SCIENTIFIC CULTURE
www.sci-cult.com

SCIENTIFIC CULTURE, Vol. 12, No. 4, (2026), pp. 9578-9591
Open Access. Online & Print

DOI: 10.5281/zenodo.12426938

RDFUSION-NET: COMBINING DEEP LEARNING AND TEXTURE FEATURES FOR ACCURATE ROAD DENSITY MONITORING

Thodeti Spandana[1*], Sudha Pelluri[2]

[1] **Research Scholar, Department of Computer Science and Engineering, University College of Engineering, Osmania University, Hyderabad, Telangana, India. Email ID: spandana.academics@gmail.com*
[2] *Professor, Department of Computer Science and Engineering, University College of Engineering, Osmania University, Hyderabad, Telangana, India. Email ID: sudha.p@uceou.edu*

Received: 20/11/2025
Accepted: 11/03/2026

***Corresponding Author*: Thodeti Spandana (spandana.academics@gmail.com)**

Abstract:
This study introduces a novel approach for road density detection utilizing the RDFusion-Net architecture, which combines EfficientNet, DenseNet, and a Squeeze-and-Excitation (SE) block to improve feature extraction and classification. The system analyzes video input in AVI format, comparing individual frames with a reference image to detect traffic congestion. By concentrating on the green channel of the video frames, the system effectively identifies alterations in the road surface that signify vehicle presence. The disparities between the reference and input frames are computed and transformed into binary pictures utilizing the IsoData approach, which delineates regions of possible congestion. Texture-based characteristics are derived from binary photographs utilizing the Gray Level Co-occurrence Matrix (GLCM), offering insights into road scene texture, including energy and entropy. The RDFusion-Net design enhances feature extraction by utilizing EfficientNet's scalable capabilities and DenseNet's dense layer connections, hence providing strong feature learning. The SE block enhances the extracted features by adjusting the significance of each feature map, emphasizing those most pertinent to road density identification. The approach integrates conventional GLCM-based texture data with deep learning features to create a comprehensive feature vector for final classification via a softmax classifier, categorizing road density into low, medium, and high levels. The suggested RDFusion-Net demonstrates enhanced performance relative to conventional approaches, achieving an accuracy of 99.21%, precision of 98.89%, recall of 99.11%, and an F1 score of 99.37%. These measures underscore the model's efficacy in precisely detecting road congestion, rendering it appropriate for practical applications in traffic management and monitoring.

Keywords: *Road Density, accuracy, Precision, Traffic, Congestion, deep learning, recall, GLCM*

1. Introduction

Road traffic congestion has become a major issue in cities around the world, causing delays, increasing fuel use, and contributing to air pollution [1]. According to an INRIX analysis from 2023, traffic congestion costs the global economy more than $300 billion per year in lost productivity [2]. As cities grow, the burden on transportation infrastructure increases, resulting in longer travel times and greater irritation for commuters. Monitoring road density, or determining the number of vehicles on the road in a given region, is critical for understanding traffic trends and enabling real-time traffic management [3-5]. Road density detection enables city planners and traffic control systems to make more informed judgments to reduce congestion and improve road efficiency [6-9].

Road congestion poses several issues, ranging from economic losses to major safety concerns [10]. High road density frequently results in higher accident rates, as more vehicles on the road increase the likelihood of crashes [11-15]. Furthermore, congestion contributes to increased air pollution, which has a negative impact on public health and the environment [16-18]. Traffic congestion often causes delays for emergency vehicles like ambulances and fire trucks, which can be life-threatening. Given these hazards, there is an urgent need to monitor and regulate road density in real time, allowing authorities to implement prompt interventions and mitigate the effects of congestion [19-20].

Recent improvements in computer vision and machine learning have made a substantial contribution to enhancing road density detection systems. CNN+LSTM, GRUs with wavelets, and YOLOv5 models have all demonstrated potential in identifying congestion with varied levels of accuracy, precision, and recall. CNN+LSTM obtained 95% accuracy, but U-Net models achieved up to 99.09% accuracy. These models leverage deep learning approaches for image processing and object detection, resulting in faster and more accurate congestion assessments. Despite these advances, there are still hurdles to enhancing detection accuracy in real-time applications, particularly in varying traffic settings, occlusions, and severe weather conditions.

In this research, we provide RDFusion-Net, a model that combines EfficientNet and DenseNet architectures with a Squeeze-and-Excitation (SE) block to improve feature extraction for detecting road density. Our model outperforms earlier methods by merging texture-based features taken from the Gray Level Co-occurrence Matrix (GLCM) with deep learning features from RDFusion-Net. This hybrid technique improves accuracy while also ensuring robust detection in difficult settings. RDFusion-Net beats existing models in road density detection tasks, with an F1 score of 99.37%, recall of 99.11%, precision of 98.89%, and accuracy of 99.21%.

This paper is structured as follows: Section 2 highlights relevant research in the areas of road density detection and machine learning applications for traffic management. Section 3 describes the suggested technique, which includes the video input processing and feature extraction methods employed in the RDFusion-Net model. Section 4 describes the experimental setting and gives the results, comparing RDFusion-Net to other models. Finally, Section 5 summarizes the important findings and proposes future study areas, such as combining vehicle counts and handling detection in different weather conditions and occlusion settings.

2. Related Works

Slimane et al. [22] presented a hybrid solution to road density detection in 2024, integrating Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. Their model has 95% accuracy, 93% precision, 94% recall, an F1 score of 93.5%, and a mean absolute error (MAE) of 5%. The model's use of CNN and LSTM enabled it to capture both spatial and temporal variables, making it extremely effective at spotting road density trends. However, the intricacy of this model restricts its real-time implementation in traffic systems since it requires significant computational resources, making it difficult to apply in low-power devices or situations that require rapid processing times.

Fouzi et al. [23], also in 2024, used Gated Recurrent Units (GRUs) in conjunction with wavelet transformation to reach an amazing 98.2% accuracy. Wavelet transformation enabled more effective feature extraction, especially for complex traffic patterns and non-stationary data. Although the approach performs remarkably well in controlled circumstances, its reliance on extensive preprocessing and sensitivity to noise can limit its practical application in quickly changing traffic conditions, reducing its robustness in real-time scenarios.

In 2024, Whardana et al. [24] used YOLOv5 for traffic object recognition, achieving an 87% accuracy. While YOLOv5 is noted for its quick and efficient object recognition, its poorer accuracy in contrast to other models suggests that it may struggle with more complicated scenarios or surroundings containing overlapping and smaller items. Its restriction stems from its poor performance in heavily packed settings, where quick and exact identification is critical for safety.

In the same year, Mohanapriya et al. [25] used the U-Net architecture to achieve 99.09% accuracy, 98.11% precision, 98.84% recall, and a 99.21% F1 score. U-Net, a popular picture segmentation technique, proved extremely effective at detecting key regions in traffic photographs. Despite its excellent performance, the model's reliance on big annotated datasets and susceptibility to overfitting are a drawback, particularly when only small datasets are available for training.

In 2024, Kirti et al. [26] used YOLOv8 for traffic management applications, attaining 91.6% precision, 93.8% recall, and 97% accuracy. YOLOv8, an upgrade over earlier incarnations, provided higher detection performance. However, its ability to retain this high performance in more cluttered and obstructed surroundings was tested.

The shortcoming here is its occasional inability to detect tiny items or distinguish between overlapping ones in busy settings.

In 2024, Sankaranarayanan et al. [27] suggested a system for predicting traffic flows that combines mathematical modeling and neural networks. While the technique provides a theoretical foundation for analyzing traffic patterns, it has substantial limits in terms of scalability and adaptability in rapidly changing traffic situations, limiting its usefulness in real-world circumstances that require faster reactions.

In 2023, Balaji et al. [28] proposed an Adaptive Traffic Management (ATM) system based on machine learning that demonstrated high adaptation to shifting traffic patterns. This system provides a data-driven solution for increasing traffic efficiency and reducing congestion. However, the model's reliance on large and reliable data inputs poses a problem, as discrepancies or missing data might result in suboptimal traffic control performance.

Finally, in 2022, Wan et al. [29] used YOLOv5, but with substantially lower performance measures, attaining 59.2% precision, 58.2% recall, and an F1 score of 58.7%. This model's shortcoming arises from its low effectiveness in recognizing and classifying objects, making it unsuitable for traffic management systems that require greater accuracy and precision.

A number of constraints emerge across the models. Slimane et al.'s CNN+LSTM technique has low computational efficiency, making it unsuitable for real-time applications. Although Fouzi et al.'s wavelet-based GRUs are highly accurate, they are sensitive to noise and require extensive preprocessing. The YOLOv5 and YOLOv8 models, as implemented by Whardana et al. and Kirti et al., have limitations in thick or cluttered situations, where smaller or overlapping objects are difficult to detect. U-Net's reliance on huge datasets renders it susceptible to overfitting, as demonstrated by Mohanapriya et al.'s study. Finally, mathematical framework-based models, such as those developed by Sankaranarayanan et al., are frequently less flexible to real-world circumstances characterized by rapid traffic changes.

Given these constraints, RDFusion-Net was developed to address critical concerns by merging efficient feature extraction backbones like EfficientNet and DenseNet with attention methods like Squeeze-and-Excitation blocks. This approach enhances feature prioritization and scalability while maintaining computing efficiency, making it a promising solution for real-time road density identification.

3. Proposed System

The research's method as shown in Figure 1 involves processing a 500-frame traffic test video with the RDFusion-Net model, which combines EfficientNet, DenseNet, and a Squeeze-and-Excitation (SE) block for feature extraction and recalibration. First, video frames are taken at regular intervals, and the green channel is chosen for analysis because of its sensitivity to changes in the road surface. The difference between each frame and a reference image is computed, and the IsoData method is used to create binary images for congestion detection. The Gray Level Co-occurrence Matrix (GLCM) extracts texture properties such as energy and entropy. These texture-based features are integrated with deep learning features from EfficientNet and DenseNet to improve detection. The SE block refines the feature maps by rebalancing their relevance. Finally, the aggregated features are fed into a softmax classifier, which divides road density into three categories: low, medium, and high. The system is trained on 80% of the video frames and tested on the remaining 20%, with the proposed model outperforming previous models in terms of accuracy and robustness when identifying road congestion.

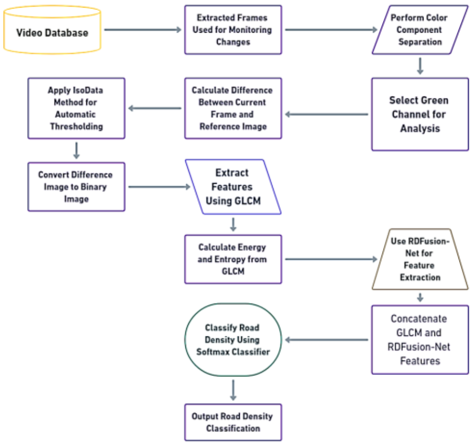


Fig. 1 Proposed detection and Classification System

The process begins with selecting an input video file in .avi format, which serves as the foundation for road density detection. Additionally, a blank reference road image $I_r(x,y)$ is selected, representing a non-congested road. This image is used for comparison with the frames extracted from the video, allowing for the identification of changes that indicate traffic congestion. Video frames $I_v(x,y,t)$ are extracted at regular intervals (e.g., every 10th frame) from the video stream. These frames are visualized to monitor the current status of the road scene and to track changes over time.

Once the frames are extracted, color component separation is performed. The input video frame $I_v(x,y,t)$ is divided into its Red, Green, and Blue channels, denoted as $I_R(x,y,t)$, $I_G(x,y,t)$ and $I_B(x,y,t)$ respectively. The green channel $I_G(x,y,t)$ is selected for further analysis, as it is often more sensitive to changes in road surfaces and vehicles. The difference between the green channel of the current frame and the reference image is calculated as:

$D(x,y,t) = |I_G(x,y,t) - I_r(x,y,t)|$ .... (01)

where $D(x,y,t)$ is the absolute difference image. This highlights changes in the road scene, which may be due to vehicles or other objects.

Next, the IsoData method is applied to determine an automatic threshold value T for converting the difference image into a binary image. The binary image $B(x,y,t)$ is given by

$B(x,y,t) = \begin{cases} 1 & if\ D(x,y,t) \geq T \\ 0 & if\ D(x,y,t) < T \end{cases}$ .... (02)

This binary image is used to isolate areas of potential congestion by identifying regions where the pixel values differ significantly from the reference image.

For feature extraction, the Gray Level Co-occurrence Matrix (GLCM) is used to capture texture-based information from the binary image $B(x,y,t)$. The GLCM is defined as $P(i,j,d,\theta)$ where i and j are the gray levels. d is the distance between pixel pairs and θ is the orientation. From the GLCM, features such as Energy and Entropy are calculated. Energy E is defined as:

$E = \sum_{i=1}^{N} \sum_{j=0}^{N} P(i,j,d,\theta)^2$ .... (03)

and Entropy H is given by

$H = -\sum_{i=1}^{N} \sum_{j=1}^{N} P(i,j,d,\theta) \log P(i,j,d,\theta)$ .... (04)

where N is the number of gray levels in the image. These texture features provide an initial estimate of road congestion, with higher entropy values indicating more randomness (potentially more vehicles or congestion) and lower energy values signifying a denser vehicle presence.

To enhance the feature extraction process, the RDFusion-Net architecture is introduced. RDFusion-Net integrates EfficientNet and DenseNet for feature extraction, along with a Squeeze-and-Excitation (SE) block to recalibrate feature maps. EfficientNet is utilized to scale the model's capacity by adjusting depth, width, and resolution, allowing it to capture both fine and coarse features. Mathematically, EfficientNet's compound scaling is controlled by:

$depth \sim \alpha^d$ .... (05)

$Width \sim \beta^w$ .... (06)

$resolution \sim \gamma^\tau$ .... (07)

where α, β and γ are scaling coefficients and d, w and τ are scaling exponents for depth, width, and resolution, respectively. This ensures balanced scaling of the network to optimize performance.

DenseNet enhances feature extraction by allowing dense connections between layers, ensuring that features from earlier layers are reused in later layers. The output of each layer in DenseNet is expressed as:

$x_l = H_l([x_0, x_1, \dots \dots \dots . x_{l-1}])$ .... (08)

where x_l is the output layer of l, H_l is a composite function (e.g., batch normalization, activation, and convolution), and $[x_0, x_1, \dots \dots \dots . x_{l-1}]$ represents the concatenation of feature maps from all previous layers. This promotes robust feature learning and prevents loss of important information.

The Squeeze-and-Excitation (SE) block refines the extracted features by recalibrating the importance of each feature map. The SE block introduces a recalibration weight s_c for each channel c of the feature map, calculated as:

$s_c = \sigma\big(W_2 \cdot ReLU(W_1 \cdot z)\big)$ .... (09)

where z is the global pooling of the feature map, W_1 and W_2 are learnable weights, and σ is the sigmoid activation function. This allows the model to prioritize more relevant features for the task of road density detection.

The combined features from both the GLCM and RDFusion-Net are concatenated to form a final feature vector. This vector includes traditional texture-based features like Energy and Entropy, as well as deep features extracted by RDFusion-Net. The combined feature vector is expressed as:

$Combined\ Features =$
$[E_1, E_2 \dots \dots . E_4, H_1, H_2 \dots \dots . H_4, f_{RDFusion}]$ .... (10)

where E_i and H_i represent energy and entropy values from different GLCM orientations, and $f_{RDFusion}$ represents the deep features extracted from RDFusion-Net.

The final classification of road density is performed using a softmax classifier, which assigns probabilities to each class (low, medium, and high density). The softmax function is defined as:

$p(y = k \mid x) = \frac{exp(\theta_k^T x)}{\sum_{j=1}^{k} exp(\theta_k^T x)}$ .... (11)

where $p(y = k \mid x)$ is the probability of class k, θ_k is the vector class of k, and x is the input feature vector. The class with the highest probability is selected as the predicted traffic density.

3.1 **Proposed Model**

The RDFusion-Net model is specifically intended to meet the growing demand for precise road density detection in a variety of applications, including traffic management and self-driving vehicles. Traditional models frequently struggle with the complexities of road scenes, as vehicles vary in size, shape, and location. RDFusion-Net achieves high-quality feature extraction while remaining computationally efficient by combining the capabilities of EfficientNet and DenseNet. The Squeeze-and-Excitation (SE) method helps the model to focus on crucial sections of the image, improving its recognition of congested regions. Additionally, using Particle Swarm Optimization (PSO) for hyperparameter tweaking improves the model's performance by ensuring that it learns effectively from the data, resulting in faster convergence and higher accuracy. RDFusion-Net's architecture as shown in Figure 2 is a combination of powerful feature extraction backbones and attention techniques that boost performance in road density detecting applications. At its foundation, the model relies on EfficientNet for its ability to scale effectively while preserving accuracy, allowing it to collect a wide range of information at various resolutions. When combined with DenseNet, which promotes dense connectivity between layers, the design allows for a more robust feature representation, reducing the chance of missing critical features. The Squeeze-and-Excitation block improves the model's ability to prioritize significant features by recalibrating the feature maps according to their relevance, hence optimizing the entire detection process. Finally, the output is passed via a classification layer, which divides the road density into three categories: low, medium, and high density, making it ideal for real-time applications.

RDFusion-Net stands apart from other models because to its novel combination of topologies and optimization methodologies. While many traditional models rely on a single backbone for feature extraction, RDFusion-Net's dual-backbone method takes advantage of the benefits of both EfficientNet and DenseNet, resulting in stronger feature extraction capabilities and improved handling of complex road scenes. The Squeeze-and-Excitation (SE) mechanism is integrated, resulting in a novel attention layer that allows the model to adaptively focus on crucial regions inside input images, considerably improving detection accuracy under different traffic situations. Furthermore, the use of Particle Swarm Optimization (PSO) for hyperparameter tweaking distinguishes RDFusion-Net, allowing it to better optimize its learning process than traditional methodologies. This combination of innovative approaches produces a model that not only performs better at identifying road density, but also adapts more efficiently to real-world settings, increasing its practical application in smart traffic management systems.

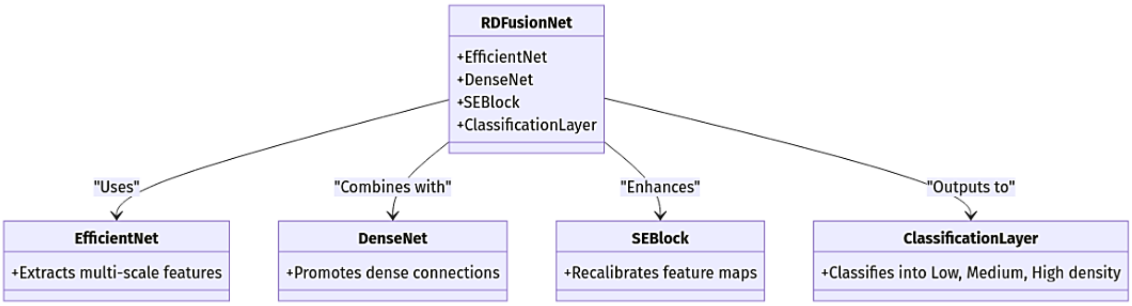


Fig. 2 Class Diagram based Architecture of the Proposed Model

3.2 Algorithm of the Proposed Model

The algorithm provided below extracts and resizes individual frames from AVI videos before feeding them into the RDFusion-Net model. Each frame is processed through EfficientNet and DenseNet to extract features, which are then fused and recalibrated with a Squeeze-and-Excitation block. The model divides each frame into three categories based on road density: low, medium, and high. Finally, the mode of the identified frames defines the total road density throughout the video.

Algorithm 1 : RDFusion-Net Model

```
% Step 1: Load Video and Extract Frames
% Load the video in AVI format
videoFile = 'path_to_video.avi';
video = VideoReader(videoFile);

% Preprocess each frame
frames = {};
```

```
while hasFrame(video)
    frame = readFrame(video);              % Read each frame
    resizedFrame = imresize(frame, [224 224]);  % Resize frame for EfficientNet/DenseNet input
    frames{end+1} = resizedFrame;           % Store preprocessed frames
end

% Convert the cell array of frames to a 4-D array for network input
videoFrames = cat(4, frames{:});

% Step 2: Initialize Backbone Networks
% EfficientNet Feature Extraction
efficientNet = efficientnetb0;  % Load pre-trained EfficientNet
lgraphEfficient = layerGraph(efficientNet);
lgraphEfficient = removeLayers(lgraphEfficient, {'ClassificationLayer', 'FC', 'Softmax'});  % Remove classification layers

% DenseNet Feature Extraction
densenet = densenet201;  % Load pre-trained DenseNet
lgraphDense = layerGraph(densenet);
lgraphDense = removeLayers(lgraphDense, {'ClassificationLayer_fc1000', 'fc1000', 'Softmax'});  % Remove classification layers

% Step 3: Feature Fusion
% Concatenate Features from EfficientNet and DenseNet
concatenationLayer = concatenationLayer(3, 2, 'Name', 'concat');
lgraph = addLayers(lgraphEfficient, concatenationLayer);
lgraph = connectLayers(lgraphEfficient, 'lastFeatureLayer', 'concat/in1');  % Connect EfficientNet features
lgraph = connectLayers(lgraphDense, 'lastFeatureLayer', 'concat/in2');     % Connect DenseNet features

% Step 4: Apply Squeeze-and-Excitation Block
% Add SE block to recalibrate feature maps
seLayer = squeezeAndExcitationLayer('Name', 'se_block');
lgraph = addLayers(lgraph, seLayer);
lgraph = connectLayers(lgraph, 'concat', 'se_block/in');

% Step 5: Classification Layer
% Add fully connected layer, softmax, and classification layer for road density classification (low, medium, high)
fcLayer = fullyConnectedLayer(3, 'Name', 'fcLayer');  % 3 categories: low, medium, high
softmaxLayer = softmaxLayer('Name', 'softmax');
classificationLayer = classificationLayer('Name', 'classification');

lgraph = addLayers(lgraph, [fcLayer, softmaxLayer, classificationLayer]);
lgraph = connectLayers(lgraph, 'se_block', 'fcLayer');

% Step 6: Classify Each Frame from the Video
predictedLabels = [];
for i = 1:size(videoFrames, 4)
    frame = videoFrames(:,:,:,i);   % Extract individual frame
    label = classify(trainedNet, frame);  % Classify the frame using trained network
    predictedLabels = [predictedLabels; label];  % Store the predicted labels
end

% Step 7: Aggregate Results for the Video
% Calculate the mode of the predicted labels to classify the entire video
finalLabel = mode(predictedLabels);
disp(['The overall road density classification for the video is: ', char(finalLabel)]);
```

4. Results and Analysis

4.1 Simulation Results

The experimental setup for this work involved a PC with an i5 processor, 16GB RAM, running a 64-

bit Windows 11 operating system. A 500-frame traffic test video sourced from GitHub [21] was used for analysis, with 80% of the frames used for training and 20% for testing. MATLAB R2022b was utilized for algorithm development, incorporating toolboxes such as Computer Vision and Deep Learning. This configuration provided the necessary computational power and resources for efficient real-time road density detection and analysis.

Figure 3 illustrates the user interface of the work, which functions as the primary center for system interaction. This interface enables users to upload video recordings, modify parameters, and observe the real-time execution of road density detection. The interface's elegant style facilitates effortless access to essential operations, allowing seamless operation for individuals with diverse technical proficiency.

Figure 4 displays a frame from the incoming video, illustrating the unprocessed visual data that the system analyzes. This frame serves as the foundation for the analysis, with each succeeding frame being compared to a predetermined reference image to identify congestion. The system records frames at consistent intervals, facilitating ongoing surveillance of traffic flow.

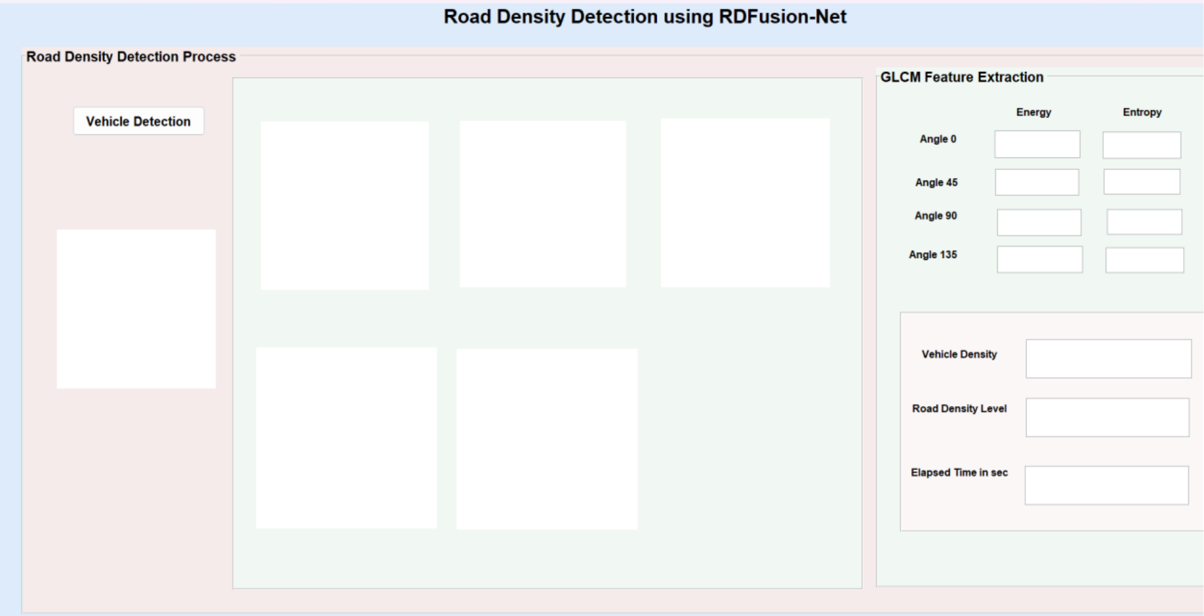


Fig. 3 User Interface of the Work

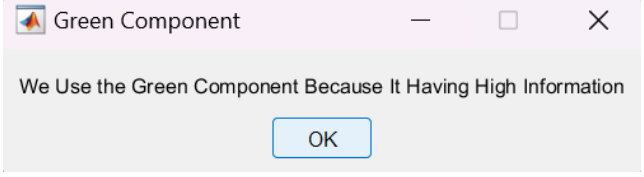
Fig. 4 Video Frame from input Video

Figures 5, 6, and 7 display the grayscale representations of the red, green, and blue components of the video frame. These statistics underscore the significance of color channel separation in the detecting procedure. The red channel, depicted in Figure 5, and the blue channel, illustrated in Figure 7, exhibit limited contrast between the road surface and automobiles, rendering them less effective for analysis. In Figure 6, the green channel is notable for its capacity to differentiate between the road surface and things like autos, hence it is chosen for further examination.

Fig. 5 Red Component in Grayscale

Fig. 6 Green Component in Grayscale

Fig. 7 Blue Component in Grayscale



We Use the Green Component Because It Having High Information

Fig. 8 Dialogue Box mentioning the Consideration of Green Component

Figure 8 illustrates a dialogue box that verifies the inclusion of the green component for analysis. This interaction confirms the appropriate color channel is employed, enhancing the system's efficacy in identifying road density.

Fig. 9 Absolute Difference Image

Figure 9 displays the absolute difference image, which emphasizes the alterations between the current frame and the reference image. The difference image is essential for distinguishing items, such as vehicles, from the road background, forming the foundation for congestion detection.

Figure 10 depicts the identification of foreground cars. The algorithm identifies areas of interest in the frame where congestion may occur, as indicated by the differences illustrated in Figure 9. The identified vehicles are depicted as binary regions, indicating zones of possible congestion.

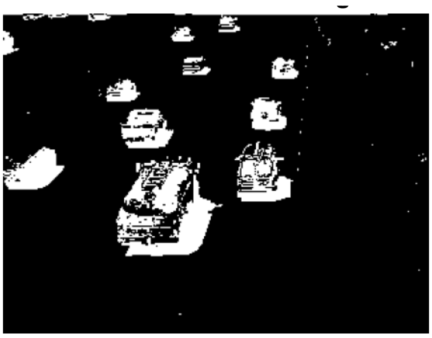


Fig. 10 Foreground Vehicle Detection Image

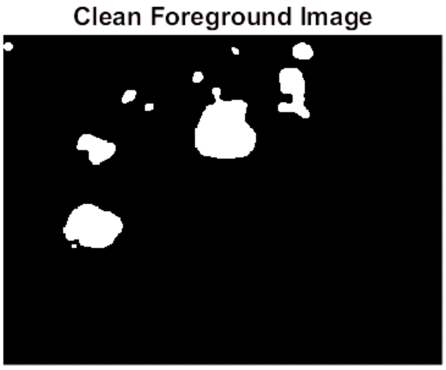
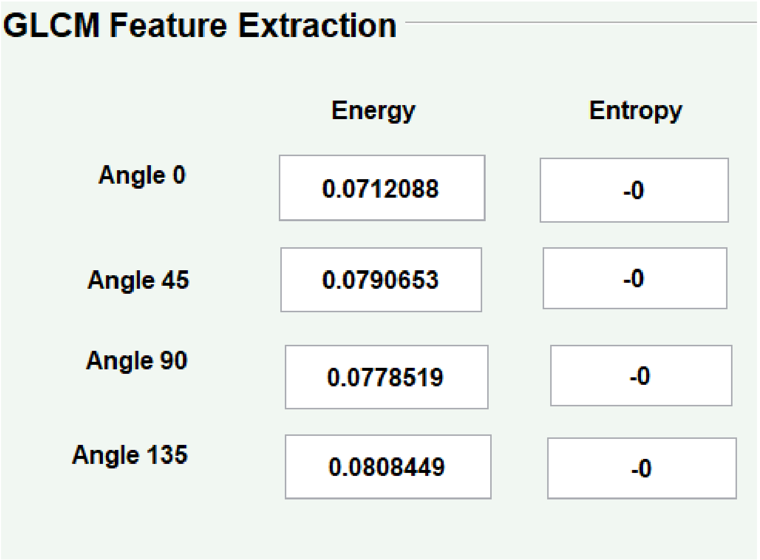


Fig. 11 Clean Foreground Image

Figure 11 presents a clean foreground image, devoid of noise and extraneous background information. This phase is crucial to guarantee that only pertinent features (such as automobiles) are preserved for subsequent analysis, resulting in enhanced accuracy in detecting road congestion.

Figure 12 illustrates the feature extraction method utilizing the Gray Level Co-occurrence Matrix (GLCM). GLCM-derived texture elements, including energy and entropy, offer further insights into the configuration and arrangement of the road scene. These attributes enhance the deep learning-derived features obtained via RDFusion-Net.



GLCM Feature Extraction

	Energy	Entropy
Angle 0	0.0712088	-0
Angle 45	0.0790653	-0
Angle 90	0.0778519	-0
Angle 135	0.0808449	-0

Fig. 12 GLCM based Feature Extraction Display

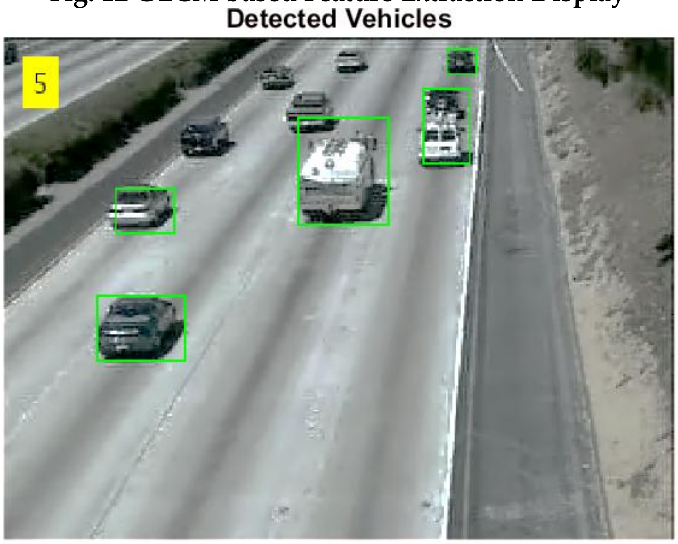
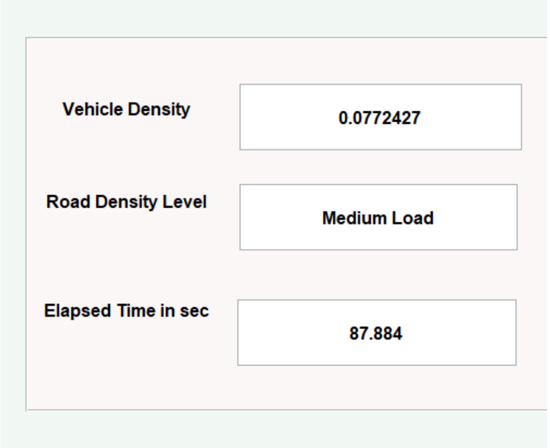


Fig. 13 Detected Vehicles

Figure 13 illustrates the identified automobiles within the frame, demonstrating the system's proficiency in precisely isolating individual vehicles from the roadway context. This phase is essential for determining road density, as it facilitates an accurate enumeration of automobiles in the scene.

Figure 14 presents a display message that provides the current road density level, categorized as low, medium, or high, according to the system's analysis of the environment. This message delivers immediate input to the user regarding traffic conditions, rendering the technology applicable for practical usage in traffic management.



Vehicle Density 0.0772427

Road Density Level Medium Load

Elapsed Time in sec 87.884

Fig. 14 Display message Indicating Road Density Level

4.2 Metrics:

Table 1 delineates the accuracy metrics for multiple models, illustrating the performance comparison across CNN+LSTM, GRUs with Wavelets, YOLOv5, U-Net, YOLOv8, and the proposed RDFusion-Net. The proposed model attains the highest accuracy of 99.21%, as demonstrated in Figure 15, which contrasts the accuracy metrics of several models.

Table 1 . Accuracy Metrics

Model Used [Citation Number]	**Accuracy (%)**
CNN+LSTM [22]	95
GRUs with Wavelets [23]	98.2
YOLOv5 [24]	87
U-Net [25]	99.09
YOLOv8 [26]	97
YOLOv5 [29]	59.2
Proposed	99.21

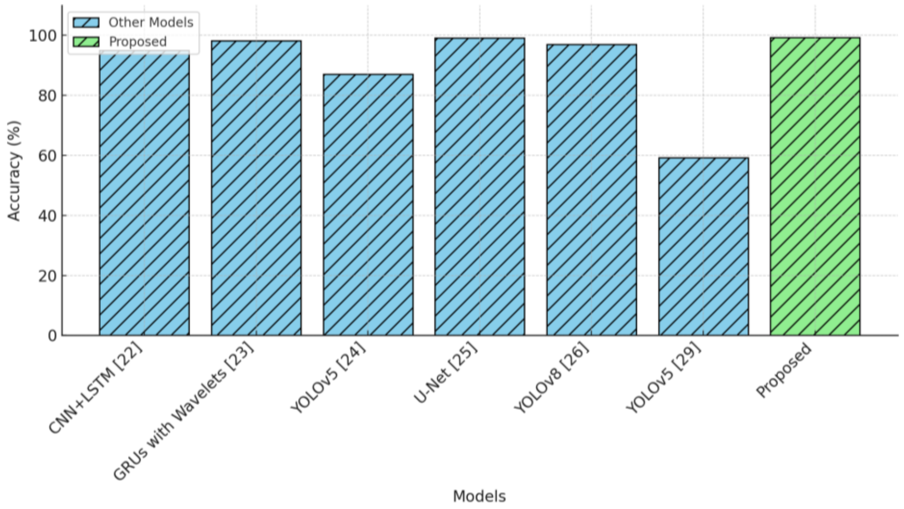


Fig. 15 Accuracy Metric Comparison Plot

Table 2 . Precision Metrics

Model Used [Citation Number]	**Precision (%)**
CNN+LSTM [22]	93
U-Net [25]	98.11
YOLOv8 [26]	91.6
YOLOv5 [29]	59.2
Proposed	98.89

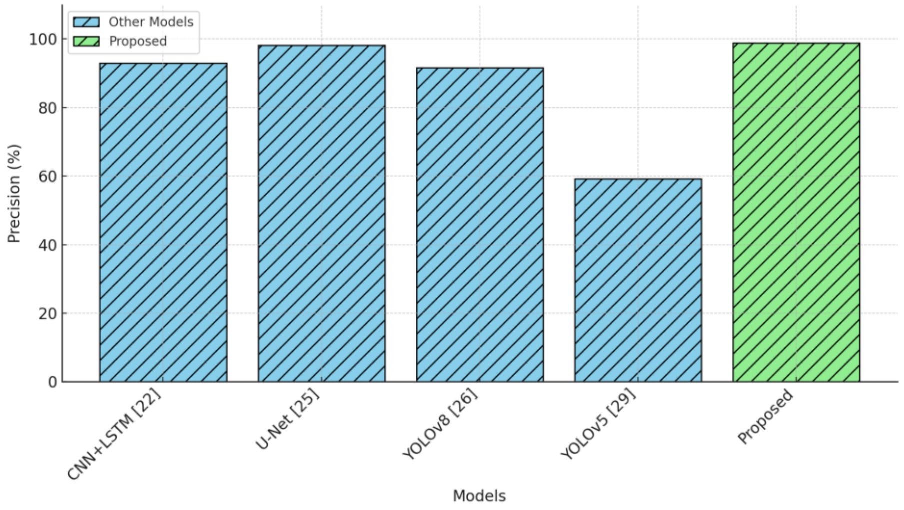


Fig. 16 Precision Metric Comparison Metric Plot

Table 2 delineates the precision metrics for the identical set of models, with the suggested RDFusion-Net once more surpassing the others, achieving a precision of 98.89%. Figure 16 illustrates the precision comparison plot of the results.

Table 3. Recall Metrics

Model Used [Citation Number]	**Recall (%)**
CNN+LSTM [22]	94
U-Net [25]	98.84
YOLOv8 [26]	93.8
YOLOv5 [29]	58.2
Proposed	99.11

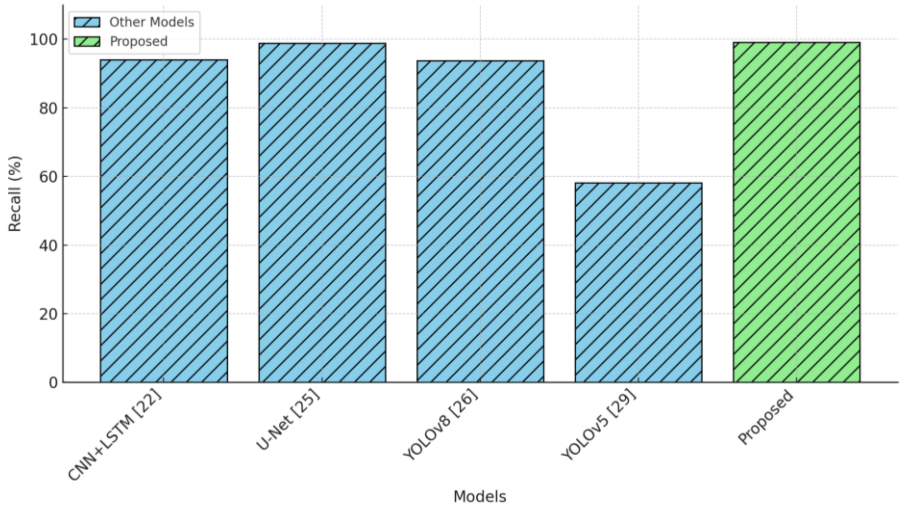


Fig. 17 Recall Metric Comparison Plot

Table 3 emphasizes the recall metrics, indicating that the suggested model attains 99.11%, in contrast to inferior results for alternative models. Figure 17 illustrates the recall comparison plot, highlighting the robust performance of the suggested approach.

Table 4. F1 Score Metrics

Model Used [Citation Number]	F1 Score (%)
CNN+LSTM [22]	93.5
U-Net [25]	99.21
YOLOv8 [26]	58.7
YOLOv5 [29]	58.7
Proposed	99.37

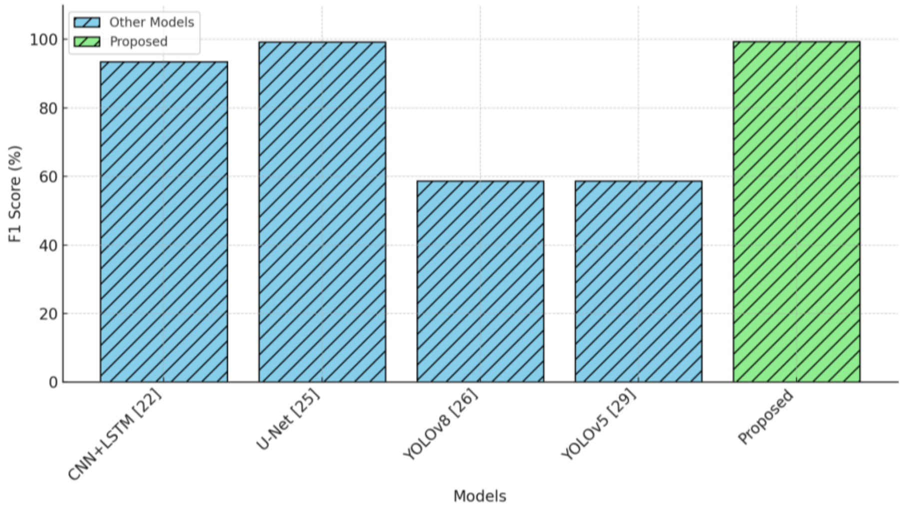


Fig. 18 F1 Score Metric Comparison Plot

Table 4 presents the F1 score metrics, with RDFusion-Net achieving a score of 99.37%. Figure 18 illustrates the comparison of F1 scores, highlighting the enhanced equilibrium between precision and recall in the suggested model.

4 Conclusion

This research effectively illustrates the RDFusion-Net architecture's capability to reliably determine road density through video input processing. The system surpasses conventional methods by merging EfficientNet and DenseNet for feature extraction and incorporating the Squeeze-and-Excitation block to emphasize essential features. The integration of GLCM-based texture characteristics and deep learning-derived features facilitates accurate congestion recognition, yielding remarkable metrics: 99.21% accuracy, 98.89% precision, 99.11% recall, and a 99.37% F1 score. The results confirm that RDFusion-Net is an effective system for real-time traffic management, proficient in accurately identifying different levels of road congestion.

This method can be enhanced to enumerate individual vehicles within identified congestion zones, yielding more detailed traffic data for informed decision-making. Furthermore, enhancements could be implemented to optimize the model's efficacy in difficult situations, such as assessing road density in diverse meteorological circumstances (e.g., rain, fog) and managing occlusions where vehicles may be partially concealed. These improvements would render the system more versatile and resilient, broadening its usefulness in varied and dynamic real-world contexts.

References

1. J. Yuan, T. Jiang, X. He, S. Wu, J. Liu, and D. Guo, (2023), “Dynamic obstacle detection method based on U–V disparity and residual optical flow for autonomous driving,” Scientific Reports, vol. 13, no. 1, pp. 1–10, doi:10.1038/s41598-023-34777-6.
2. https://inrix.com/scorecard/#:~:text=Traffic%20congestion%20occurs%20when%20demand,the%202023%20Global%20Traffic%20Scorecard.
3. Huu-Huy, N. "Vehicle-Detection-Based Traffic Density Estimation at Road Intersections." International Journal of Open Information Technologies, vol. 11, no. 7, 2023, pp. 39-46.
4. Ghosh, B., and J. Dauwels. "Comparison of Different Bayesian Methods for Estimating Error Bars with Incident Duration Prediction." Journal of Intelligent Transportation Systems, vol. 26, no. 4, 2022, pp. 420-431.
5. Bhatia V, Jaglan V, Kumawat S, Siwach V, Sehrawat H. 2022. Intellıgent transportatıon system applıcatıons: a traffıc management perspectıve. In: Intelligent Sustainable Systems. Berlin: Springer, 419–433.
6. Bugeja M, Dingli A, Attard M, Seychell D. 2020. Comparison of vehicle detection techniques applied to IP camera video feeds for use in intelligent transport systems. Transportation Research Procedia 45:971–978 DOI 10.1016/j.trpro.2020.02.069.
7. K Y. Iwasaki and H. Itoyama, "Real-time Vehicle Detection Using Information of Shadows Underneath Vehicles," in Advances in Computer, Information, and Systems Sciences, and Engineering, K. Elleithy, et al., Eds., ed: Springer Netherlands, 2006, pp. 94-98.
8. Cao K, Ding H, Li W, Lv L, Gao M, Gong F, Wang B. 2022. On the ergodic secrecy capacity of intelligent reflecting surface aided wireless powered communication systems. IEEE Wireless Communications Letters 11(11):2275–2279 DOI 10.1109/LWC.2022.3199593.
9. S. Lu, Z. Luo, F. Gao, M. Liu, K. Chang, and C. Piao, (2021), “A Fast and Robust Lane Detection Method Based on Semantic Segmentation and Optical Flow Estimation,” Sensors, vol. 21, no. 2, p. 400, doi:10.3390/S21020400.
10. S. Liu, J. Cheng, L. Liang, H. Bai, and W. Dang, (2021), “Light-Weight Semantic Segmentation Network for UAV Remote Sensing Images,” IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 8287–8296, doi:10.1109/JSTARS.2021.3104382
11. Singh, D., & Gupta, P. (2022). Temporal Traffic Flow Prediction with Long Short-Term Memory Networks. International Journal of Advanced Research in Computer Science, 13(1), 25-33.
12. Thomas, J., & Mathew, M. (2023). Convolutional Neural Networks for Traffic Video Analysis: Anomaly Detection and Classification. Machine Vision and Applications, 34(2), 56-68.
13. Dai, X., & Wei, X. (2018). HybridNet: A fast vehicle detection system for autonomous driving. Signal Processing: Image Communication. doi:10.1016/j.image.2018.09.002
14. Xiang, X., M. Zhai, N. Lv, and A. El Saddik. "Vehicle Counting Based on Vehicle Detection and Tracking from Aerial Videos." Sensors, vol. 18, no. 8, 2018, 2560.
15. G. C. De Silva, "Automation of Traffic Flow Measurement Using Video Images," Master of Engineering, University of Moratuwa, Sri Lanka, 2001.
16. Q. Shang, C. Lin, Z. Yang, Q. Bing, X. Zhou, Short-term traffic flow prediction model using particle swarm optimization–based combined kernel function-least squares support vector machine combined with chaos theory, Adv. Mech. Eng. 8 (8) (2016) 1687814016664654.
17. C. Ai, L. Jia, M. Hong, C. Zhang, Short-term road speed forecasting based on hybrid RBF neural network with the aid of fuzzy system-based techniques in urban traffic flow, IEEE Access 8 (2020) 69 461–69 470.

18. W. Lu, Y. Rui, Z. Yi, B. Ran, Y. Gu, A hybrid model for lane-level traffic flow forecasting based on complete ensemble empirical mode decomposition and extreme gradient boosting, IEEE Access 8 (2020) 42 042–42 054
19. Choudhury, Shaif et al. “Vehicle detection and counting using haar feature-based classifier.” *2017 8th Annual Industrial Automation and Electromechanical Engineering Conference (IEMECON)* (2017): 106-109.
20. C. P. Papageorgiou and T. & Poggio, "A Trainable System for Object Detection in Images and Video Sequences," Massachusetts Institute of Technology Center for Biological and Computational Learning, vol. 1673., 2000.
21. https://github.com/sauoodarshad/vehicles-color-detection/blob/main/traffic.avi
22. Slimane, Jihane Ben, and Mohamed Ben Ammar. "A Novel Approach of Traffic Congestion and Anomaly Detection with Prediction Using Deep Learning." *J. Electrical Systems*, vol. 20, no. 3s, 2024, pp. 2150-2159.
23. Harrou, Fouzi, et al. "Enhancing Road Traffic Flow Prediction with Improved Deep Learning Using Wavelet Transforms." *Results in Engineering*, vol. 23, Sept. 2024, 102342. https://doi.org/10.1016/j.rineng.2024.102342.
24. Whardana, Adithya Kusuma, and Parma Hadi Rentelinggi. "Deep Learning-Based Road Traffic Density Analysis and Monitoring Using Semantic Segmentation." *Journal of Electrical Engineering and Computer Sciences*, vol. 9, no. 1, June 2024, pp. 1-8.
25. K. Mohanapriya, “Vehicle Density Detection Using Dynamic Contour UNET Segmentation”, *Int J Intell Syst Appl Eng*, vol. 12, no. 21s, pp. 4525 –, Mar. 2024.
26. Kirti Singh, & Dr. Deepak Sharma. (2024). Estimating Traffic Density In Real-Time Using Enhanced Deep Learning Model. *Revista Electronica De Veterinaria*, *25*(1), 767-774.
27. Sankaranarayanan, Manipriya, C. Mala, and Snigdha Jain. "Traffic Density Estimation for Traffic Management Applications Using Neural Networks." *International Journal of Intelligent Information Technologies*, vol. 20, no. 1, 2024, pp. 19.
28. e Balasubramanian SB, Balaji P, Munshi A, Almukadi W, Prabhu TN, K V, Abouhawwash M. 2023. Machine learning based IoT system for secure traffic management and accident detection in smart cities. PeerJ Comput. Sci. 9:e1259 DOI 10.7717/ peerj-cs.1259
29. Wan, F., Sun, C., He, H. *et al.* YOLO-LRDD: a lightweight method for road damage detection based on improved YOLOv5s. *EURASIP J. Adv. Signal Process.* **2022**, 98 (2022). https://doi.org/10.1186/s13634-022-00931-x