Neutrosophic Based Traffic Management System

Tushar Gupta [12312021], Nitish Kumar Choubey [12312031], Harshit Singhal [12312016], Nikhil Nagar [12312007], Devansh Bansal [12312006], Aryan Panwar [12312027]

Under Supervision of

Dr. Jitendra Kumar Samriya

Indian Institute of Information Technology Sonepat,

Haryana, India

Abstract: - The Neutrosophic Based Traffic Management System offers a new approach to managing traffic signals intelligently by combining advanced object detection with a distinctive decision-making technique known as neutrosophic logic. We used top-notch deep learning tools like YOLOv8, SSD, YOLOv5, and RetinaNet to spot vehicles such as cars, buses, motorcycles, and trucks, along with obstacles like people or stray animals (like dogs, cows, etc.) on the road, all from images captured in real time. The system figures out how crowded the road is and how many obstacles are around, showing the results through easy-to-understand heatmaps and boxes around detected objects. At its core, the system relies on neutrosophic logic, which examines traffic using three factors-truth for traffic density, indeterminacy for obstacles, and falsity for uncertainty-helping it make well-rounded decisions in challenging situations. Using these insights, it suggests whether the traffic light should be red, yellow, or green, keeping both smooth traffic flow and safety in mind. We built this system with Streamlit, giving it a simple interface where anyone can upload a road image, pick a model, and see a breakdown of vehicles types, traffic density, and logic results. Aimed at solving city traffic jams and safety issues, this system has the potential to fit into smart city projects, with smarter learning to handle ever-changing traffic conditions.

Keywords: - Neutrosophic fuzzy Logic, Objects and Obstacles Detection, Density Heat Map, Vehicle Detection

1. Objective of the Project

The purpose of making this Neutrosophic Based Traffic Management System is to build a intelligent AI based traffic signal setup that makes traffic smoother by using heat map to detect objects and obstacles density and a special decision-making tool called neutrosophic fuzzy logic. We designed the system which takes images to spot vehicles and obstacles on the road in real time with the help of powerful deep learning tools like YOLOv8, YOLOv5, SSD, and RetinaNet. It checks how busy the road is and looks for things like people or animals that might get in the way, that suggests the best traffic signal-whether red, yellow, or green-using neutrosophic logic to make thoughtful decisions. We have also added User interface so that signals are clearly visible and makes project more adaptable and reusable.it can also make easier to implement on large scale.

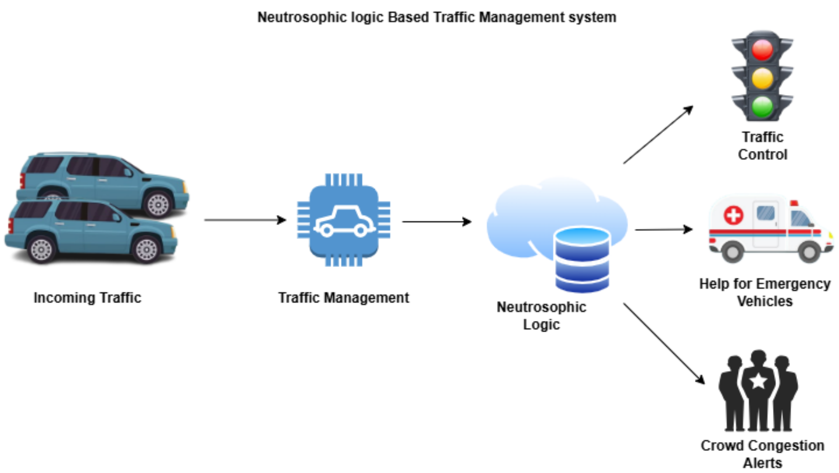


Fig. 1

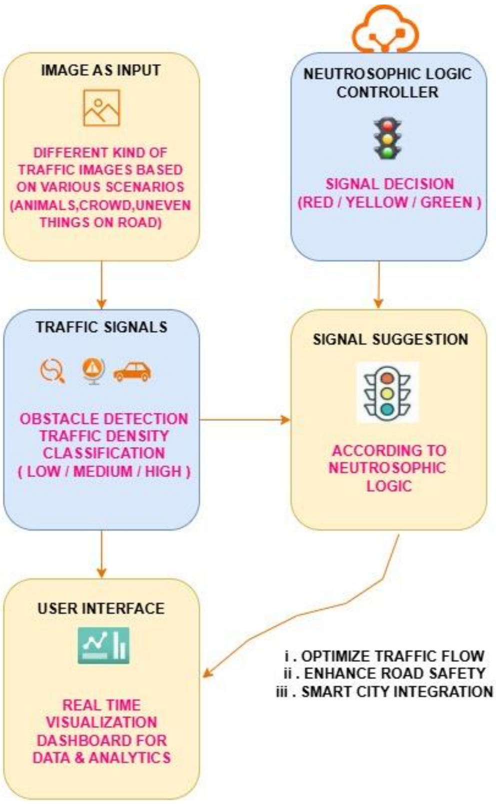


Fig. 2

2. History of the Project (2018-25)

From 2018 to 2025, computer vision has made big changes in smart traffic systems, thanks to advancements in object detection algorithms and neural architectures. Four models—**YOLOv5, YOLOv8, RetinaNet, and SSD**—have shaped this evolution and formed the backbone of real-time vehicle detection in neutrosophic traffic signal controllers.

2018–2019: Early Adoption and SSD Maturity

Mr. Sathishkumar Veerappampalayam Easwar Moorthy talks about the Single Shot MultiBox Detector (SSD). It had already become a popular object detection model due to its speed and relatively high accuracy. SSD provided a strong alternative to older R-CNN approaches, being capable of detecting vehicles in real-time.

2020: RetinaNet's Dominance and SSD Optimization

Mr. Muhammad Ovais Yusuf and Ahmad Jalal successfully talks about the RetinaNet. RetinaNet is good for its robustness in detecting small and overlapping objects—an essential feature for traffic junctions and intersections.

2021–2022: YOLOv5 Revolution

Mr. Dontabhaktuni Jayakumar and Samineni Peddakrishna talks about the evolution of YOLOv5.It marked a major shift in the object detection landscape. It combined PyTorch implementation, modular design, and extensive pre-trained weights. Its ability to detect vehicles of varying sizes and in different lighting/weather conditions made it ideal for traffic analysis.

During this time, traffic researchers adopted YOLOv5 for:

- Real-time vehicle detection.
- Density-based signal control.
- Traffic rule violation detection.

2023–2024: Rise of YOLOv8 and Lightweight Optimization

Mr. Mupparaju Sohan and Venkata Rami Reddy talks about the YOLOv8 model. It brought significant performance gains in both speed and accuracy.
Key benefits that made YOLOv8 preferred in intelligent traffic systems:

- Lighter model size suitable for edge deployment.
- Faster inference, critical for high-speed vehicle detection.
- Better precision-recall trade-off for dynamic traffic environments.

2025: Multi-Model Frameworks and Unified Deployments

Mr. Hesham A. Sakr and Magda I. EI-Afifi give the full overview of Intelligent Traffic Management Systems. With the maturity of edge AI platforms and high-speed 5G/6G connectivity in 2025, traffic management systems began supporting hybrid deployment of multiple models:

- YOLOv8 for primary real-time vehicle detection.
- YOLOv5 for scenarios requiring extensive training data adaptation.
- RetinaNet in high-accuracy surveillance intersections.
- SSD-MobileNet in budget-constrained setups or battery-powered intersections

Table no. 1: - Comparison Table: Evolution from 2018 to 2025

Year(s)	Model(s)	Key Developments	Application in Traffic Systems
2018–2019	SSD, Retina Net	- SSD matured with fast inference but poor small object detection- Retina Net introduced Focal Loss	- Basic real-time vehicle detection- Retina Net used for high-precision, crowded scenarios
2020	Retina Net, SSD	- Retina Net became preferred for robustness- SSD optimized as Mobile Net-SSD and Lite SSD variants	- Deployed on GPU/TPU edge devices- SSD used in low-power devices like Jetson Nano
2021–2022	YOLOv5	- YOLOv5 gained popularity (Ultralytics)- Lightweight versions (“n”, “s”) and heavy versions (“x”)	- Real-time detection, signal control, rule violation monitoring- Deployed on edge devices
2023–2024	YOLOv8, YOLOv5	- YOLOv8 introduced anchor-free design- AutoShape, faster and lighter than YOLOv5	- Rapid vehicle detection at signals- Integrated into neutrosophic logic signal systems
2025	All 4	- Hybrid deployments: YOLOv8 (main), YOLOv5 (custom), Retina Net (surveillance), SSD-Mobile Net (low-cost)	- Ensemble models switch based on weather, traffic, device- Tied into 6G edge AI systems

3. Our Contribution

In this project, we propose an intelligent Neutrosophic-Based Traffic Management System that integrates advanced object detection models - YOLOv8, YOLOv5, RetinaNet, and SSD—for real-time vehicle detection and classification. The system uses neutrosophic logic to handle uncertainties in traffic conditions and dynamically adjust signal timings based on traffic density. We benchmarked and optimized the models for performance on edge devices, ensuring low latency and energy efficiency. A secure, scalable architecture and an interactive Streamlit dashboard enable real-time monitoring and control. Our solution addresses both computational efficiency and real-world adaptability, offering a robust framework for future smart traffic systems.

Approach/Technique: - Our approach for the Neutrosophic Based Traffic Management System involved integrating four advanced object detection models—YOLOv8, YOLOv5, SSD, and RetinaNet—using PyTorch to accurately detect vehicles and obstacles in road images. We developed custom density analysis functions to compute coverage and visualize it through heatmaps using OpenCV. To make the traffic system smarter, we applied Neutrosophic logic across all four models, a decision-making method that uses truth (traffic density), indeterminacy (obstacle presence), and falsity (combined uncertainty) values to suggest optimal signals (Red, Yellow, Green). Using Streamlit, we created an interactive UI for image uploads, model selection, and result visualization, ensuring a modular design for scalability.

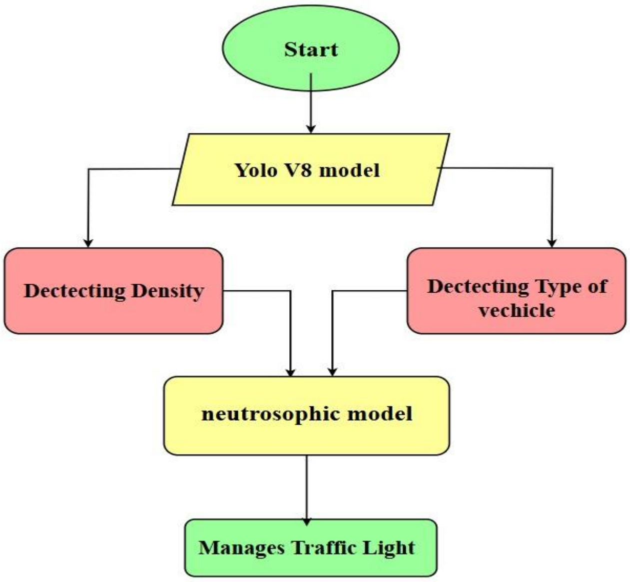


Fig. 3

Implementation: - In building the Neutrosophic Based Traffic Management System, our primary focus was to create an intelligent traffic signal control mechanism by combining advanced object detection with a decision-making framework. We started by integrating four object detection models YOLOv8, SSD, YOLOv5, and RetinaNet using PyTorch and the Ultralytics library to ensure accurate detection of vehicles (such as cars, motorcycles, buses, trucks) and obstacles (such as pedestrians, animals) in road images. Each model was carefully initialized with error handling to detect loading issues, and we defined specific vehicle and obstacle classes for each to generalize detection across models. For each model, we developed custom density analysis functions (e.g., yolov8_density, ssd, density) that calculate the coverage percentage of vehicles and obstacles on the road, generating heatmaps and bounding boxes using OpenCV for visual interpretation. The core of our system lies in the application of Neutrosophic logic, a decision making approach that evaluates traffic conditions using three key values: Truth (T), which reflects vehicle density which scaling coverage to a 0-1 range; Indeterminacy (I), which detects for obstacle presence by normalizing obstacle coverage; and Falsity (F), which captures combined uncertainty as the complement of T and I. We applied this logic uniformly across all four models, enabling the system to analyse traffic scenarios. For instance, if vehicle coverage exceeds 50% ($T > 0.5$), the system suggests a "Green" signal to prioritize traffic flow, but if obstacle coverage is high ($I > 0.6$), it switches to "Red" for safety. These signal suggestions (Red, Yellow, Green) effectively control traffic lights, optimizing both flow and safety. To make the system user-friendly, we used Streamlit to build an interactive interface where users can upload images, select a model, and view detailed results, including vehicle counts, density metrics, heatmaps, and signal recommendations, ensuring a seamless experience for traffic management.

Pseudocode

Algorithm 1 Object Detection Neutrosophic Traffic Decision Algorithm

Initialize model type $M \in$ YOLOv5, YOLOv8, RetinaNet, SSD
Load pretrained weights and configuration for model M
Load traffic video/image dataset D
Set Neutrosophic thresholds: Truth T, Indeterminacy I, Falsity F
Set decision thresholds: $\tau_{\text{density}}, \tau_{\text{priority}}$
for each frame f in D **do**
    Detect vehicles $V_f \leftarrow \text{ModelDetect}(M, f)$
    Classify objects by type: car, bike, bus, truck
    Count vehicles C_f and determine lane-based density
    Calculate Neutrosophic values T_f, I_f, F_f based on density, time, and motion
    Evaluate congestion score $S_f = \alpha T_f + \beta I_f + \gamma F_f$
    if $S_f \geq \tau_{\text{density}}$ **then**
        Update signal timing for frame f (e.g., extend green)
        Trigger alert or rerouting logic if accident or crowd detected
    else
        Maintain default traffic cycle
    end if
    Log results: vehicle count, decision, frame index
end for
Generate report: average congestion, response times, model accuracy

Tables of the Graphs

YOLOv5

Image	Total vehicles	Vehicle Types	Coverage	Obstacle coverage	Density	T	I	F
Image1	4	Cars:3, Trucks:1	27.0%	15.9%	HIGH	0.27	0.32	0.71
Image2	22	Cars: 20, Trucks: 2	44.1%	0.00%	HIGH	0.44	0.00	0.78
Image3	3	Cars: 2, Trucks: 1	11.0%	13.5%	MEDIUM	0.11	0.27	0.81
Image4	18	Cars: 16, Trucks: 2	5.9%	1.0%	LOW	0.06	0.02	0.96
Image5	36	Cars: 31, Trucks: 3, Buses: 2	39.0%	0.0%	HIGH	0.39	0.00	0.81
Image6	0	None	0.0%	0.2%	LOW	0.00	0.00	1.00
Image7	28	Cars: 16, Buses: 2, Motorcycles: 7, Trucks: 3	24.5%	1.4%	MEDIUM	0.24	0.03	0.86
Image8	34	Buses: 4, Cars: 26, Trucks: 4	52.5%	0.5%	HIGH	0.52	0.01	0.73
Image9	0	None	0.00%	55.4%	LOW	0.00	1.00	0.50
Image10	20	Trucks: 8, Cars: 11, Buses: 1	17.9%	0.0%	MEDIUM	0.18	0.00	0.91
Image11	23	Cars: 20, Trucks: 3	19.6%	0.0%	MEDIUM	0.20	0.00	0.90
Image12	0	None	0.00%	36.8%	LOW	0.00	0.74	0.63
Image13	0	None	0,00%	84.7%	LOW	0.00	1.00	0.50
Image14	1	Trucks: 1	8.0%	16.8%	LOW	0.08	0.34	0.79
Image15	4	Cars: 2, Trucks: 2	13.5%	22.8%	MEDIUM	0.13	0.46	0.70

Table 1

RetinaNet

Image	Total Vehicles	Vehicle Types	Coverage	Obstacle Coverage	Density	T	I	F
Image1	5	Cars:3, Trucks:2	34.9%	0.9%	HIGH	0.35	0.02	0.82
Image2	27	Cars:24, Trucks: 3	42.2%	0.0%	HIGH	0.42	0.00	0.79
Image3	3	Cars:2, Trucks:1	11.4%	0.0	MEDIUM	0.11	0.00	0.94
Image4	7	Cars:6, Trucks:1	4.5%	0.8%	LOW	0.04	0.02	0.97
Image5	28	Cars:26, Buses:2	31.0%	0.0%	HIGH	0.31	0.00	0.84
Image6	0	None	0.0%	0.2%	LOW	0.00	0.00	1.00
Image7	36	Cars:26, Buses:1, Motorcycles:9	27.1%	2.5%	HIGH	0.27	0.05	0.84
Image8	47	Cars:40, Buses:2, Motorcycles:1, Trucks:4	55.1%	0.3%	HIGH	0.55	0.01	0.72
Image9	0	None	0.0%	53.4%	LOW	0.00	1.00	0.50
Image10	18	Cars:15, Trucks:3	11.9%	0.0%	MEDIUM	0.12	0.00	0.94
Image11	24	Cars:24	17.1%	0.0%	MEDIUM	0.17	0.00	0.91
Image12	0	None	0.0%	36.3%	LOW	0.00	0.73	0.64
Image13	0	None	0.0%	92.5%	LOW	0.00	1.00	0.50
Image14	1	Trucks:1	8.3%	16.3%	LOW	0.08	0.33	0.80
Image15	3	Cars:2, Trucks:1	12.3%	0.0%	MEDIUM	0.12	0.00	0.94

Table 2

SSD (Single Shot MultiBox Detector)

Image	Total Vehicles	Vehicle Types	Coverage	Obstacle Coverage	Density	T	I	F
Image1	4	Cars:3, Trucks:1	27%	15.9%	HIGH	0.27	0.32	0.71
Image2	27	Cars:24, Trucks: 3	44.1%	0.0%	HIGH	0.44	0.32	0.78
Image3	3	Cars:2, Trucks:1	11.4%	13.5%	MEDIUM	0.11	0.27	0.81
Image4	18	Cars:16, Trucks:2	5.9%	1.0%	LOW	0.06	0.02	0.96
Image5	36	Cars: 31, Trucks: 3, Buses: 2	24.5%	0.0%	HIGH	0.39	0.00	0.81
Image6	0	None	0.0%	0.2%	LOW	0.00	0.00	1.00
Image7	28	Cars:26, Buses:1, Motorcycles:9	27.1%	2.5%	HIGH	0.24	0.05	0.86
Image8	47	Cars:40, Buses:2, Motorcycles:1, Trucks:4	24.1%	0.5%	HIGH	0.52	0.01	0.73
Image9	0	None	0.0%	55.4%	LOW	0.00	1.00	0.50
Image10	20	Cars:15, Trucks:3	11.9%	0.0%	MEDIUM	0.18	0.00	0.91
Image11	23	Cars: 20, Trucks: 8	17.1%	0.0%	MEDIUM	0.07	0.00	0.9
Image12	0	None	0.0%	36.3%	LOW	0.00	0.74	0.69
Image13	0	None	0.0%	84.9%	LOW	0.00	1.00	0.50
Image14	1	Trucks:1	8.0%	16.8%	LOW	0.08	0.34	0.79
Image15	3	Cars: 2, Trucks: 2	13.5%	22.8%	MEDIUM	0.13	0.46	0.70

Table 3

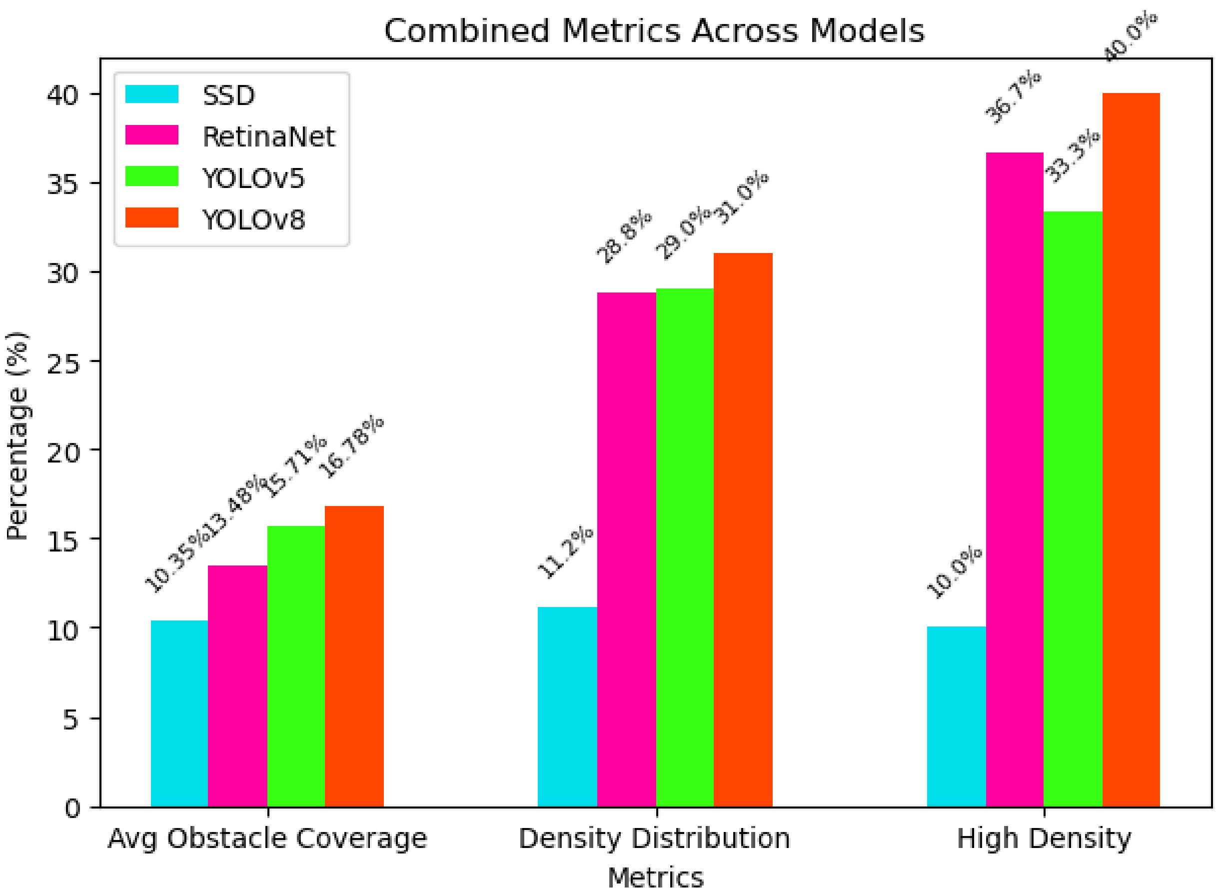
YOLOv8

Image	Total Vehicles	Vehicle Types	Coverage	Obstacle Coverage	Density	T	I	F
Image1	4	Cars:3, Trucks:1	27.3%	16.0%	HIGH	0.27	0.32	0.70
Image2	34	Cars:33, Trucks:1	48.4%	0.0%	HIGH	0.48	0.00	0.76
Image3	3	Cars:2, Trucks:1	11.4%	13.2%	MEDIUM	0.11	0.26	0.81
Image4	12	Cars:12	6.4%	1.1%	LOW	0.06	0.02	0.96
Image5	34	Cars:30, Buses:2, Truncks:2	37.8%	0.0%	HIGH	0.38	0.00	0.81
Image6	0	None	0.0%	0.2%	LOW	0.00	0.00	1.00
Image7	33	Cars:22, Motorcycles:9, Buses:1, Trucks:4	27.0%	0.7%	HIGH	0.27	0.01	0.86
Image8	53	Cars:46, Buses:4, Trucks:3	54.4%	0.0%	HIGH	0.54	0.00	0.73
Image9	1	Cars:1	0.2%	53.7%	LOW	0.00	1.00	0.50
Image10	29	Cars:28, Buses:1	15.6%	0.0%	MEDIUM	0.16	0.00	0.92
Image11	24	Cars:24	17.7%	0.0%	MEDIUM	0.18	0.00	0.91
Image12	0	None	0.0%	35.6%	LOW	0.00	0.71	0.64
Image13	0	None	0.0%	89.3%	LOW	0.00	1.00	0.50
Image14	1	Trucks:1	7.9%	17.7%	LOW	0.08	0.35	0.78
Image15	4	Cars:2, Trucks:2	14.5%	21.5%	MEDIUM	0.14	0.43	0.71

Table 4

Graphs

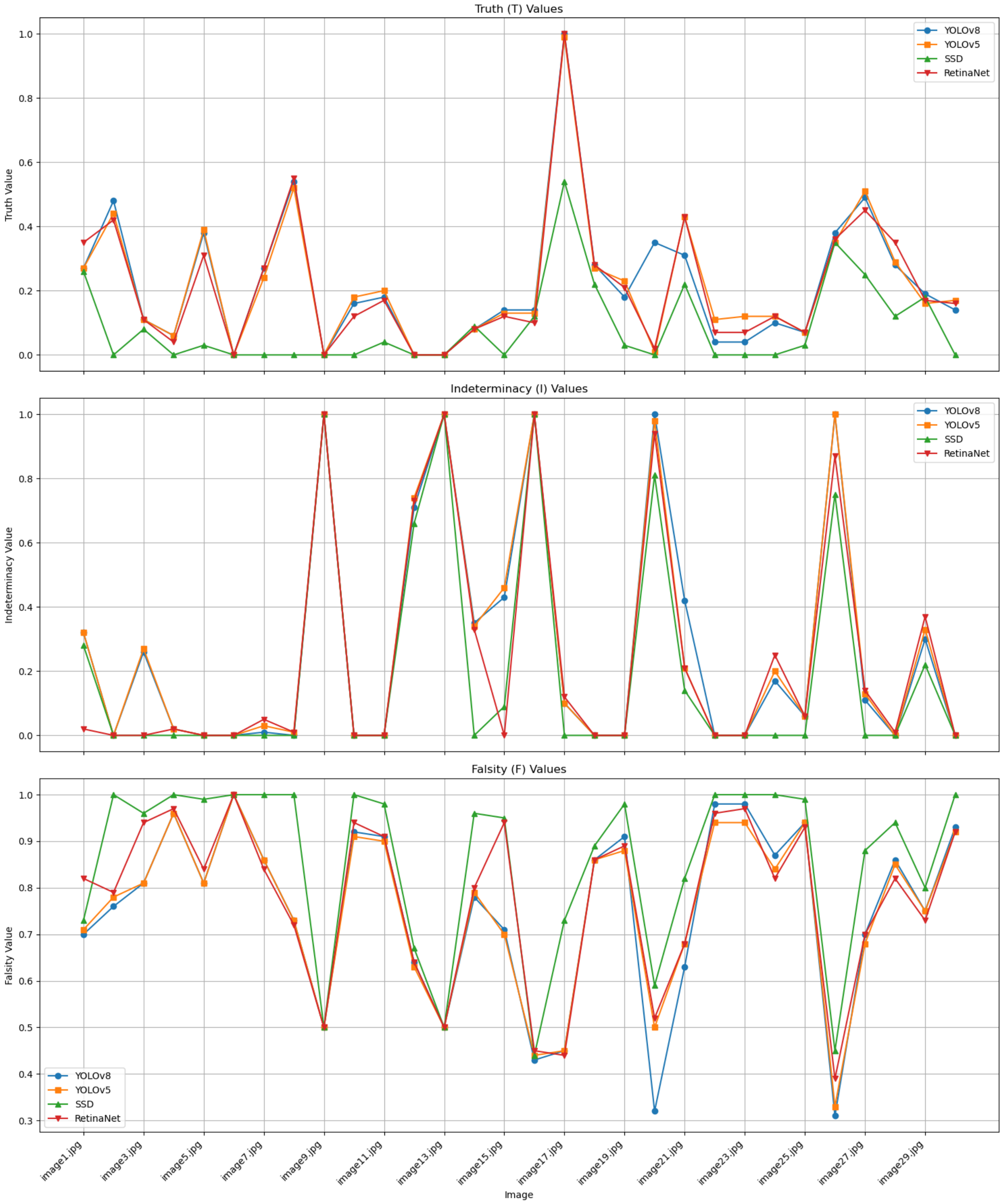
This graph gives us the information about Combines Metrics Across all four models.



Graph 1

Reference of the Graph: - Table 1 , Table2 , Table 3 , Table 4

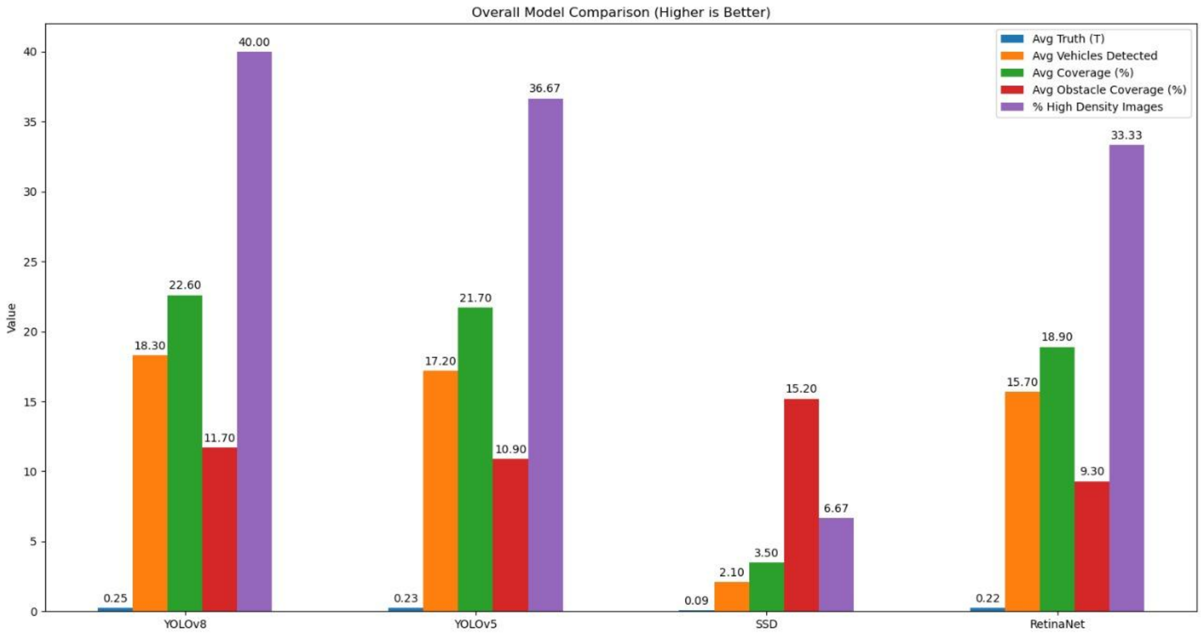
This graph gives us the information about Truth, Indeterminacy and Falsity values of all four models



Graph 2

Reference of the Graph: - Table 1 , Table2 , Table 3 , Table 4

This graph gives us information about Overall Model Comparison for all four models



Graph 3

Reference of the Graph: - Table 1 , Table2 , Table 3 , Table 4

So, from above graphs we can conclude that YOLOv8 model is more efficient.

Limitations

The traffic management system which is based on Neutrosophic Logic, it contains certain limitations. This system is based on static images which analysis the image input and there are some limits apply in dynamic traffic scenarios. The above four models may some time lead to inaccuracy in results with various situations such as poor lighting, heavy traffic. According to some research papers from which we have taken the reference, the Neutrosophic logic uses threshold values for decision-making which led to reduce flexibility. The another limitation is that we use only images or videos in some advanced projects. There are different density functions for each model which apply loose coupling for scalability for large and complex projects. The absence of real time feedback requires manual analysis for each model. These limitations enhance the practicality for real-world traffic management

4. Future Work

The traffic management system which is based on Neutroscopic Logic. It is implemented and has strong foundation for intelligent traffic control, as we know that any project is not fully completed.so there can be several enhancement and improvements by adding some new features which increases its efficiency in results among different models in future. There can be possible features such as real-time video processing and live camera feeds which enable continuous monitoring which further making the system more suitable for dynamic traffic scenarios. the object detection models (YOLOv8, YOLOv5, SSD, Retina Net) uses different datasets that could improve accuracy under limitations which we discuss above paragraph. We can also implement IOT devices or hardware which enable actual traffic control signals. We can optimize the system for cloud deployments available on internet.by using some techniques we can also make robust and versatile for real life.

5. References

[1] “Traffic Management System Using YOLO Algorithm”.

https://doi.org/10.3390/engproc2023059210

[2] "Performance Evaluation of YOLOv5-based Custom Object Detection Model for CampusSpecific Scenario".
https://doi.org/10.52756/ijerr.2024.v38.005

[3] "Smart Traffic Management System"
DOI:10.5120/13123-0473

[4] "Intelligent Traffic Management Systems: A review"

DOI:10.21608/njccs.2023.321169

6. Individual Contribution

Group Members & their contributions: -

- **Tushar Gupta: -** Dataset Collections, Generating the data for the table & graphs, Graph Generation
- **Nitish Kumar Choubey: -** Code & Working on the models, Documentation Work, Formatting
- **Harshit Singhal: -** Tables, Contents for the Documentation, Draw.io Flow Chart Diagrams
- **Nikhil Nagar: -** Tables, Exploring the research papers for reference
- **Devansh Bansal: -** Content for the various topics of the Documentation, Research Work
- **Aryan Panwar: -** Insights & Research Work