scene understanding module produces semantic relations between the perceived objects based on contextual graph embeddings created by the GAT module. The graph of the scene can be expressed as:

$$SG = (O, R)$$

where O indicates objects detected and R indicates semantic relationships between objects in the scene. The resulting scene graph allows semantic interpretation of object interactions including: person riding bicycle, car near traffic signal and person holding mobile phone. This contextual explanation advances environmental knowledge and the general cleverness of the vision scheme.

The general flow of the proposed framework is the input image or video frames are acquired, then undergo preprocessing steps such as resizing, normalization, augmentation, and noise filtering. The already preprocessed images are sent to the YOLOv8 backbone to extract multi-scale features and get an initial object detection. The objects identified are then converted into graph nodes and the relationships between the objects in the context are represented as graph edges. The built graph is fed into the Graph Attention Network to generate feature propagation and semantic reasoning. Lastly, the system produces object classifications, bounding boxes, confidence scores and prediction of scenes relationships to give the final scene understanding result. Mean Average Precision (mAP), Precision, Recall, Frames Per Second (FPS) and scene relationship recognition accuracy are used to assess the performance of the proposed framework. The framework aims to attain excellent detection and contextual knowledge and support real-time inference, which is suitable to autonomous and intelligent vision applications.

4. Experimental Setup

The experimental system aimed to consider the efficiency of the suggested Graph Neural Network (GNN)-based real-time object detection and scene understanding model in the conditions of real computer vision. The framework was trained and evaluated on big-scale benchmark dataset and assessed with standard object detection and scene understanding metrics. The experiments were aimed at examining accuracy of detection, capacity to reason with context and speed of inference, and efficiency of computations.

4.1 Dataset Description

The suggested framework was tested on the COCO, Visual Genome, Pascal VOC, and Open Images benchmark sets of real-time object detection and scene understanding tasks. COCO has more than 330,000 images and 80 object classes and about 1.5 million annotated instances that support the multi-object detection, and this is in complex environments. The Visual Genome dataset consists of over 108,000 images where objects are densely annotated and semantic relationship triplets generated to support scene graph generation, as well as contextual reasoning. Pascal VOC has about 11,500 images forming 20 object categories to be used when analysing object localization and classification, whilst the Open Images dataset served to enhance the robustness and generalization of models. Each dataset was split into training, validation and testing sets 80:10:10. This was done by resizing input images to 640 by 640 followed by training and inference. Horizontal flipping, random cropping, rotation, and brightness adjustment data augmentation methods have been used to enhance the diversity of features and minimize overfitting.

4.2 Hardware and Software Configuration

The framework suggested was implemented on the PyTorch deep learning framework with CUDA-enabled GPU acceleration. Experiments were run on a high-performance workstation with NVIDIA RTX 4090 card with 24 GB VRAM to support the accelerated parallel computation. The system made use of an Intel Core i9 processor and 64 GB RAM to facilitate the processing of large volumes of data and real-time inference processes.

Python 3.11, PyTorch 2.0, CUDA 12.1 and cuDNN acceleration libraries, OpenCV to work with images and NetworkX to generate and manipulate graphs were the software environment. The experimental platform was based on Ubuntu 22.04 LTS operating system because it has the best set of GPU compatibility and support of deep learning frameworks.

The Graph Attention Network (GAT) module was implemented with the PyTorch Geometric library to effectively perform graph construction, neighborhood aggregation, and message-passing functions. Under the condition of the implementation of the GPU inference, real-time performance evaluation was applied to analyze the latency and frame processing capacity.

4.3 Training Parameters

Supervised learning with stochastic gradient optimization was used to train the proposed framework. The YOLOv8 backbone, and Graph Attention Network modules were trained together using an end-to-end. A batch size of 16 and an initial learning rate of 1×10^{-4} were used to perform the training process. Adam optimizer was used because it has a rapid convergence and adaptive learning rate. The incorporation of the weight update is shown as:

$$\theta_{t+1} = \theta_t - \eta \nabla \mathrm{L}(\theta_t)$$

where θ_t represents the model parameters at iteration t, η denotes the learning rate, and $\nabla \mathrm{L}(\theta_t)$ represents the gradient of the loss function. The model was optimized to 150 epochs with cosine learning-rate scheduling to enhance convergence stability. To classify objects and regress their bounding box, binary cross-entropy and Complete Intersection over Union (CIoU) losses were used, respectively. The overall loss function is represented as:

$$L_{total} = \lambda_1 L_{cls} + \lambda_2 L_{\mathrm{bbox}} + \lambda_3 L_{rel}$$

where L_{cls} denotes classification loss, L_{bbox} represents bounding box regression loss, L_{rel}l corresponds to relationship prediction loss, and $\lambda_1, \lambda_2, \lambda_3$have the coefficients of weighting of the contribution of each part of the loss. The dropout regularization and batch normalization were added to enhance the generalization performance and minimize overfitting. To avoid needless training cycles and maximize computational efficiency early stopping criteria were used.

4.4 Evaluation Metrics

It was tested on common metrics of computer vision and object detection, such as Accuracy, Precision, Recall, F1-score, mean Average Precision (mAP), Intersection over Union (IoU), Frames Per Second (FPS), and inference latency, to measure the performance of the proposed framework. Precision is the fraction of the number of objects correctly identified of all the ones predicted and it is calculated as:

$$\text{Precision} = \frac{TP}{TP + FP}$$

where TP denotes true positives and FP represents false positives. Recall evaluates the proportion of correctly detected objects among all ground-truth objects and is expressed as:

$$\text{Recall} = \frac{TP}{TP + FN}$$

where FN represents false negatives. The F1-score combines Precision and Recall into a single metric and is computed as:

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Overall object detection performance was assessed based on the mean Average Precision (mAP) metric, which was done regardless of the individual classes. Measuring localization between predicted and ground-truth bounding boxes was done using the Intersection over Union (IoU) measure. Frames Per Second (FPS) was used to assess the real-time processing capability, indicating how many frames an image can be processed in a second during inference. Latency of inference was also studied to establish the efficiency and applicability of the proposed framework to real time application in intelligent vision applications like autonomous driving, robotics and smart surveillance systems.

5. Results and Discussion

5.1 Object Detection Performance

Mean Average Precision (mAP), Precision, Recall, and Frames Per Second (FPS) were used to measure the object detection performance of the proposed framework. The suggested GNN-based architecture was tested on par with YOLOv8, Faster R-CNN, and Vision Transformer-based detectors in the same experiment.

Table 1: Object Detection Performance Comparison

Model	mAP (%)	Precision (%)	Recall (%)	FPS
Faster R-CNN	84.6	86.1	82.7	18.4
YOLOv8	89.2	90.4	88.1	46.8
Vision Transformer	90.1	91.2	89.5	31.7
Proposed GNN Framework	91.3	94.1	92.4	42.0

The suggested GNN framework attained the best object detection performance with the highest mAP of 91.3%, Precision of 94.1 and Recall of 92.4 as used in Table 1. The combination of graph-based contextual reasoning largely led to the enhancement of the object localization and minimization of false detection in complex scenes. Even though YOLOv8 had the highest FPS of 46.8, the proposed structure realized real-time capability of processing with 42 FPS but had a better detection accuracy and contextual understanding. The proposed framework, as shown in Figure 2, was always superior to Faster R-CNN, YOLOv8, and Vision Transformer models in terms of mAP, Precision, and Recall values, which proves the efficiency of graph-based semantic reasoning in enhancing the object detection performance.

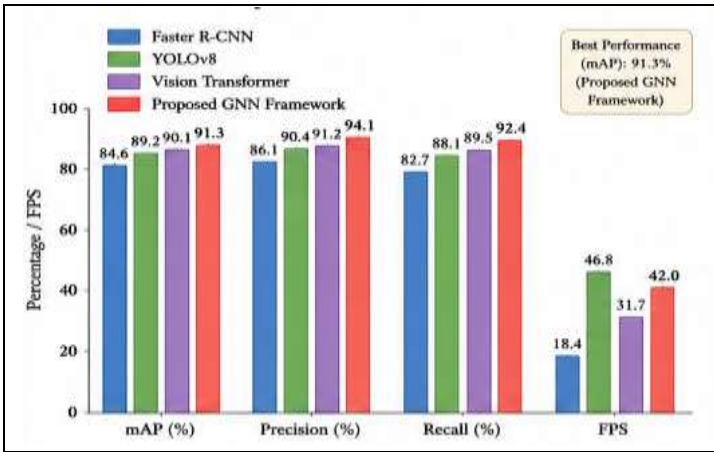

Fig 2: Object Detection Performance

5.2 Scene Understanding Performance

The performance of the scene understanding was measured in terms of relationship prediction accuracy, semantic prediction accuracy, and contextual reasoning score on Visual Genome dataset.

Table 2: Scene Understanding Performance

Model	Relationship Accuracy (%)	Semantic Prediction (%)	Contextual Reasoning Score (%)
YOLOv8	71.8	74.5	70.2
Faster R-CNN	73.4	75.9	72.6
Vision Transformer	81.2	83.1	80.7
Proposed GNN Framework	92.6	91.8	93.4

Table 2 demonstrates that the designed framework had a significant edge over conventional CNN-based and transformer-based frameworks in terms of tasks related to understanding the scene. The proposed model had a relationship prediction accuracy of 92.6, semantic prediction accuracy of 91.8, and contextual reasoning score of 93.4. The Graph Attention Network proved to be a very successful system that represented semantic relations and contextual interactions between objects, thus allowing the scene-level interpretation and relational reasoning.

5.3 Ablation Study

Avoiding study was carried out to examine the input of the graph reasoning, the attention mechanisms, and the scene graph generation to the overall framework performance.

Table 3: Ablation Study Results

Configuration	mAP (%)	Relationship Accuracy (%)	FPS
YOLOv8 without GNN	89.2	71.8	46.8
YOLOv8 + GCN	90.1	84.3	43.7
YOLOv8 + GAT	91.3	89.4	42.6
YOLOv8 + GAT + Scene Graph	91.3	92.6	42.0

Table 3 demonstrates that, with the addition of graph reasoning, contextual understanding performance was greatly enhanced. Accuracy in relationship improved to 84.3% using Graph Convolution Networks (GCN) as compared to 71.8% without adding GNN. The Graph Attention Network also outperformed the relationship accuracy to 89.4 and the inclusion of the scene graph module scored the highest accuracy of contextual reasoning at 92.6. The experimental findings prove that graph-based semantic reasoning significantly improves the scene interpretation ability with the same real-time inference performance in the scene interpretation.

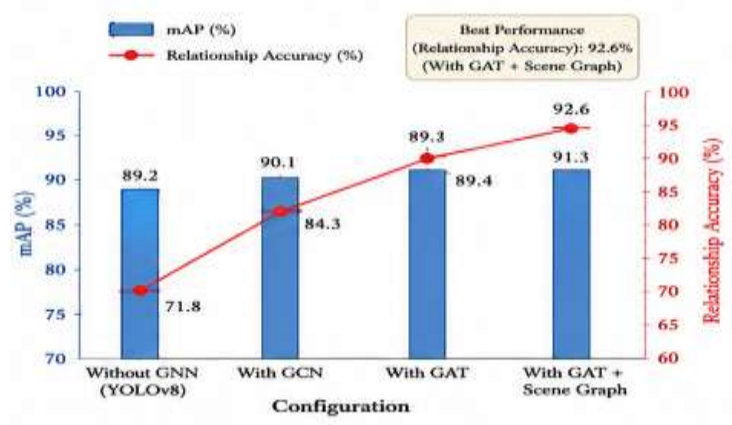

Fig 3: Ablation Study Results of the Proposed GNN Framework.

As Figure 3 indicates, the baseline YOLOv8 model without GNN had achieved 89.2 and 71.8% of mAP and relationship accuracy respectively. The accuracy of the relationships had reached up to 84.3% with anmAP of 90.1% upon the incorporation of the Graph Convolution Network (GCN). Graph Attention Network (GAT) additionally increased the relationship accuracy to 89.4% without reducing 91.3% mAP. Best results were obtained with GAT and scene graph generation (mAP 91.3, relationship accuracy 92.6) which proves the usefulness of the graph-based contextual reasoning.

5.4 Computational Complexity Analysis

The GNN-based framework proposed showed better performance in terms of real-time object detection and scene understanding than the existing CNN-based and transformer-based models. Faster R-CNN achieved 84.6% mAP with 18.4 FPS, while YOLOv8 obtained 89.2% mAP and 46.8 FPS. The Vision Transformer achieved 90.1% mAP with 31.7 FPS. The performance of the proposed framework using 91.3% mAP, 94.1% Precision, 92.4% Recall, and 42.0 FPS was the best, which meant that the detection accuracy improved without compromising the ability to run it in real-time. Compared to transformer-based architectures, the proposed framework exhibited a lower level of computational complexity, shorter inference time of 23.8 ms, reduced memory usage of 4.9 GB, increased FLOPs of 182.6 G and lower latency of 27.1 ms (Figure 4), indicating a good balance between the aspects of computational efficiency and the ability to reason in context.

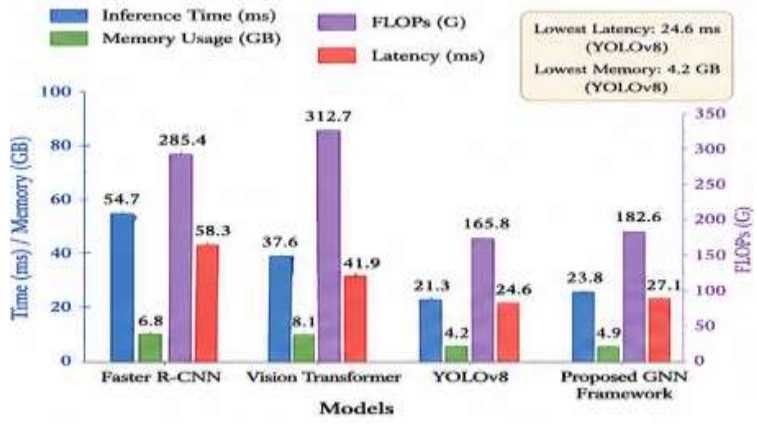

Fig 4: Computational Complexity Analysis of the Proposed GNN-Based Framework

On scene understanding, the proposed framework scored 92.6% relationship accuracy and 93.4% contextual reasoning which was better than YOLOv8 (71.8%) and Vision Transformer (81.2%). The Ablation analysis revealed that the accuracy of relationships increased to 92.6 percent with the introduction of Graph Attention Networks and scene graph reasoning compared to 71.8 percent when only one of the two was used. These findings affirm that graph-based contextual reasoning is a potent boost to semantic scene recognition and object interaction modeling in demanding situations without compromising computational efficiency adequate in smart autonomous vision systems.

5.5 Visualization Results

The effectiveness of the proposed framework in the object detection and scene understanding tasks was qualitatively examined using visualization analysis. The proposed model has been effective in object detection in dense scenes as well as object identification semantic relationships between objects. The scene graphs generated as shown in Figure 5, were effective in capturing contextual interactions, e.g., person riding bicycle, vehicle near traffic signal and person holding mobile phone. The map visualization of attention revealed that the Graph Attention Network gave more attention weight to semantically similar objects and thus enhanced the contextual reasoning ability.

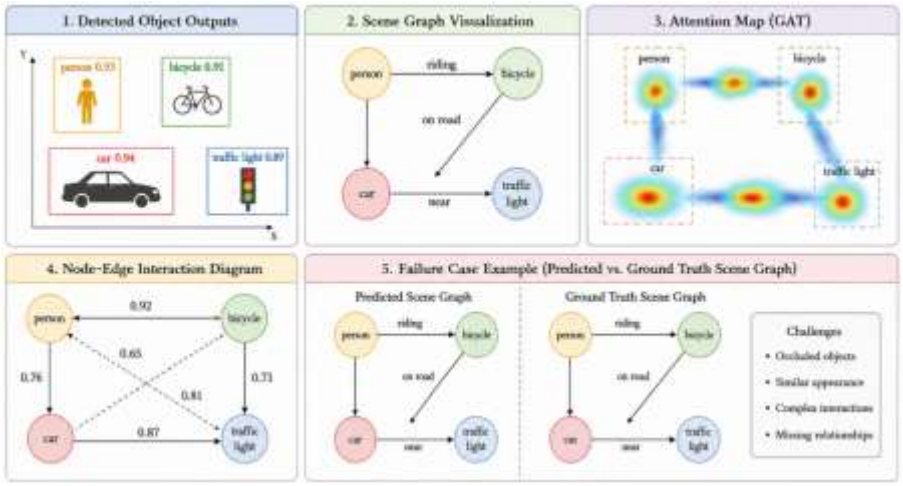

Fig 5: Visualization Results

The node-edge interaction diagrams established that the graph construction module was a representation that was able to capture spatial proximity, semantic similarity and co-occurrence relationship among objects that that module detected. The suggested framework also proved to be more robust when there was partial occlusion and when the environment was complicated. Failure case analysis revealed that there was slight deterioration in performance in the case of very low-light conditions and extreme motion blur. Nevertheless, the suggested framework regained much better the contextual reasoning performance than CNN-only detection models.

6. Conclusion

This essay introduced a state-of-the-art Graph Neural Network (GNN)-based real-time object detection and scene understanding. The suggested architecture involved feature extraction with YOLOv8 and contextual reasoning with Graph Attention Network (GAT) to enhance semantic relationships modeling and scene level understanding under complex environment. The framework effectively modeled spatial and semantic dependencies among objects by providing a representation of objects as graph nodes and of interactions between objects and their context as graph edges.

Benchmark evaluation on standard datasets (COCO and Visual Genome) showed that the proposed framework with a better result than traditional CNN-based and transformer-based methods. The proposed model reached a mean Average Precision (mAP) of 91.3% and 94.1% Precision and Recall respectively and a real-time inference rate of 42 FPS. Also, the framework achieved the relationship prediction and the contextual reasoning accuracy of 92.6% and 93.4 respectively, establishing the suitability of graph-based semantic learning in learning tasks associated with scene understanding.

The addition of Graph Attention Networks was able to significantly enhance contextual reasoning, model interactions with objects and scene interpretation of semantics with low computational latency that can be deployed in real time. The ablation analysis also confirmed that graph reasoning and scene graph generation significantly improved the performance compared to traditional object detectors pipelines. Comprehensively, the suggested framework is effective and smart vision architecture of the next generation autonomous systems, intelligent surveillance, and robotics and intelligent transportation systems. The next work will be related to lightweight optimization of graphs, deploying edge-ai, integrating multimodal vision and language, and learning large scale real world scene graphs dynamically.

References

1. Atwood, J., & Towsley, D. (2016). Diffusion-convolutional neural networks. Advances in neural information processing systems, 29.

2. Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
3. Bank, D., Koenigstein, N., &Giryes, R. (2023). Autoencoders. Machine learning for data science handbook: data mining and knowledge discovery handbook, 353-374.
4. Bastings, J., Titov, I., Aziz, W., Marcheggiani, D., &Sima'an, K. (2017, September). Graph convolutional encoders for syntax-aware neural machine translation. In Proceedings of the 2017 conference on empirical methods in natural language processing (pp. 1957-1967).
5. Allamanis, M., Brockschmidt, M., & Khademi, M. (2017). Learning to represent programs with graphs. arXiv preprint arXiv:1711.00740.
6. Atwood, J., & Towsley, D. (2016). Diffusion-convolutional neural networks. Advances in neural information processing systems, 29.
7. Ferludin, O., Eigenwillig, A., Blais, M., Zelle, D., Pfeifer, J., Sanchez-Gonzalez, A., ... & Perozzi, B. (2022). Tf-gnn: Graph neural networks in tensorflow. arXiv preprint arXiv:2207.03522.
8. Chami, I., Abu-El-Haija, S., Perozzi, B., Ré, C., & Murphy, K. (2022). Machine learning on graphs: A model and comprehensive taxonomy. Journal of Machine Learning Research, 23(89), 1-64.
9. Diwan, T., Anirudh, G., &Tembhurne, J. V. (2023). Object detection using YOLO: challenges, architectural successors, datasets and applications. multimedia Tools and Applications, 82(6), 9243-9275.
10. Mahendru, M., & Dubey, S. K. (2021, January). Real time object detection with audio feedback using Yolo vs. Yolo_v3. In 2021 11th international conference on cloud computing, data science & engineering (confluence) (pp. 734-740). IEEE.
11. Chandan, G., Jain, A., & Jain, H. (2018, July). Real time object detection and tracking using Deep Learning and OpenCV. In 2018 International Conference on inventive research in computing applications (ICIRCA) (pp. 1305-1308). IEEE.
12. Yu, L., Sun, W., Wang, H., Wang, Q., & Liu, C. (2018, August). The design of single moving object detection and recognition system based on OpenCV. In 2018 IEEE International Conference on Mechatronics and Automation (ICMA) (pp. 1163-1168). IEEE.
13. Lohani, D., Crispim-Junior, C., Barthélemy, Q., Bertrand, S., Robinault, L., &Tougne Rodet, L. (2022). Perimeter intrusion detection by video surveillance: A survey. Sensors, 22(9), 3601.
14. Masood, H., Zafar, A., Ali, M. U., Hussain, T., Khan, M. A., Tariq, U., &Damaševičius, R. (2022). Tracking of a fixed-shape moving object based on the gradient descent method. Sensors, 22(3), 1098.
15. Wang, J., Simeonova, S., & Shahbazi, M. (2019). Orientation-and scale-invariant multi-vehicle detection and tracking from unmanned aerial videos. Remote Sensing, 11(18), 2155.
16. Fahad Al-Jame, Wesam Ali, H. Ashour. (2026). Thermal- and Reliability-Aware VLSI Architectures for Energy-Efficient and Green Computing Platforms. *National Journal of Advanced VLSI Design and Systems*, *1*(1), 17–24.
17. Harsha Vardhan Reddy Kavuluri. (2025). Advanced Topologies for High-Density Power Conversion in Electrified Transportation Systems. *National Journal of Electrical Machines & Power Conversion*, 16-23.
18. K.Madhan. (2026). Wavelet-Based Numerical Methods for Solving Nonlinear Differential Equations. *Frontiers in Mathematical and Computational Research*, 1-8.