IJITCE

ISSN 2347- 3657

International Journal of Information Technology & Computer Engineering

Email : ijitce.editor@gmail.com or editor@ijitce.com

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2, 2025

DeepRoute: Smart Traffic Optimization with AI-Powered Insights

Mrs. Rayala Pushpalatha
Department of Computer Science & Engineering
Vignan's Lara Institute of Technology & Science
rplatha307@gmail.com

Boppudi Likhitha
Department of Coputer Science & Engineering
Vignan's Lara Institute of Technology & Science
boppudilikhitha03@gmail.com

Gottimukkala Deekshitha
Department of Coputer Science & Engineering
Vignan's Lara Institute of Technology & Science
deekshi0511@gmail.com

Badugu Vidyasagar
Department of Coputer Science & Engineering
Vignan's Lara Institute of Technology & Science
vidyasagarb967@gmail.com

Gubbala Sasidhar
Department of Coputer Science & Engineering
Vignan's Lara Institute of Technology & Science
dharsasi10@gmail.com

Abstract: Traffic congestion, which has become rampant in recent times, has fused itself in every nook and corner of cities. The cities appear to be heavily impacted by it further. Road Traffic density should be known in real-time to ease signal control and facilitate efficient traffic management due to its bigger nature. Some causes of Traffic Jams include: Insufficient capacity, excessive demand, delays in red light, etc. However, insufficient capacity and infinite demand have some connection, but there are hard-coded delays for each light and traffic independent. Henceforth, traffic management must improve and optimize the simulation to build it in an advanced way that satisfies the increasing demand for traffic management measures. Traffic Management Imaging based processing and monitoring systems have become visible for real-time information, lamp measures, and updates to travelers recently. Image processing can also measure traffic density. The live images acquisition procedure from the camera will be described in this project. For real-time traffic density with image processing, cameras are placed in traffic migration. It also incorporates traffic-light algorithms, which switch to a particular road according to the density of vehicles on it. It is devised to minimize traffic congestion and reduce the number of road accidents. This allows harmless passage to human beings, reduces fuel consumption, and cuts time spent waiting. An extensive data base is also available for future planning and evaluation of roads. Over much more advanced stages, three or more signals in traffic can then confer with one another to relieve traffic and blockage in traffic. No other devices are used mounted on the road - actually processed by the system itself - and only photographed vehicles recognize them before counting. Cameras near an intersection will be the area to produce an image series. Changes in traffic light conditions could also be controlled on the basis of image processing easily. Thus, under such conditions, green light on empty roads means that time wasted can be reduced and traffic jams can be reduced. The use of real photos of the traffic condition further enhances reliability in estimating vehicle presence. Instead, it is very likely much better than system deployment based on detecting the metallic content of vehicles

Keywords—Vehicle Recognition, Road User Detection, yolo Method, Traffic Engineering

Introduction

Over the past decade, ambitious technologies such as mobility, cloud computing, social platforms, and real-time data analytics have changed the way people live, work and access them. From mobile banking to online booking for cinema cards, this technology is inseparable from everyday life and offers comfort, efficiency and comfort. We are researching intelligent cities and intelligent communities, but the real challenge lies not only in introducing new technologies, but also in developing intelligent services that improve quality of life. One of the most urgent urban challenges today is traffic management, which impacts millions of commuters every day. Rush hours are discarded, fuel consumption is increased, air pollution is increased, and stress levels are added. These problems have mainly been a consequence of conventional traffic management owing to outdated traffic management static signal systems. Most systems are based on timers and do little to adapt to the momentary flow of traffic; thus, they cause many unnecessary delays and poor traffic distribution. Badly designed intersections and lack of actual data continue to be an overload issue. Rapid urbanization and increased vehicle possession make the situation worse. For example, Chandigarh, an Indian union region, holds the highest vehicle density record in the country. Every day, above 45,000 registered vehicles tend to tarnish the roads of Chandigarh. The street infrastructure of the city is struggling to manage the increased speed of traffic flow. Because of this, commuters are experiencing a lot of inconveniences, especially within the main lessons, particularly inside the city's internal sector, where long queues are formed by vehicles. To tackle this escalating issue, our project proposes a Smart Traffic Management System that reduces overload using real-time image processing and analysis. Surveillance cameras installed at critical intersections record live traffic materials analyzed to calculate vehicle density. Based on these actual data, traffic times are dynamically configured, which corresponds to the actual flow of traffic and replaces inefficient solid cycles. This adaptive real-time mechanism improves traffic efficiency, reduces unnecessary waits, and compensates for vehicle loads through the network. Additionally, the system can provide drivers** intelligent traffic warning** and** alternative route suggestions**. This reduces overload. By integrating real-world data analysis, image processing and augmented analytics, the system aims to provide a smoother, faster, and predictable travel experience, while simultaneously reducing fuel consumption, air pollution and commuting stress. Through these innovative

ISSN 2347–3657
Volume 13, Issue 2, 2025

solutions, intelligent cities can change the mobility of cities and provide safer and environmentally friendly transportation systems for everyone.

I. PURPOSE

Road traffic is currently one of the most basic and most common means of transportation in the world. The number of vehicles on the street grows rapidly every day. As a result, urban traffic congestion is inevitable. Lower traffic management leads to waste time, increased pollution, increased fuel consumption, increased transportation costs and stressful drivers. All of these traffic rat races have been barely controlled by a single density-based traffic control system. Designing the system with such good features in it should solve these feasible conditions of traffic accident avoidance, minimize collision sufferings, and control the traffic jams. Indeed, intelligent transport management systems in the city would be collaborating with newly developed models that might be an important source of smooth management of traffic through data analytics. The models would provide real-time analysis from these sources and combined with some trends left more roads opened for traffic flow management avenues.

II. PROBLEM STATEMENT

The research study is basically based on the development of intelligent traffic management systems using image processing online. The intention is to analyse the live images from surveillance cameras set up by local authorities in strategic positions along urban road networks, using state-of-the-art computer vision technology. It proposes to extract critical pieces of information, including vehicle density, flow, anomalies, etc. With these, the system intends to optimize overall traffic management, making it highly efficient, less congested, and more secure from accidents. Urban traffic management is a major issue today, and the application of intelligent traffic management systems is desperately needed. Urban traffic flow is very complex, and as a result, traffic congestion, longer travel times, and an increased risk of accidents are common.

III. PROPOSED SYSTEM

It consists of a preferred Yolo model algorithm for predicting vehicle traffic congestion. These algorithms are used to track, identify and count many types of cars. We recommend calculating the number of vehicles in advance and alternative methods to reach the vehicle. It requires solely a neural network for the complete image view. Acceptance of net-folding neurons corresponds with the input video sequence. The entire training procedure had been using the budding structure based yoga algorithm of neural networks. Most tracking algorithms begin at the first input: spatial detection of objects inside a video frame. In our solution, the film image rate is 45 frames per second, and the objects are segmented using rectangular regions (ROIs). This provides a lie model frame for counting, capturing and pursuing the pal off. The object detection algorithm works on all frames. Finally, count the entire vehicle. If the number of vehicles is below the threshold, it is a normal traffic signal circuit. Otherwise, the number of vehicles is a more alternative route to reduce time power. Detecting the wrong car is the last step. Our system defined the vehicle being recognized as a fake vehicle when it escapes from the camera. Assume the vehicle is in the camera and is correct. After detection, an incorrect vehicle will automatically record an image of the frame. The use of images of recorded images is treated as a fake vehicle

IV. ARCHITECTURE

A. *Input Layer*

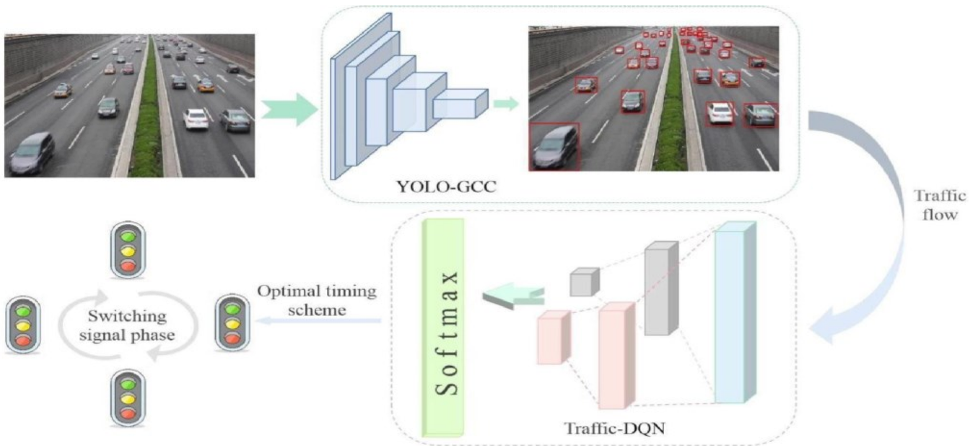
The input layer takes traffic images captured by cameras positioned across urban road networks. These images serve as the raw input for the YOLO system.

B. *Image Preprocessing*

- Before sending the images to the Yolo models, one can use the previous image processing steps, such as resizing and normalizing for sameness to maximize model output.

C. *YOLO Model Backbone*

The deep backbones use neuronal networks mostly in European architectures, such as darknets and Yolov3, which consist of various levels of folding. The levels are tasked with collecting properties, meaning hierarchical types from the input images.



An architecture of the proposed PDABSA task that improves ABSA by prompt-based learning

D. Head of Object Recognition

This describes the actual network whose backbone properties evoke or learn bounding boxes, class probability, and confidence values for each detected object. The Yolo structure gives a grid-like structure to the detection of objects at different scales within your image.

E. Bounding Box Regression

It makes improvements in the alignment of the original bounding box, leading to much higher accuracy for object localization. It is most vital for the precise orientation of the recognized objects in a traffic scene vignette.

F. Non-Maximum Suppression (NMS)

This should be done as post process to shoot down the super bounding boxes for keeping the most trusting outputs. The NMS operates in such a manner that only retains the most accurate and relevant detection.

G. Output Layer

It's the start-end layer which gives out the final results including bounding box coordinates, the related class name vehicles, pedestrians, etc. and trust reviews depicting the certainty in predictions.

H. Integration with Traffic Management System:

Recognized objects act on creating decisions in real-time by integrating the outcomes of Yolo into a broader traffic control system.

D. IMPLEMENTATION

a. Image Acquisition

An image is a two-dimensional function incised at every point x on a plane expressed as f(x, y). That position can be taken to imply the grey level of an image, and hence, the amplitude of an image at that moment can be defined by f. However, to digitize the image, we need to convert these x and y coordinates into finite discrete values. The input image is a fundus from stare data base and drive database. It takes a picture of the retina for processing, in order to check on people's conditions. For analyzing images of the analog type on a digital computer, it becomes necessary to convert them into digital images. Each image is digitalized into a defined number of elements which in turn is digitalized into pixels, as each element is considered pixel.

b. Formation of Image

Being a system property on which energy is emitted by a physical source, the image value has different conditions to form an image f(x, y) which implies that f(x, y) cannot be zero or infinite,

i.e. $0 < f(x, y) < \infty$.

c. Change in image size:

Expression:
blob = cv2.dnn.blobfromimage(picture, scalefactor = 1.0/255, size = (416,416), swaprb = true, crop = false)
Description: Size of input image with fixed dimensions of 416x416 pixels. Yolo requests a specific size for photos in a specific format to make the detection

d. Normalization

Expression:
blob=cv2.dnn.blobfromimage(picture, scalefactor=1.0/255, size=(416,416),swaprb=true, true, false)
Description: Normalizes to [0,1]. 255 is used to divide by pixel value for training and its subsequent performance improvements during inference

e. Color Channel Swapping:

Expression:

blob=cv2.dnn.blobfromimage(picture, scalefactor=1.0/255, size=(416,416), swaprb=true, crop=false)
Description: Swapping of red and blue channels. OPENCV loads images in BGR format, while Yolo is expected in RGB format. The swaps take place for ensuring the Yolo receives the correct input format.

f. Non-maximum Suppression (NMS):

Formal statement: indices=cv2.dnn.NMSBoxes(out_boxes, confidence, args["Confidence"], args["threshold"])
Description: Whenever an object (in this case, the car) is detected, NMS suppresses all the bounding boxes of the same object. Keeps certain detections and deletes duplicate boxes.

g. Bounding Box Drawing:

Formula CV2.Rectangle (photos, (x, y), (x+ w, y+ h), color, 2) is used to draw rectangles in open cv photos. It represents primary colors, three main colors, three primary colors, three primary colors, three primary colors, three primary colors, three primary colors, three primary colors, and blue. Our eyes recognize colors based on how three different types of cones in the retina respond to light of different wavelengths. That’s why color images are typically stored using three separate matrices, one for each color: red (R), green (G), and blue (B), which together form the RGB format In grayscale images, there's no differentiation between these colors; the same amount of light is emitted across all channels, meaning that the image reflects variations in brightness rather than color. The brighter the pixel, the more light is emitted, and the darker the pixel, the less light it emits. When you convert an RGB picture from the format into a black and white image, you take the RGB value of each pixel and form a summary number by which that pixel will be recognized as the brightness of it. One way is to calculate this averaged contribution from every channel: (R+B+C)/3. In other terms, the so-called "human way" involves considering a weighted average since actually, the perceived brightness is predominantly influenced by the green factor. E.g.: 0.3R + 0.59 G+ 0.11B.

h. Edge Detection

Edge detection is a technique that is usually used to find areas of digital images where brightness changes suddenly due to sharp transitions or noise. These points are usually divided into curve segments called edges. Edge detection is extremely important for image processing, machines, and computer

vision. It is especially useful to highlight and extract important features from photos.
Below are the different edge-detection methods :- Sobel Edge Detection Technique Perwitt Edge Detection
For our project, we've chosen the "Canny Edge Detection Method" because it provides more benefits than other methods like Roberts and Zero Cross Threshold. Its precision in detecting edges and minimizing noise makes it a more reliable option for our work.

I. Canny Edge Detection

Indeed, the Canny edge detector is one of the really popular algorithms for edge detection in general image processing and works very vigorously. The process denoted by this name consists of several sequential stages which can actually be seen in terms of a series of filtering operations done on the GPU. The Canny edge-detection method truly rests on the three key objectives.

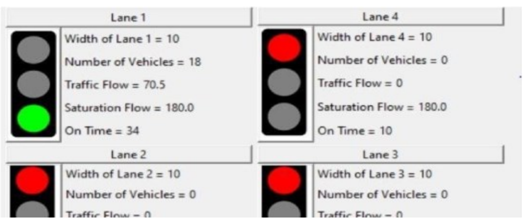
J. Image Matching

View each category using a consistent identification method prototype pattern. To classify unknown patterns, the system compares it with a prototype and selects the closest one based on a particular scale. The minimum distance classifier is a simple method. The distance between an unknown pattern and each reference point is calculated in terms of Euclidean distance, with the pattern assigned to the category at which the shortest distance occurs. The intuitive correlation-based approach can be straightforwardly applied to the image. However, in terms of image manipulation, I completely took another route.

E. SAMPLE OUPUT

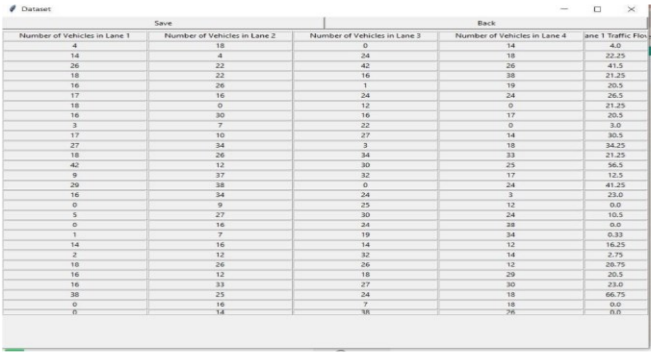
A. USER INTERFACE

This allows a simple interaction between the users and the application using the Flask framework in Python. This allows the user to select input files from the file system for easy file handling



Create Dataset:

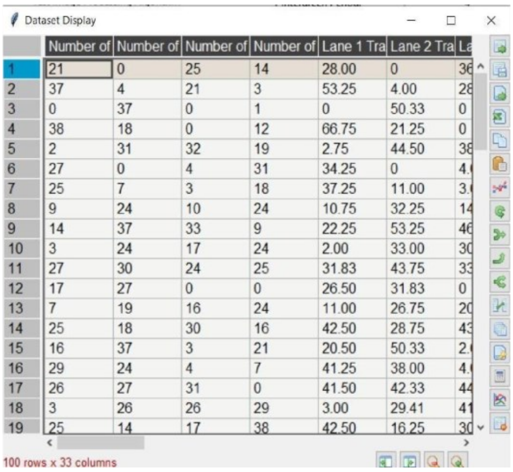
This Create Data Record option creates a dataset for transpo rt analysis as input to the algorithm.



Dataset

Save | Back

Number of Vehicles in Lane 1	Number of Vehicles in Lane 2	Number of Vehicles in Lane 3	Number of Vehicles in Lane 4	ane 1 Traffic Flo
4	18	0	14	4.0
14	4	24	18	22.25
26	22	42	26	41.5
18	22	16	38	21.25
16	26	1	19	20.5
17	16	24	24	26.5
18	0	12	0	21.25
16	30	16	17	20.5
3	7	22	0	3.0
17	10	27	14	30.5
27	34	3	18	34.25
18	26	34	33	21.25
42	12	30	25	56.5
9	37	32	17	12.5
29	38	0	24	41.25
16	34	24	3	23.0
0	9	25	12	0.0
5	27	30	24	10.5
0	16	24	38	0.0
1	7	19	34	0.33
14	16	14	12	16.25
2	12	32	14	2.75
18	26	26	12	20.75
16	12	18	29	20.5
16	33	27	30	23.0
38	25	24	18	66.75
0	16	7	18	0.0
0	14	38	26	0.0



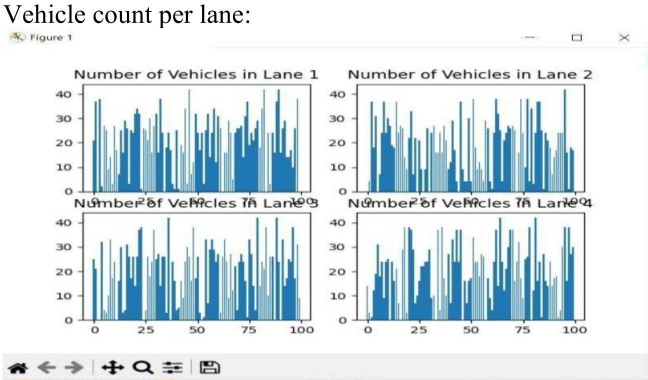
Dataset Display

	Number of	Number of	Number of	Number of	Lane 1 Tra	Lane 2 Tra	La
1	21	0	25	14	28.00	0	36
2	37	4	21	3	53.25	4.00	28
3	0	37	0	1	0	50.33	0
4	38	18	0	12	66.75	21.25	0
5	2	31	32	19	2.75	44.50	38
6	27	0	4	31	34.25	0	4.
7	25	7	3	18	37.25	11.00	3.
8	9	24	10	24	10.75	32.25	14
9	14	37	33	9	22.25	53.25	46
10	3	24	17	24	2.00	33.00	30
11	27	30	24	25	31.83	43.75	33
12	17	27	0	0	26.50	31.83	0
13	7	19	16	24	11.00	26.75	20
14	25	18	30	16	42.50	28.75	43
15	16	37	3	21	20.50	50.33	2.
16	29	24	4	7	41.25	38.00	4.
17	26	27	31	0	41.50	42.33	44
18	3	26	26	29	3.00	29.41	41
19	25	14	17	38	42.50	16.25	30

100 rows x 33 columns

Data Visualization:

Vehicle count per lane:



F. RESULTS

Density Traffic Analysis: The analysis can also be extended to specific traffic days captured from the image processing module and across lanes. The analysis can be made in a form of histograms and heating caps, displaying the variation within traffic density.
Traffic Flow Management: This gives a complete description of how well the traffic flow management system works according to the code you have written. Comparatively, gross discrepancies and inconsistencies between the real flows of

traffic and the saturated flows will be discussed and compared here.
Ambel Optimization: You could analyze the performance of traffic light optimization algorithms by checking the timely phase of each traffic light.

G. Conclusion

Better flow of traffic: Smart traffic management systems will have real-time data from cameras, sensors, and connected vehicles to make the most of traffic time, lane track allocation, and route planning. This will minimize traffic congestion and delays. All these improvements will lead to smoother traffic movement for commuters and shorter times spent traveling for more efficient travel.

Strengthening Security: Some advanced functions of detection and observation in traffic management systems would allow it to ability to detect and react to a specific threat, such as an accident, intersections between pedestrians, or vehicles that violate traffic laws. Such will increase overall traffic safety and prevent accidents by informing and intervening.

Safety guarantees: By means of its present intelligent features for detection and observation, it makes it possible for a system to see and react to hazards relative to the road, for example, accidents, pedestrian crossings, or even vehicles that do not comply with the rules of the road. In such a way, traffic safety overall improves and accidents are avoided through the warnings and interventions. These advanced detection and observation functions allow the system to detect and respond to specific hazards such as accidents, pedestrian crossings, and vehicles disobeying traffic regulations, thus improving traffic safety in general and preventing accidents by means of alerts and interventions.

Less Environmental Damage: Smart traffic systems can lower mileage and fuel consumption from vehicles by improving traffic conditions and reducing stoppage time at an intersection. It reduces pollution caused by vehicles, contributing to a smaller CO_2 footprint within the city.

Dynamic Flexibility: The smart traffic signal-control systems are not alarming like the other types of traffic control systems that only run by fixed schedules. They change real times signal timings when a prevailing event happens to track emergencies. This kind of dynamism results to a traffic control strategy that is much adaptive and very much responsive to real events within the conditions.

Data-driven Decision Making: The system generates large data from traffic patterns, vehicle movement and environmental conditions "This data should be analyzed in determining trends, predicting expected larger traffic patterns, and making possible optimization of infrastructure investments and transportation guidelines." Hence, taking into account the entire aspect of the system creates a more holistic methodology for planning in urban mobility and transport. In fact, this integrated all aspects of the system promotes a more comprehensive approach to planning urban mobility and transport.

Improved public transport integration: Public transport schemes integrated at intersections and prioritized for buses or trams, with waiting time for traffic lights adjusted. At some point, this could foil cities into a more sustainable scenario for moving with lesser environmental impacts from mobility.

Scalability and Modularity: The intelligent traffic management system is developed on modular and scalable themes so that it becomes economically viable and can be enlarged easily at any city and transport network. In turn, it must also integrate into the present infrastructure and legacy systems, thereby reducing result failure and costs.

Public Engagement and Transparency: By ways of mobile apps or digital signage, the system promotes public engagement of citizens and awareness of traffic conditions with data traffic information, alerts, and updates for commuters. Opening up the management of urban mobility.

Continuous Innovation and Improvement: Because of the development of intelligent traffic systems for active traffic management, the roadways could benefit from continual improvements in the use of technology and new data types. The tactics of traffic controls that respond dynamically to the changing nature of cities may be optimized by machine learning algorithms.

References

[1] http://opencv.org/about.html

[2] https://thingspeak.com/pages/learn_more

[3] https://www.slideshare.net/louiseantonio58/image-processing-based-intelligent- traffic-control- systemmatlab-gui

[4] Khekare, G.S.; Sakhare, A.V., "A smart city framework for intelligent traffic system using VANET," Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 2013

[5] Badura, S.; Lieskovsky, A., "Intelligent Traffic System: Cooperation of MANET and Image Processing," Integrated Intelligent Computing (ICIIC), 2010 First International Conference on, vol., no., pp.119,123, 5-7 Aug. 2010

[6] http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/canny_detector/canny_detector.htm

[7] Salama, A.S.; Saleh, B.K.; Eassa, M.M., "Intelligent cross road traffic management system (ICRTMS)," Computer Technology and Development (ICCTD), 2010 2nd International Conference on, vol., no., pp.27,31, 2-4 Nov. 2010

[8] Haimeng Zhao; Xifeng Zheng; Weiya Liu, "Intelligent Traffic Control System Based on DSP and Nios II," Informatics in Control, Automation and Robotics, 2009. CAR '09. International Asia Conference on, vol., no., pp.90,94, 1-2 Feb. 2009