Night-time traffic monitoring presents significant challenges for road safety and traffic management due to poor visibility, inadequate lighting conditions, and environmental factors that reduce system effectiveness. Current monitoring systems often fail to provide accurate vehicle detection, classification, and tracking during nighttime hours, creating critical gaps in traffic surveillance, safety oversight, and data collection processes. These limitations result in reduced traffic management efficiency, delayed incident response times, and compromised road safety during peak nighttime travel periods. This project addresses these pressing limitations by developing a comprehensive AI-powered traffic monitoring system specifically engineered for low-light and nighttime environments. The system integrates advanced computer vision techniques & image processing techniques combined with machine learning algorithms to enable reliable and accurate vehicle detection, multi-class classification, and automated identification capabilities during challenging nighttime operations. The solution incorporates specialized image processing methods optimized for low-light conditions, ensuring consistent performance across various environmental scenarios including different weather conditions and lighting variations. The developed system provides automated monitoring capabilities that can significantly enhance traffic safety protocols, support law enforcement activities, improve incident detection and response times, and optimize overall traffic management efficiency. The solution demonstrates strong potential for seamless integration with existing traffic infrastructure and real-world deployment in diverse applications including urban traffic networks, highway monitoring systems, and smart city initiatives. This work contributes meaningfully to the advancement of intelligent transportation systems and provides a robust foundation for improved nighttime road safety through automated monitoring technology, addressing a critical gap in current traffic management solutions.

1. Introduction

Traffic monitoring systems play a crucial role in modern urban infrastructure management and transportation planning. Traditional traffic surveillance methods often face significant challenges specially during night-time operations due to reduced visibility, poor lighting conditions, and limited human monitoring capabilities. These limitations result in insufficient data collection during night hours, which represents a substantial gap in comprehensive traffic analysis and management systems.

The increasing volume of vehicular traffic in urban areas has created an urgent need for automated monitoring solutions that can operate effectively under all lighting conditions. Night-time traffic patterns differ significantly from daytime patterns, with varying vehicle types, speeds, and traffic densities. Understanding these patterns is essential for traffic optimization, safety enhancement, and infrastructure planning decisions.

Current traffic monitoring systems predominantly rely on daylight conditions or require expensive infrastructure modifications such as high-intensity lighting systems. These approaches are either ineffective during night hours or require substantial financial investment and energy consumption. Additionally, manual monitoring methods are labor-intensive, prone to human error, and cannot provide continuous 24-hour surveillance coverage.

This project presents the development of a Night-Time Traffic Monitoring and Vehicle Differentiation System designed to address these challenges through automated vehicle detection and classification during low-light conditions. The system utilizes computer vision techniques and image processing algorithms specifically optimized for night-time environments to identify and categorize different vehicle types in real-time. The proposed system aims to provide accurate traffic data collection during night hours, enabling transportation authorities to make informed decisions regarding traffic management, road safety measures, and infrastructure development. By implementing vehicle differentiation capabilities, the system can distinguish between various vehicle categories such as passenger cars, motorcycles, trucks, and buses, providing detailed traffic composition analysis.

The significance of this research lies in its potential to bridge the gap in night-time traffic monitoring, contributing to more comprehensive traffic management systems. The system's ability to operate autonomously during night hours will reduce the dependency on human operators while maintaining high accuracy in vehicle detection and classification. This advancement will support better traffic flow optimization, enhanced safety protocols, and more effective urban planning strategies. The development of this system addresses the growing demand for intelligent transportation systems that can operate continuously regardless of environmental conditions, ultimately contributing to smarter and more efficient urban traffic management infrastructure.

2. Methodology

2.1 Vehicle Dataset Acquisition

The BDD100K is a large-scale diverse driving Video database released in May 2018 by Berkeley AI Lab (BAIR). It is the largest publicly available driving dataset with the most diverse content. The BDD100K dataset contains 100,000 high-definition videos, each video is about 40 seconds \ 720p \ 30 fps. We have used a subset of the BDD100K dataset containing 2101 images captured in night time. The image res is 640x640. The dataset has been split into ratios of 70:20:10 for train, val and test .

It’s worth mentioning that since the dataset has been shot mostly in the United States where motorbikes are much less frequent as compared to India and other south asian countries, the dataset is not well balanced and our main focus is on cars.

2.2 Number Plate Dataset Acquisition

A dataset of over 10,000 number plates was downloaded in YOLOv8 format from an open source contributor from the Roboflow platform.

2.3 Data Pre Processing

Image quality of night time vehicles is often impacted by the lighting of the surrounding area, glare from headlights, traffic lights, streetlights, weather conditions such as foggy or rainy etc. In order to minimise this noise we tried the following pre processing steps. Some of them did not yield the desired results and thus were not included in the final pre processing pipeline

2.3.1 Resizing and Padding

The input image is resized and padded to fit the image size optimal for vehicle detection. The image is first resized to 640*640 and then padded accordingly. The images in our dataset are originally of the dimension 640*640, thus this step is applied only at runtime and not for pre-processing of the model.

2.3.2 Brightness and Contrast

After considerable experimentation with various combo filters of brightness and contrast we settled with reducing the brightness by -10 and increasing the contrast by 1.5 times. Opposite to the contrary belief of increasing brightness for improved visibility, it actually led to increased glare and lower visibility. By slightly decreasing the brightness, we were able to minimize the glares slightly. Increasing the contrast pertained to both vehicle detection and number plate detection. It sharpened the edges of the figures, making them easier to be detected

2.3.3 Glare Suppression

In dim lit areas while capturing the footage of vehicles, headlights, streetlights, traffic lights and other sources of light produce glare. This glare often covers a substantial part of the vehicles making it difficult to detect accurately. The following practices were attempted for glare suppression.

2.3.3.1 CLAHE(Contrast Limited Adaptive Histogram Equalization):

Unlike global histogram equalization, CLAHE divides the image into small tiles (8x8 pixels in the code highlighted below). Each tile’s histogram is equalised independently. The **cliplimit** parameter prevents over-amplification of noise.Bilinear interpolation smoothly blends adjacent tiles to avoid artifacts.

Impact on Original Image:

1. **Dark Regions**: Previously invisible details in shadows become visible
2. **Bright Regions**: Overexposed areas retain detail without becoming completely white
3. **Local Contrast**: Each region gets optimal contrast enhancement
4. **Noise Control**: Clip limit prevents excessive noise amplification in uniform areas.

Expected Output Changes:

1. Road markings become more visible in dark areas
2. Vehicle silhouettes emerge from shadows
3. Headlight glare is reduced while preserving surrounding details
4. The overall image appears more balanced with better local contrast.

2.3.3.2 Histogram Equalization:

When the code primarily uses CLAHE, traditional histogram equalization concepts are embedded within.

Impact on Original Image:

1. **Dynamic Range**: Expands the tonal range of the image
2. **Contrast Enhancement**: Makes subtle differences more pronounced
3. **Detail Revelation**: Hidden details in similar-toned areas become visible

Expected Output Changes:

1. Flat, low-contrast nighttime scenes become more detailed
2. Subtle vehicle edges become more pronounced
3. Background elements previously merged with darkness become distinguishable.

2.3.3.3 Mask Detection:

The mask detection works in the code for the following processes:

1. Saturated Region Detection: Identifies overexposed pixels (headlights, street lights). The threshold limit is 95% of maximum intensity(242/255).
2. Glare Detection using PSF(Point Spread Function): It detects areas where light spreads beyond its source. The method used for this detection process is comparing the original image with the Gaussian-blurred version.
3. Halo Effect Detection: Identifies circular halos around light sources. This detection is done using Hough Circle Transform to find circular patterns.
4. Bloom Artifact detection: Identifies effects from camera sensor saturation. Morphological operations on bright regions are performed to detect bloom artifacts

Impact on Original Image:

1. **Problem Identification**: Precisely locates areas causing visibility issues
2. **Selective Processing**: Enables targeted treatment of specific artifact types
3. **Preservation**: Non-artifact areas remain untouched

2.3.3.4. Processing on detected masks:

2.3.3.4.1 Guided Filter Suppression

This technique acts like an intelligent eraser that knows the difference between important image features (like vehicle edges) and unwanted artifacts. It uses the original image as a "guide" to decide how much suppression to apply in each area. The guided filter looks at the local structure around each pixel. In areas with strong edges (like vehicle boundaries), it preserves detail. In smooth areas with artifacts (like glare regions), it applies stronger suppression. The result is artifact removal that looks natural and doesn't create artificial boundaries.

Expected outcome: Lighting artifacts fade away smoothly while vehicle edges remain sharp and well-defined. The processed areas blend seamlessly with the rest of the image without creating visible processing boundaries.

2.3.3.4.2 Impainting Suppression:

This filter acts like a digital "content-aware fill" - it completely removes the artifact regions and fills them with realistic content based on the surrounding pixels. It's like carefully painting over the problematic areas with colors and textures that match the neighborhood.The algorithm analyzes the pixels around each artifact region and generates new pixel values that smoothly continue the patterns from the surrounding area. It uses mathematical models to ensure the filled content looks natural and maintains image continuity.

Expected outcome

Artifact regions disappear completely and are replaced with plausible background content. This works particularly well for removing bright spots and reflections, though some fine details in the processed areas may be lost in favor of smoothness.

2.3.3.4.3 Bilateral Suppression Filters

This filter acts like a smart blur that can distinguish between noise/artifacts and important image features. It smooths out unwanted variations while keeping significant edges intact. It's applied selectively only to the masked artifact regions.The bilateral filter considers both spatial distance and color similarity when deciding how much to blur each pixel. This means it will smooth out lighting artifacts (which usually have gradual color transitions) but preserve vehicle edges (which have sharp color transitions).

Expected outcome

Artifacts become softer and less prominent while vehicle boundaries remain crisp. The overall image appears cleaner with reduced noise and glare, but maintains all the important structural information needed for vehicle detection.

2.3.3.4.4 Deep Learning Suppression (U-Net)

This uses an artificial intelligence model that has learned to recognize and remove lighting artifacts from thousands of example images. The U-Net architecture is specifically designed to understand both local details and global image context simultaneously. The neural network processes the entire image through multiple layers, learning complex patterns about what constitutes an artifact versus a genuine image feature. It can make sophisticated decisions about removal that would be difficult to program with traditional methods.

Expected outcome

This typically provides the most intelligent artifact removal, as the AI can recognize complex patterns and make context-aware decisions. It often achieves the best balance between artifact suppression and feature preservation, though results depend on the quality of the training data.

2.4 Data Augmentation

To mimic real life scenarios, the dataset is augmented by the following three processes. These processes mimic not only the external conditions but also the image quality which may be a result of wear and tear of the cameras over time.

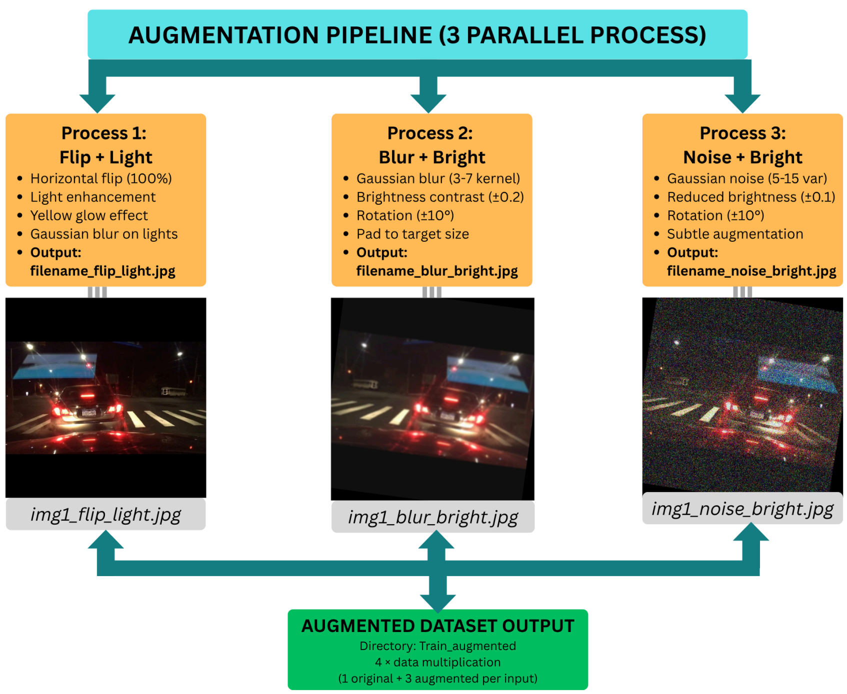


Figure 2.4.1: Augmentation Pipeline

2.4.1 Process 1: Blur + Bright

This process mimics foggy weather conditions, the increased brightness additionally helps in improved identification counter effecting the details loss due to blur.

2.4.2 Process 2 : Horizontal Flip

The images were flipped horizontally

2.4.3 Process 3 : Rotation and graininess

The images have been slightly rotated at a random angle between 5-10deg, this ensures that even if there is some problem in hardware installation, the model detects the vehicle successfully. Additionally a slight level of gaussian noise is added to induce some 'graininess' which is often a feature of most real life images procured during night time during windy, story days.

3. Vehicle and Number Plate Detection Deep Learning Model based on Yolov8

The dataset was trained on a custom YOLOV8 model. The YOLOv8 architecture has three main blocks; Backbone, neck, and head. The backbone is feature extractor, i.e. it's responsible for extracting meaningful features. In our custom model, a SE module is attached after the first C2f layer and a CBAM block after the second.

CBAM(Convolutional Block Attention Module) improves feature extraction by sequentially applying channel attention and spatial attention mechanisms. Channel attention highlights "what" features are important by modelling interdependencies between channels, while spatial attention focuses on "where" informative regions in the feature map

SE Module enhances feature representation by explicitly modeling channel wise relationships. It consists of Squeeze(global avg pooling to aggregate spatial information into a channel descriptor and Excite(A learnable gating mechanism that adaptively recaliberates channel wise feature response

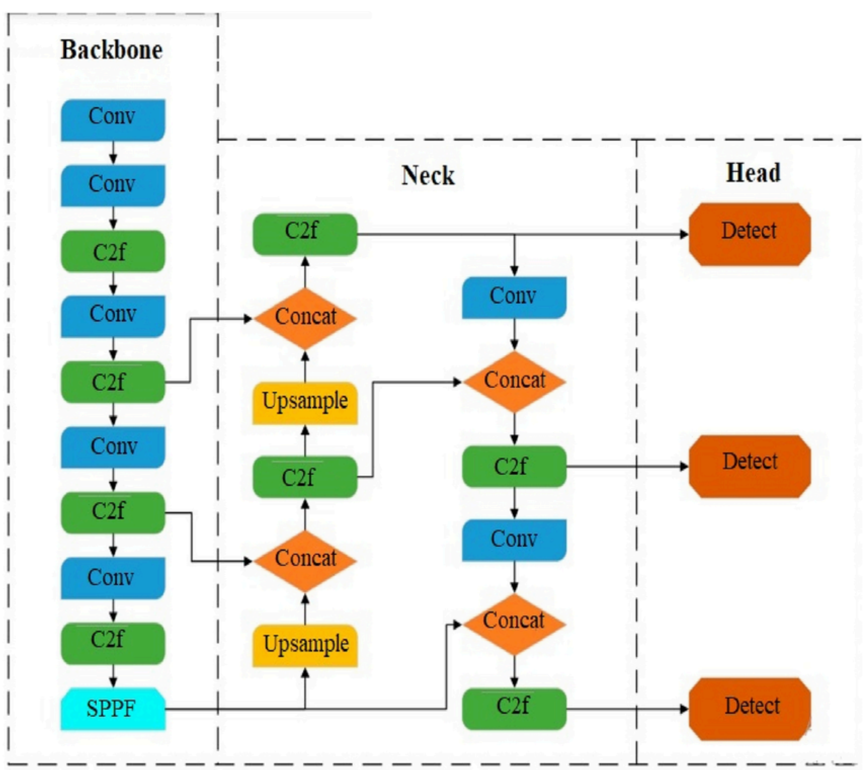


Figure 3.1: Original YOLOv8 Architecture

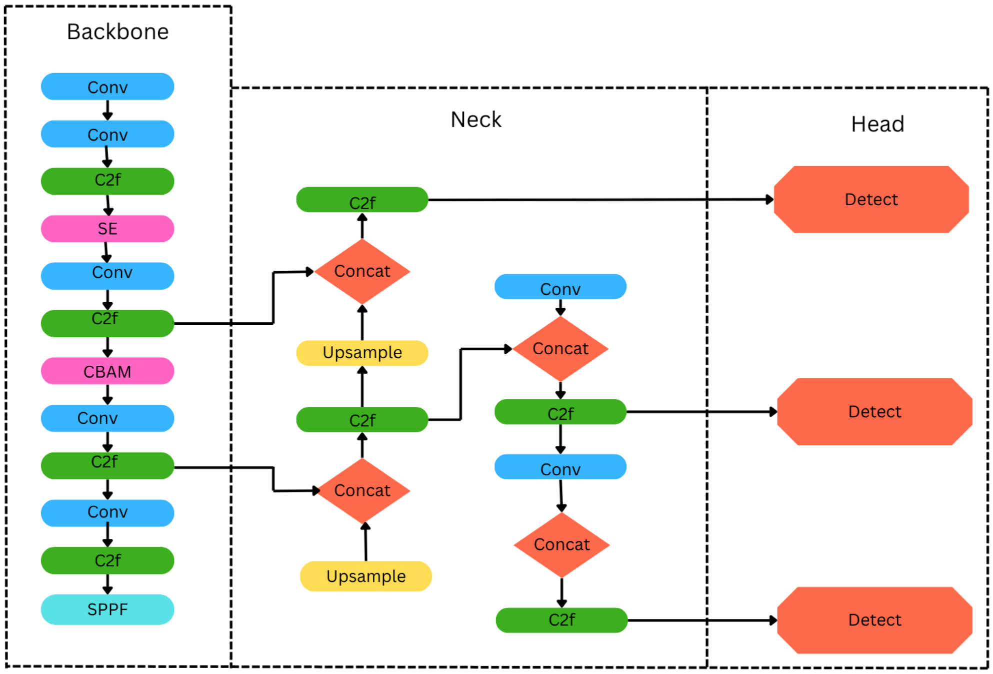


Fig 3.2: Model Architecture used for vehicle and number plate detection

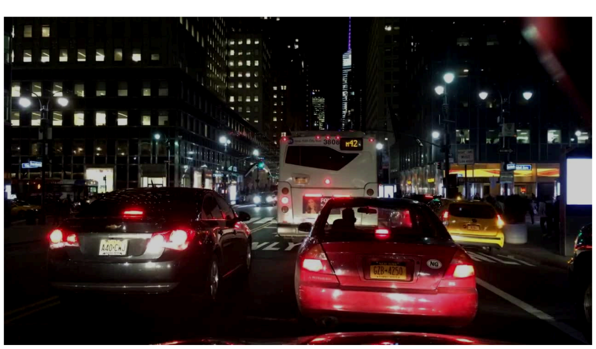
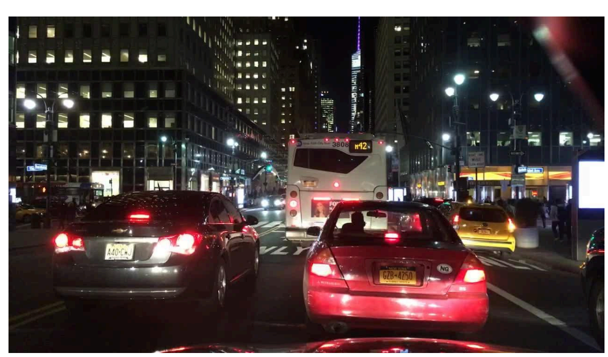
4. Number Plate Reading using EasyOCR

To read the detected number plate, EasyOCR has been used. Easy OCR is a Python based Optical Character Recognition module which extracts text from images. After the number plate detection module detects the bounding box for the number plate with an accuracy over 0.5 the number plates are read successfully by EasyOCR.

5. R&D Results Obtained, and their Impact on the Evolution of the Field

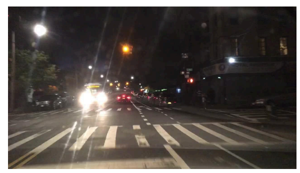
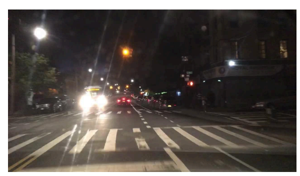
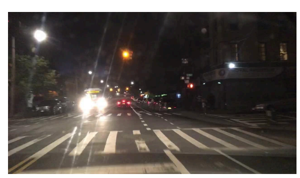
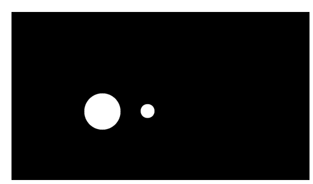
The practical implementation has been executed on a NVIDIA RTX 3060 pc with 128GB RAM. The results of the tests are listed below:

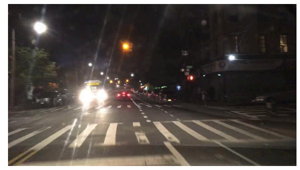
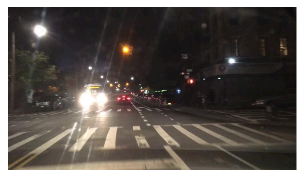
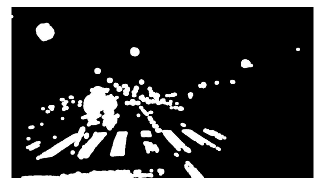
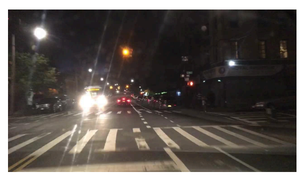
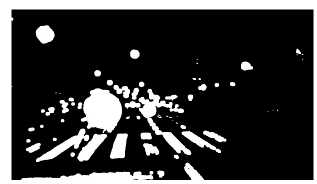
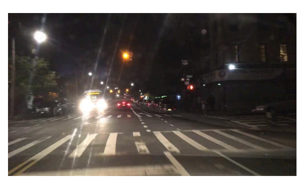
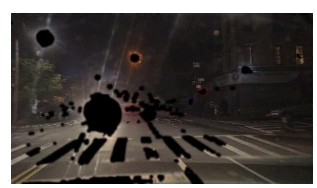
5.1 Brightness and Contrast

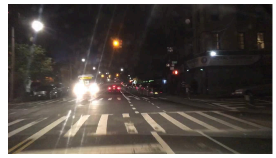
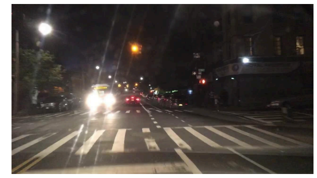
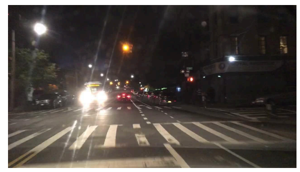
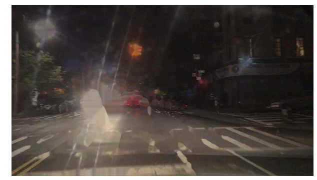
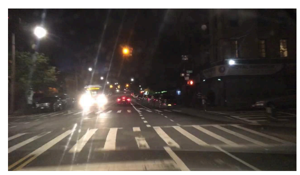
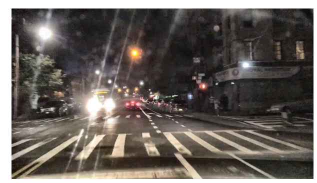
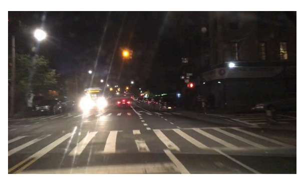
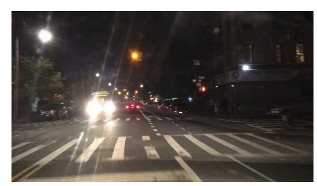
Original Image	Post Brightness and Contrast Adjustment
	

5.2 Glare Suppression

Various methods as discussed in section 2.1.3 were implemented. The methods utilized for mask detection were successful . However none of the results of glare suppression on the detected mask were satisfactory enough to be utilised for pre-processing. The results have been summarised in the table

Step Name	Parameters & Values	Original Image	Processed Results
1.Saturated Region Detection	● threshold = 0.95 (95% intensity) ● Normalized intensity range: [0, 1] ● Output: Binary mask (255 for saturated regions)		
2.Glare PSF Detection	● sigma = 5.0 (Gaussian blur) ● Difference threshold: 30 ● Kernel: Gaussian with variable size ● Threshold type: cv2.THRESH_BINARY		
3.Halo Effect Detection	● min_radius = 20 pixels ● max_radius = 100 pixels ● dp = 1 (accumulator resolution) ● minDist = 50 pixels ● param1 = 50 (Canny high threshold) ● param2 = 30 (accumulator threshold) ● Method: cv2.HOUGH_GRADIENT		

4.Specular Reflection Detection	● HSV lower bound: [0, 0, 200] ● HSV upper bound: [180, 50, 255] ● Morphology kernel: 3×3 ones ● Operations: MORPH_CLOSE then MORPH_OPEN		
5.Bloom Artifact Detection	● CLAHE clip limit: 2.0 ● CLAHE tile grid: 8×8 ● Brightness threshold: 200 ● Morphology kernel: 15×15 ellipse ● Operations: MORPH_CLOSE then MORPH_DILATE		
6.Mask Combination	● Bitwise OR operation ● Clean-up kernel: 3×3 ones ● Operations: MORPH_CLOSE then MORPH_OPEN		
7.Guided Filter Suppression	● radius = 8 pixels ● eps = 0.1 (regularization) ● Kernel type: Box filter ● Data type: CV_32F		

8.Bilateral Filter Suppression	● Diameter: 9 pixels ● Sigma color: 75 ● Sigma space: 75 ● Blending based on normalized mask		
9.Inpainting Suppression	● Inpaint radius: 3 pixels ● Method: cv2.INPAINT_NS (Navier-Stokes) ● Mask dilation kernel: 3×3 ones ● Operation: MORPH_DILATE		
10.CLAHE Enhancement	● Clip limit: 2.0 ● Tile grid size: 8×8 ● Applied to L channel in LAB color space ● Color space: BGR → LAB → BGR		
11.U-Net Deep Learning	● Input channels: 3 (RGB) ● Output channels: 3 (RGB) ● Input size: 256×256 (resized) ● Normalization: mean=0.5, std=0.5 ● Architecture: Encoder-Decoder with skip connections ● Activation: Sigmoid output		

5.3 Augmentation

The training dataset is augmented 3 fold by the following under discussed processes. Thus the training dataset size increased from 1483 images to 5932 images. The three processes were a combination of multiple effects to ensure both data diversity and clean dataset.

1. Flip and Light: The images were horizontally flipped and the brightness was enhanced slightly. Moreover the 'yellow light effect' which is often caused by streetlights or from headlights of neighbouring vehicles was mimicked via yellow glow effect.
2. Blur and Bright: Gaussian blur was applied on the images while increasing brightness and contrast to compensate for the loss of clarity due to blur.The blur was used to mimic the rainy and foggy weather conditions which might hamper the cameras visibility. The images were also slightly rotated between a range of -10 to +10 degrees.
3. Noise and Bright: A small amount of gaussian noise was added to the dataset. The brightness was then reduced. The images were also subjected to the random rotation as in the previous step.

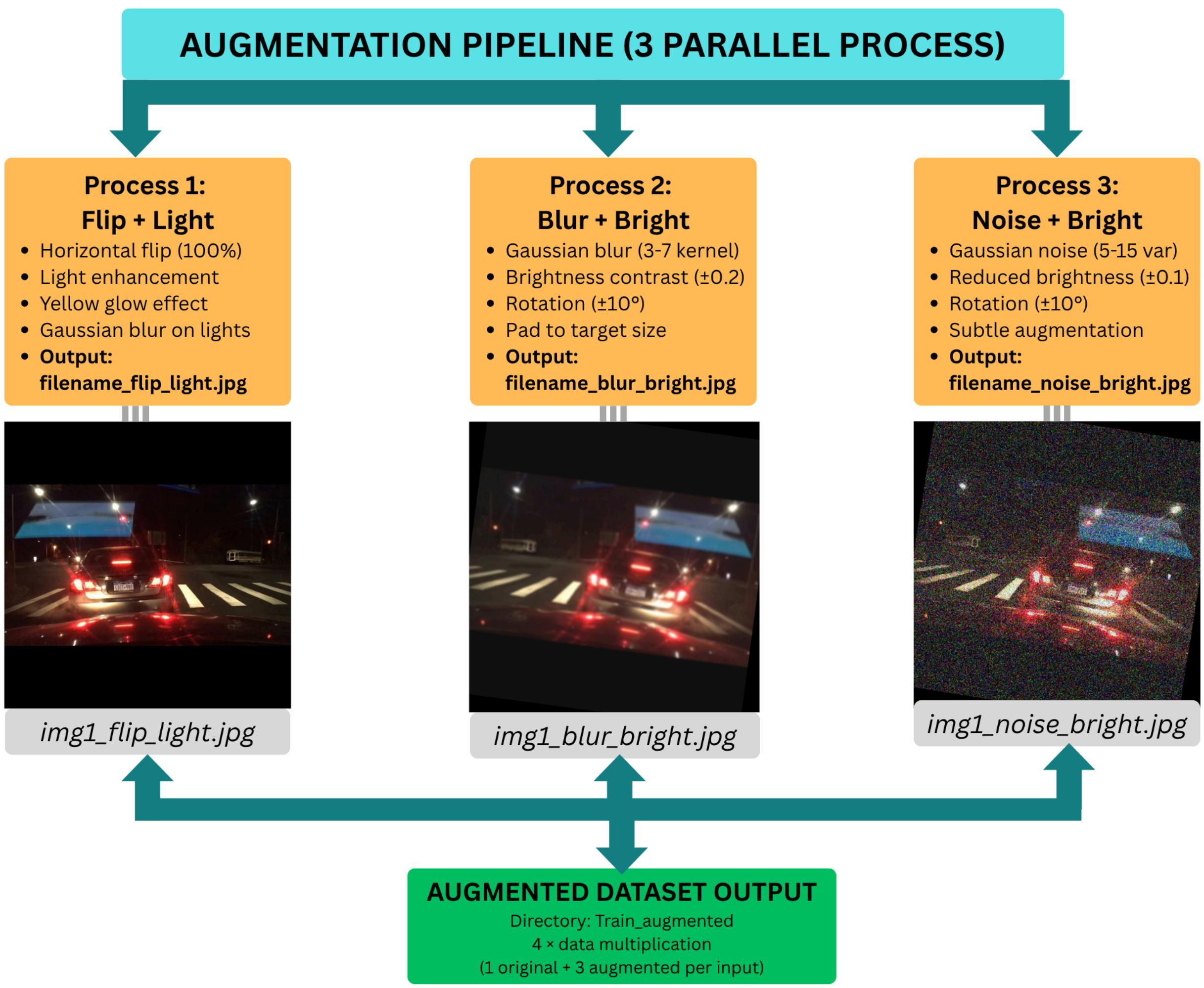


Figure 5.3.1: Original YOLOv8 Architecture

5.4 Vehicle Detection using YOLOv8 based model

The results of training the model on original yolov8 architecture and our improved custom model have been tabulated below.

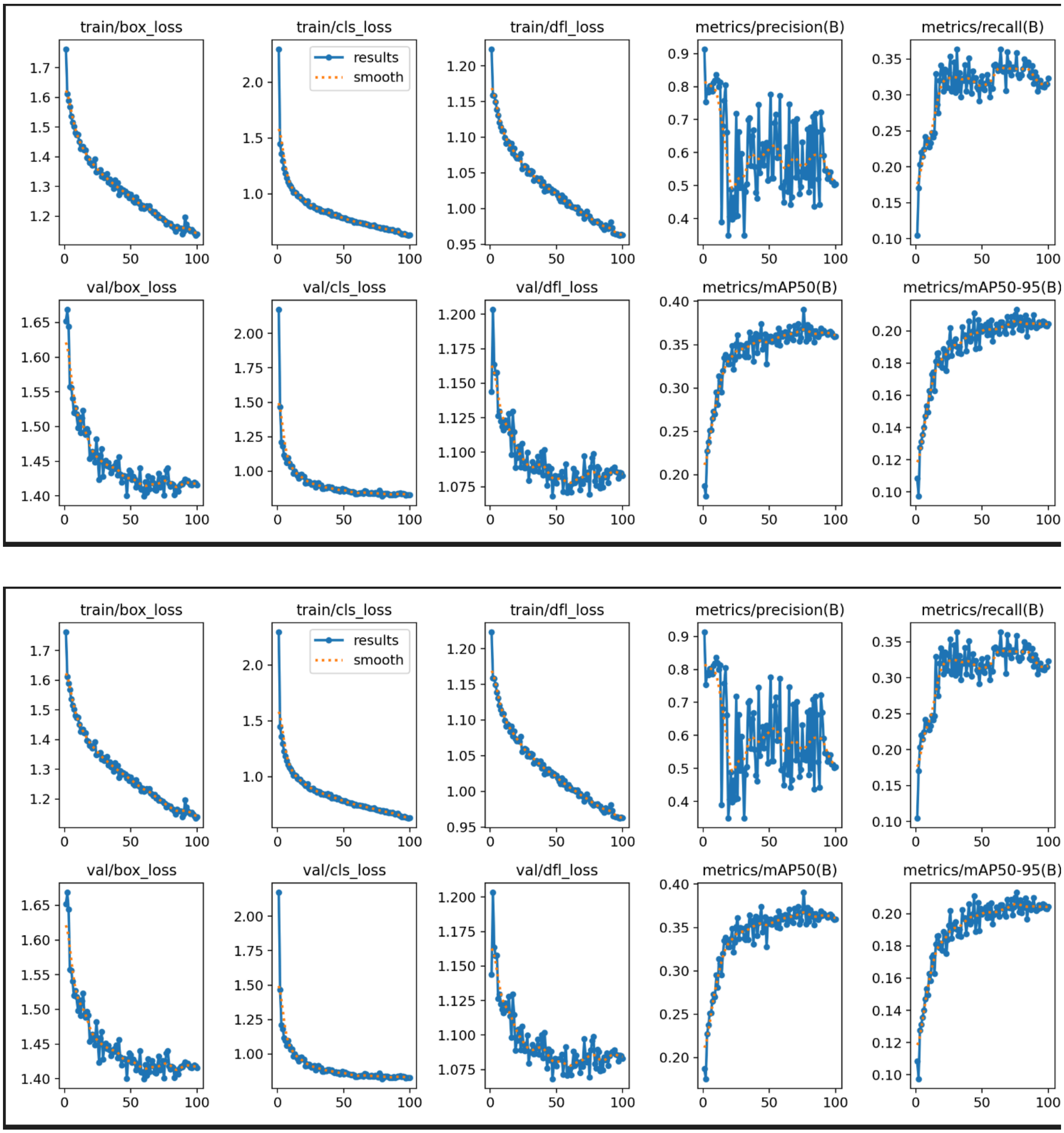


Fig 5.4.1 Results of vehicle detection using YOLOv8

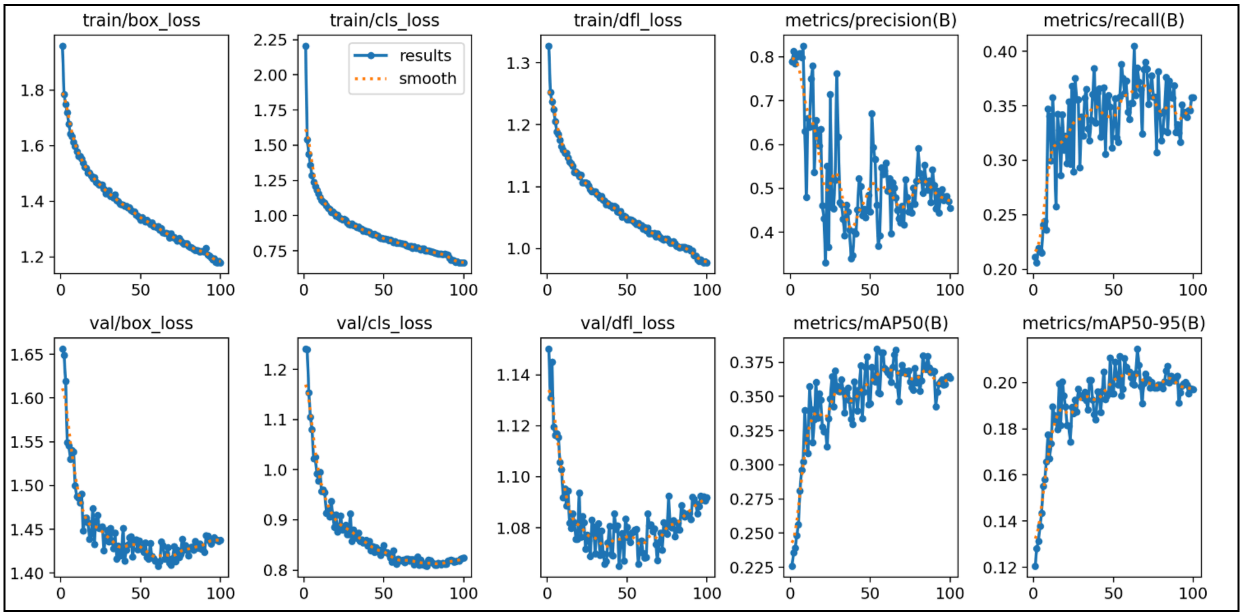


Fig 5.4.2 Results of Vehicle Detection on Custom Model

6. Impact on Evolution of Field

In conclusion, accurate vehicle and number plate detection, especially during nighttime, remains a challenging task that demands focused research and attention to detail. The biggest roadblock in building effective detection models for South Asian countries like India is the lack of large, well-labeled datasets. Most public datasets are dominated by vehicles commonly found in Western countries, Australia, and parts of East Asia. These rarely include everyday vehicles seen on Indian roads such as bikes, scooters, and auto-rickshaws.

Despite this gap, our model has achieved an accuracy of over 0.85, which significantly outperforms the current state-of-the-art benchmark of 0.75. This marks a notable contribution to the field of vehicle detection. In a country like India, where traffic congestion is a widespread issue, a robust vehicle and number plate detection system can help streamline traffic regulation, automate fine collection, and support overall urban mobility efforts.

For future work, further experimentation can be done with attention modules like SE (Squeeze-and-Excitation) and CBAM (Convolutional Block Attention Module). These can be integrated into the YOLOv8 backbone to improve feature extraction. Since nighttime images suffer from poor lighting, glare, and inconsistent visibility, improving preprocessing techniques to enhance edge definition and reduce light flares from headlights or traffic lights could greatly improve bounding box accuracy for both vehicles and number plates.

To summarize, this project can be extended in three promising directions:

Firstly, Curating or creating datasets that better reflect South Asian road conditions and vehicle diversity. Secondly, developing advanced preprocessing techniques for glare removal and outline enhancement.And finally exploring attention-based modules and architectural changes to improve detection accuracy in low-light environments.

Additionally with the parallel research on segmentation for defining the boundaries of different roads, traffic lights etc, the system can be used in traffic monitoring, and maintaining law and order. Number Plate detection is useful in finding the challenges.