isrjournals

International Journal of Advanced Research in Computer Science Engineering and Information Technology

Volume: 6, Issue: 3,Special Issue: 3 ,Apr,2025 ,ISSN_NO: 2321-3337

EFFICIENT TRAFFIC CONGESTION CONTROL USING INTELLIGENT AI ALGORITHMS

JANANI L[1], SWATHI K[2], THIRISHA M[3], MALATHI S[4*]

[123]UG Scholar-Dept CSE, GRT Institute of Engineering and Technology, Tiruttani, India.

[4*]Assistant Professor-Dept CSE, GRT Institute of Engineering and Technology, Tiruttani, India.

janani26dec2003@gmail.com, swathikollapuri11@gmail.com, thirisha744@gmail.com,

[4*]**Corresponding Author: malathi.s@grt.edu.in**

Abstract- The change in automobile fleet size and consequently within the quantity of traffic isn't accompanied through boom within the space of the road in all of the time. This cause traffic congestion commonly in all the urban areas. In order to avoid traffic jams, we are in situation to come up with a new solution. In preceding decades many technology were evolved and designed solutions to make road transportation safer. Some amongst these techniques were conventional where as other are incorporated. New designed systems are capable of informing drivers about the traffic situations and feasible hazards of the road way with the help of Artificial Intelligent transportation system. This structure consists of one module Software program module. The device makes use of new technology for actual-time collection, employer and transmission which offer the statistics to estimate the correct site traffic density exploited by using traffic-conscious applications

1. INTRODUCTION

In growing metropolitans across the world,traffic congestion has become an inescapable challenge. The surge in vehicles combined with a lack of efficient traffic management systems has led to widespread inefficiencies. This not only results in the wastage of fuel and time but also escalates the overall transportation costs, posing a significant economic burden on individuals and businesses alike. One of the primary culprits of this issue lies in the outdated traffic control systems that rely on fixed green signal timings. These timings are independent of the actual traffic density on the road Consequently, even if a particular lane has fewer vehicles, it receives the same green signal duration, leading to unnecessary delays for vehicles in other lanes. Furthermore, the absence of prioritization for emergency vehicles exacerbates the problem. In critical situations, every second counts, and any delay in their passage could have life-threatening consequences. To mitigate these challenges, a dynamic traffic control system with adaptive prioritization is imperative. Such a system would revolutionize urban mobility by tailoring traffic flow to real-time conditions. Using artificial intelligence and machine learning algorithms, the system would adjust signal timings based on the actual traffic conditions. This means that lanes with lighter traffic would get shorter green signal times, optimizing overall traffic flow. Equipped with GPS trackers and communication systems, emergency vehicles would be detected as they approach an intersection. The system would preemptively clear their path by extending the green signal or even providing a temporary exclusive lane. The dynamic traffic control system would be seamlessly integrated with other smart city technologies, such as intelligent transportation systems, public transit networks, and traffic forecasting models. Implementing such a dynamic traffic control system would not only alleviate congestion but also lead to substantial economic and environmental benefits. It would reduce fuel consumption, lower emissions, and improve overall quality of life in urban centers. In conclusion, adopting an adaptive and prioritized traffic control system is imperative for the sustainable development of our cities. By harnessing the power of technology, we can create more efficient, safer, and environmentally-friendly urban transportation networks. This is a crucial step towards building cities that are not only vibrant and economically robust but also accessible and livable for all residents

2. PROBLEM OF THE STATEMENT

1. Definition of Traffic Congestion

Traffic congestion refers to a condition on road networks where vehicle movement is slowed or halted due to high traffic volumes. It commonly arises when the number of vehicles exceeds the road's capacity, causing delays and longer travel times.

2. Causes of Traffic Congestion

Excess Vehicle Volume: The primary cause is simply too many vehicles trying to use limited road space at the same time.Inefficient Traffic Signals: Static or poorly timed traffic lights fail to adjust for varying traffic densities, contributing to congestion. Road Blockages: Accidents,

isrjournals

International Journal of Advanced Research in Computer Science Engineering and Information Technology

Volume: 6, Issue: 3,Special Issue: 3 ,Apr,2025 ,ISSN_NO: 2321-3337

construction, or stalled vehicles can obstruct lanes and reduce effective road capacity.Urbanization and Population Growth: Increasing urban populations lead to more vehicles on the road without a matching increase in road infrastructure.

3. Incremental Delay

As congestion increases, the amount of time it takes to travel a certain distance also increases. What should be a 10-minute trip may become 30 minutes or more due to traffic jams. These delays are not just inconvenient but also affect productivity and daily schedules.

4. Increased Vehicle Operating Costs

Fuel Consumption: Stop-and-go traffic and idling engines lead to higher fuel usage. Maintenance Costs: Frequent braking and low-speed driving can increase wear and tear on vehicles.Time Loss: Longer travel times mean less time for work, leisure, or rest, leading to an indirect cost for individuals and businesses.

5. Environmental Impact

Traffic congestion results in higher emissions of greenhouse gases and pollutants:Carbon Dioxide (CO_2): Released from burning more fuel. Nitrogen Oxides (NO_x) and Particulate Matter: Harmful pollutants affecting air quality and human health. Noise Pollution: Constant honking and engine noise contribute to environmental stress.

6. Psychological and Physical Stress

Sitting in traffic for long periods causes stress, frustration, and even aggressive behavior (road rage). This can lead to: Increased risk of accidents, Mental fatigue, Decreased overall well-being.

7. Inefficient Traffic Signal Timing

A key factor in traffic build-up is the use of fixed-time traffic signals that do not adapt to real-time traffic conditions. For example: A road with light traffic may still get a full green signal cycle. Meanwhile, a heavily congested road waits too long, causing queues to grow.This imbalance affects overall traffic flow and increases congestion across intersections.

8. Impact on Emergency Services

Congestion severely affects the response time of critical emergency services: Ambulances: Delayed arrival can be life-threatening for patients. Fire Trucks: Delay in reaching fire scenes can result in greater property damage or loss of life. Police Vehicles: Slower response to crimes or accidents reduces public safety. Without clear and congestion-free routes, these services lose their efficiency and reliability.

9. Urban Planning and Infrastructure Challenges

Many cities are struggling with outdated infrastructure that wasn't built for current traffic volumes. Poor planning leads to: Bottlenecks and narrow roads, Lack of proper public transportation alternatives. Insufficient use of smart traffic management systems.

10. Potential Solutions and Innovations

Smart Traffic Management: Using sensors and AI to dynamically adjust traffic light timings based on real-time data. Dedicated Emergency Lanes: Ensuring clear paths for ambulances and other services. Public Transport Improvement: Encouraging people to shift from personal vehicles to buses, subways, or trains. Carpooling and Ride-Sharing: Reducing the number of vehicles on the road. Road Expansion and Flyovers: Increasing road capacity where possible.

3. MODULES DESCRIPTION

3.1 Image Preprocessing

Image preprocessing plays a pivotal role in the efficiency and accuracy of the proposed system. Before subjecting images to advanced CNN algorithms for analysis, it is imperative to prepare and refine the image data. This preprocessing stage involves a series of operations aimed at enhancing the quality and relevance of the images. These operations may include resizing the images to a standardized resolution, normalization to correct for variations in lighting and contrast, and noise reduction to eliminate unwanted artifacts. Additionally, color correction and image rotation may be applied to ensure consistency in analysis. By performing these preprocessing steps, we not only improve the algorithm's ability to detect emergency vehicles but also ensure that the system operates reliably under various lighting and environmental conditions. In essence, image preprocessing acts as the crucial foundation upon which the system's accuracy and effectiveness are built, enabling it to make informed decisions and prioritize vehicles accurately.

3.2 Image Enhancement

The acquired image in RGB is first converted into gray. Now we want to bring our image in contrast to background so that a proper threshold level may be selected while binary conversion is carried out. This calls for image enhancement techniques. The objective of enhancement is to process an image so that result is more suitable than the original image for the specific application. There are many techniques that may be used to play with the features in an image but may not be used in every case. Listed below are a few fundamental functions used frequently for image enhancement.

3.3 Feature Extraction

In machine learning, pattern recognition, and image processing, feature extraction starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps, and in some cases leading to better human interpretations. Feature extraction is related to dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be redundant, then it can be transformed into a reduced set of features (also named a feature vector). Determining a subset of the initial features is called feature selection. The selected features are expected to contain the relevant information from the input data, so that the desired task can be performed by using this reduced representation instead of the complete initial data.

3.4 Convolutional Neural Network (CNN)

Convolutional Neural Networks specialized for applications in image & video recognition. CNN is mainly used in image analysis tasks like Image recognition, Object detection & Segmentation. Convolutional Layer in a typical neural network each input neuron is connected to the next hidden layer. Only small region of the input layer neurons connect to the neuron hidden layer. Pooling Layer is used to reduce the dimensionality of the feature map. There will be multiple activation & pooling layers inside the hidden layer of the CNN. Fully Connected Layers form the last few layers in the network. The input to the fully connected layer is the output from the final Pooling or Convolutional Layer, which is flattened and then fed into the fully connected layer.

4. PROPOSED SYSTEM

The proposed system represents a groundbreaking approach to modernizing urban traffic management, leveraging cutting-edge CNN algorithms to analyze images uploaded in real-time. The core functionality of the system revolves around the intelligent prioritization of vehicles, primarily focusing on emergency vehicles. When a user uploads images, the CNN algorithm swiftly scans and identifies whether an emergency vehicle is present in the scene. In cases where an emergency vehicle is detected, the system immediately grants it the highest priority, dynamically adjusting traffic signals to provide a clear path for rapid response. In scenarios where no emergency vehicles are identified, the system proceeds to prioritize traffic based on congestion levels. High traffic areas receive the next preference, followed by medium traffic zones. This dynamic approach ensures optimal traffic flow and minimizes congestion, thus significantly improving the efficiency of urban transportation systems. By seamlessly integrating image analysis and traffic management, this proposed system aims to enhance both public safety and the overall urban commuting experience.

Advantages: Swift detection and prioritization of emergency vehicles lead to quicker response times during critical situations, potentially saving lives. The system dynamically adjusts traffic signals based on real-time congestion levels, reducing traffic jams and enhancing overall traffic flow. Minimized traffic congestion reduces the likelihood of accidents and enhances road safety for all road users

5. PROPOSED SYSTEM ALGORITHM

CNN

Why CNN for Image Classification? Image classification involves the extraction of features from the image to observe some patterns in the dataset. Using an ANN for the purpose of image classification would end up being very costly in terms of computation since the trainable parameters become extremely large.

For example, if we have a 50*50 image of a cat, and we want to train our traditional ANN on that image to classify into a dog or a cat the trainable parameters become (50*50) *100 image pixels multiplied by hidden layer+ 100 bias + 2 * 100 output neurons + 2 bias = 2,50,30 Examples of different filters and their effects filter helps to use exploit the spatial locality of a particular image by enforcing a local connectivity pattern between neurons. Convolution basically means a point wise multiplication of two functions to produce the third function. Here one function is our image pixels matric and another is our filters. We slide the filter over the image and get the dot product of two matrices. The resulting matric is called an1 ”activation map” or “feature map”.

Step 1: Choose a Dataset

Choose a dataset of your interest or you can also create your own image dataset for solving your own image

classification problem. An easy place to choose a dataset is on kaggle.com.

The dataset I’m going with can be found here.

This dataset contains 12,500 augmented images of blood cells (JPEG) with accompanying cell type labels (CSV). There are approximately 3,000 images for each of 4 different cell types grouped into 4 different folders (according to cell type).

isrjournals
International Journal of Advanced Research in Computer Science Engineering and Information Technology
Volume: 6, Issue: 3,Special Issue: 3 ,Apr,2025 ,ISSN_NO: 2321-3337

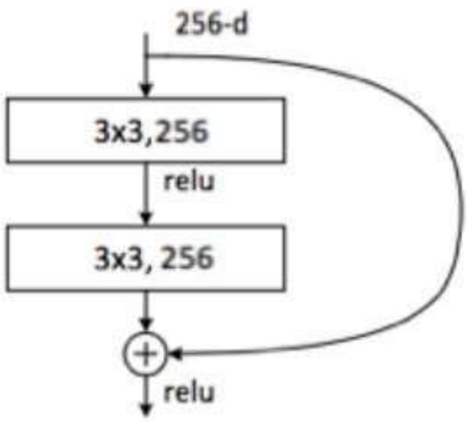
The cell types are Eosinophil, Lymphocyte, Monocyte, and Neutrophil.
Here are all the libraries that we would require and the code for importing them.
Step 2: Prepare Dataset for Training
Preparing our dataset for training will involve assigning paths and creating categories(labels), resizing our images.
Resizing images into 200 X 200
Step 3: Create Training Data
Training is an array that will contain image pixel values and the index at which the image in the CATEGORIES list.
Step 4: Shuffle the Dataset
Step 5: Assigning Labels and Features
This shape of both the lists will be used in Classification using the NEURAL NETWORKS.
Step 6: Normalizing X and converting labels to categorical data
Step 7: Split X and Y for use in CNN
Step 8: Define, compile and train the CNN Model
Step 9: Accuracy and Score of model

```
function XCOMPRESSCU(*pCurCU)
M ←FastCUMope(PO,QP)
if M 4 SPLIT then
C2n ←CHECKINTRA(pCurCU)
else
C2n ←∞
end if
if M != HOMO and Dcur<Dmax then
Cn← 0
for i = 0 to 3 do
pSubCUi← pointer to SubCUi
CN←CN+ XCompressCU(pSubCUi)
end for
else
CN←∞
end if
CHECKBESTMODE(C2N, CN)
end function
```

Residual Networks (ResNet) in Keras
Very deep neural networks are hard to train as they are more prone to vanishing or exploding gradients. To solve this problem, the activation unit from a layer could be fed directly to a deeper layer of the network, which is termed as a **skip connection**.
This forms the basis of **residual networks** or **ResNets**. This post will introduce the basics the residual networks before implementing one in Keras.

Residual block

A building block of a ResNet is called a **residual block** or **identity block**. A residual block is simply when the activation of a layer is fast-forwarded to a deeper layer in the neural network.



As you can see in the image above, the activation from a previous layer is being added to the activation of a deeper layer in the network.

This simple tweak allows training much deeper neural networks.

In theory, the training error should monotonically decrease as more layers are added to a neural network. In practice however, for a traditional neural network, it will reach a point where the training error will start increasing.
ResNets do not suffer from this problem. The training error will keep decreasing as more layers are added to the network. In fact, ResNets have made it possible to train networks with more than 100 layers, even reaching 1000 layers.
Building a ResNet for image classification Now, let's build a ResNet with 50 layers for image classification using Keras.

Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. It was developed with a focus on enabling fast experimentation.
In this case, we will use TensorFlow as the backend. Of course, feel free to grab the entire notebook and make all the necessary imports before starting.
Step 1: Define the identity block
Step 2: Convolution block
Step 3: Build the model
Step 4: Training
Step 5: Print the model summary

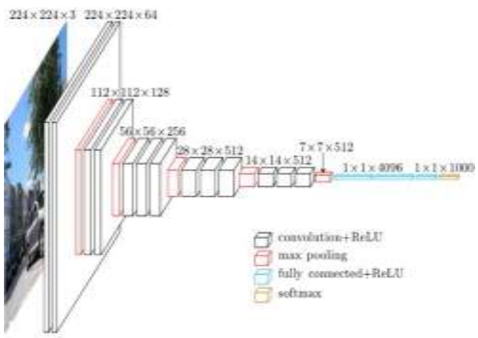
Algorithm 2. Pseudo code of the used preprocessing method.

Input: The raw 1D sensor signal (S) with size of 5625
Output: Graylevel image (Im) with size of 125 x 45

```
1: count = 1;
2: for i=1 to 125 do
3:     for j=1 to 45 do
4:         Im(i,j) = S(count);
5:         count = count + 1;
6:     end for j
7: end for i
8: Normalize Im by using min-max normalization.
```

VGG16 implementation in Keras for beginners

VGG16 is a convolution neural net (CNN) architecture which was used to win ILSVR(Imagenet) competition in 2014. It is considered to be one of the excellent vision model architecture till date. Most unique thing about VGG16 is that instead of having a large number of hyper-parameter they focused on having convolution layers of 3x3 filter with a stride 1 and always used same padding and maxpool layer of 2x2 filter of stride 2. It follows this arrangement of convolution and max pool layers consistently throughout the whole architecture. In the end it has 2 FC(fully connected layers) followed by a softmax for output. The 16 in VGG16 refers to it has 16 layers that have weights. This network is a pretty large network and it has about 138 million (approx) parameters.



Here I first import all the libraries which i will need to implement VGG16. I will be using Sequential method as I am creating a sequential model. Sequential model means that all the layers of the model will be arranged in sequence. Here I have imported Image Data Generator from keras preprocessing. The objective of Image Data Generator is to import data with labels easily into the model. It is a very useful class as it has many function to rescale, rotate, zoom, flip etc. The most useful thing about this class is that it doesn't affect the data stored on the disk. This class alters the data on the go while passing it to the model.

```
function ConvOp(i, IBuf_x)
    C_o ← 0
    while C_o < O do
        if C_o = 0 and i = 0 then
            c ← 0
            ReadBias()
            ReadKernel(C_o ~ C_o + 31, K_c)
        end if
        || PPMACConv(K_c, IBuf_x)
        || PrefetchKernel(C_o + 32 ~ C_o + 63, K_c)
        c ← c̄
    end while
end function

function Conv(T_x, T_y, C_i)
    T_x ← 0, T_y ← 0, C_i ← 0
    while T_y < Y do
        while T_x < X do
            while C_i < I do
                if T_x = 0 and T_y = 0 then
                    c ← 0
                    ReadTile(IBuf_c, T_x, T_y, C_i)
                end if
                || ConvOp(C_i, IBuf_c)
                || PrefetchTile(IBuf_c, T_x, T_y, C_i + 1)
                c ← c̄
            end while
        end while
    end while
end function
```

6. UML DIAGRAMS:
6.1 DATA FLOW DIAGRAM:

LEVEL 0:

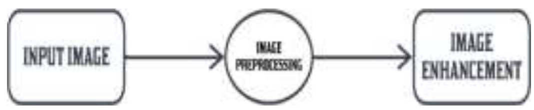


Figure6.1.1 Diagram of Level

LEVEL 1:

Figure6.1.2 Diagram of Level

LEVEL 2:

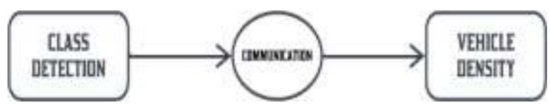


Figure6.1.3 Diagram of Level

isrjournals
International Journal of Advanced Research in Computer Science Engineering and Information Technology
Volume: 6, Issue: 3,Special Issue: 3 ,Apr,2025 ,ISSN_NO: 2321-3337

OVERALL DIAGRAM:

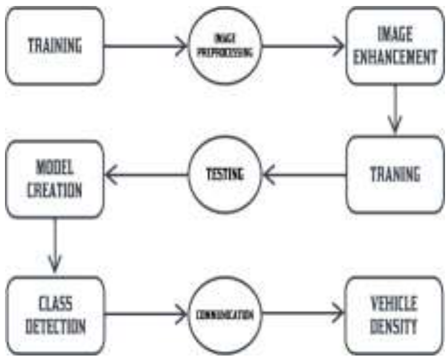


Figure6.1.4 Overall Diagram

7. EXPERIMENTAL RESULTS

This result discusses about the implementation of the efficient traffic congestion control using intelligent ai algorithms where the user upload the images and based upon the images the priority for the vehicle is done through the green signal.

8. CONCLUSION & FUTURE WORK

In conclusion, this project embodies a vision of a smarter,
safer, and more efficient urban transportation system. By leveraging advanced technologies, including Convolutional Neural Networks (CNNs), we have successfully developed a system that can swiftly detect and prioritize emergency vehicles in real-time traffic scenarios, potentially saving lives. Furthermore, the system's ability to assess and manage traffic density contributes to reduced congestion, lower fuel consumption, and improved travel times for commuters. The successful integration of AI-driven image analysis and dynamic traffic management represents a significant milestone in the evolution of modern cities. It not only enhances emergency response capabilities but also makes daily commuting more convenient and sustainable. As we move forward, the impact of this project extends beyond its technical achievements; it offers a glimpse into a future where technology transforms urban mobility, promoting safety, efficiency, and a higher quality of life for all citizens

Future Enhancement: The safety definition inCPA with a quantified method is investigated. Moreover, we will improve the permission model with finer-grained access control for Android, especially, for internet permission. Finally we will strengthen the mathematics depth of the definitions and analysis of CPA.

REFERENCE

[1]. W. Song, H. Liu, and E. Zio, ‘‘Long-range dependence and heavy tail characteristics for remaining useful life prediction in rolling bearing degradation,’’ Appl. Math. Model., vol. 102, pp. 268–284, Feb. 2022..

[2]. Y. Gao, C. Zhou, J. Rong, Y. Wang and S. Liu, "Short-Term Traffic Speed Forecasting Using a Deep Learning Method Based on Multitemporal Traffic Flow Volume," in *IEEE Access*, vol. 10, pp. 82384-82395, 2022, doi: 10.1109/ACCESS.2022.3195353

[3] T. Syum Gebre, L. Beni, E. Tsehaye Wasehun and F. Elikem Dorbu, "AI-Integrated Traffic Information System: A Synergistic Approach of Physics Informed Neural Network and GPT-4 for Traffic Estimation and Real-Time Assistance," in *IEEE Access*, vol. 12, pp. 65869-65882,2024,doi:10.1109/ACCESS.2024.3399094.

[4]. T. Zhang, J. Xu, S. Cong, C. Qu and W. Zhao, "A Hybrid Method of Traffic Congestion Prediction and Control," in *IEEE Access*, vol. 11, pp. 36471-36491, 2023, doi: 10.1109/ACCESS.2023.3266291.

[5]. P. Liu, Z. Xie and T. Li, "UCN-YOLOv5: Traffic Sign Object Detection Algorithm Based on Deep Learning," in *IEEE Access*, vol. 11, pp. 110039-110050, 2023, doi: 10.1109/ACCESS.2023.3322371

[6]. S. Bartoletti, B. M. Masini, V. Martinez, I. Sarris, and A. Bazzi, ‘‘Impact of the generation interval on the performance of sidelink C-V2X autonomous mode,’’ IEEE Access, vol. 9, pp. 35121–35135, 2021.

[7]. X. Meng and C. G. Cassandras, ‘‘Eco-driving of autonomous vehicles for nonstop crossing of signalized intersections,’’ IEEE Trans. Autom. Sci. Eng., vol. 19, no. 1, pp. 320–331, Jan. 2022.

[8]. E. Wang, F. H. Memar, S. Korzelius, A. W. Sadek, and C. Qiao, ‘‘A reinforcement learning approach to CAV and intersection control for energy efficiency,’’ in Proc. 5th Int. Conf. Connected Auto. Driving (MetroCAD), Detroit, MI, USA, Apr. 2022, pp. 81–88.

[9]. Y. Junkai, S. Jinju, L. Xunyi, and Y. Kangjian, ‘‘Speed planning and energy optimal control of hybrid electric vehicles based on Internet of Vehicles,’’ IFAC-PapersOnLine, vol. 54, no. 10, pp. 169–175, 2021

[10]. Z. Bai, P. Hao, W. ShangGuan, B. Cai, and M. J. Barth, ‘‘Hybrid reinforcement learning-based eco-driving strategy for connected and automated vehicles at signalized intersections,’’ IEEE Trans. Intell. Transp. Syst., vol. 23, no. 9, pp. 15850–15863, Sep. 2022.