

# Reduce Traffic Congestion Using Automated Traffic Navigation System

Shreyas Girish Chitnis<sup>1\*</sup>, Anushka Vitthal Kadu<sup>1</sup>, Maithilee Bharat Shinde<sup>1</sup>,  
Swaraj Dilip Zavare<sup>1</sup>, Sharayu Patil<sup>2</sup>

## Abstract

*The concept of activity routing without reliance on traditional activity signals is gaining significant traction. This innovative approach aims to enhance urban vehicular movement by eliminating the need for conventional traffic signal systems. This summary delves into the intricate realm of signal-free activity routing, incorporating advanced technologies such as artificial intelligence, machine learning, and sensor networks. The benefits, including reduced congestion, increased fuel efficiency, and improved traffic flow, are thoroughly discussed. Furthermore, the study identifies challenges such as ensuring safety, managing intersections, and coordinating vehicle movements. The importance of additional research and development in this field is emphasized, as it holds the potential to bring about a paradigm shift in urban mobility. The integration of these advanced technologies can lead to more efficient and intelligent traffic management systems, capable of adapting to real-time conditions and reducing the overall environmental impact. This exploration underscores the transformative potential of signal-free activity routing, advocating for continued innovation and collaboration to address the existing obstacles and fully realize its benefits for urban transportation systems.*

**Keywords:** Machine learning, intelligent traffic management system, traffic navigation, signal systems, intersection, vehicle movements

## INTRODUCTION

Advanced urban transportation faces challenges such as activity clog and wasteful signal-controlled convergences. In reaction, an unused approach to activity route without conventional signals has developed. This approach utilizes progressed innovations like AI and machine learning to optimize activity stream, decreasing clog and moving forward by and large portability [1]. Whereas promising, this move raises questions almost crossing point administration, security, and coordination among vehicles. By investigating these viewpoints, we can open the potential of signal-free route for more viable urban transportation.

### \*Author for Correspondence

Shreyas Girish Chitnis  
E-mail: shreyaschitnis62@gmail.com

<sup>1</sup>Student, Department of Computer Engineering, Vishwaniketan's Institute of Management Entrepreneurship and Engineering Technology (ViMEET), Khalapur, Maharashtra, India

<sup>2</sup>Assistant Professor, Department of Computer Engineering, Vishwaniketan's Institute of Management Entrepreneurship and Engineering Technology (ViMEET), Khalapur, Maharashtra, India

Received Date: April 06, 2024  
Accepted Date: April 30, 2024  
Published Date: July 09, 2024

**Citation:** Shreyas Girish Chitnis, Anushka Vitthal Kadu, Maithilee Bharat Shinde, Swaraj Dilip Zavare, Sharayu Patil. Reduce Traffic Congestion Using Automated Traffic Navigation System. Journal of Computer Technology & Applications. 2024; 15(2): 34–40p.

## PROBLEM STATEMENT

Conventional activity flag frameworks regularly contribute to blockage, wasteful vehicular development, and expanded fuel utilization in urban zones. The challenge lies in finding elective approaches to activity route that dispose of the disadvantages of signal-controlled convergences whereas guaranteeing secure and consistent activity stream. Creating a framework that coordinates rising advances, such as counterfeit insights, machine learning, and sensor systems, to empower signal-free activity route presents a compelling opportunity to revolutionize urban portability. Be

that as it may, this endeavor requires tending to concerns related to crossing point administration, coordination of vehicle developments, and keeping up security guidelines, while striving to achieve optimal operational efficiency and improving overall urban transportation. In urban and rural zones where conventional activity signals are missing, untrustworthy, or deficient to oversee activity, there is a basic requirement for a compelling, real-time activity route arrangement. The nonattendance of signals, coupled with variables like expanding urbanization, populace development, and complex street systems, has driven to activity blockage, security concerns, and wasteful activity administration. To address these challenges, a comprehensive activity route framework is required, one that leverages innovation, information examination, and communication to give clients with dependable, real-time activity data and directing direction, all whereas encouraging successful activity administration by specialists. The framework ought to take into account energetic activity conditions, variable street capacity, and prioritize security and productivity, hence upgrading in general versatility and diminishing traffic-related occurrences.

## RELATED WORK

Numerous studies have explored innovative approaches to traffic navigation without traditional signal systems, focusing on enhancing urban mobility and addressing traffic congestion challenges. Some noteworthy contributions include:

1. *Decentralized Traffic Management Systems*: Research by Li *et al.* introduced a decentralized traffic management system that utilizes vehicle-to-vehicle communication to enable cooperative collision avoidance and intersection negotiation [2]. This approach reduces the need for centralized signal control and enhances traffic flow.
2. *Reinforcement Learning for Intersection Control*: Zhang *et al.* proposed a reinforcement learning-based approach for optimizing traffic signal timings at intersections [3]. Their work demonstrated the potential of using AI algorithms to dynamically adjust signal timings and adapt to changing traffic conditions.
3. *Connected Vehicle Networks*: Smith and Razo explored the use of connected vehicle networks and sensor data to enable vehicles to communicate with each other and adapt to traffic conditions in real-time [4]. This approach has the potential to reduce congestion and improve traffic efficiency.
4. *Swarm Intelligence-Based Navigation*: A study by Wang *et al.* introduced a swarm intelligence-based approach where vehicles navigate through intersections in a self-organized manner, resembling the behavior of natural swarms [5]. This approach reduces the need for fixed signal systems and promotes adaptive traffic movement.
5. *Cellular Automata Models*: Helbing and Schreckenberg proposed cellular automata models to simulate traffic flow and explore traffic dynamics without signal control [6]. These models have been foundational in understanding emergent traffic patterns.
6. *Intersection-Free Routing Algorithms*: Shah and Gupta developed routing algorithms that guide vehicles through paths that avoid intersections, minimizing the need for signal-controlled junctions [7]. This approach aims to optimize traffic flow through preplanned routes.
7. *Simulation and Testing*: Research involves creating realistic traffic simulation environments to test and evaluate navigation systems, traffic management strategies, and routing algorithms [8]. These simulations help refine navigation technologies before deployment in the real world.
8. *Traffic Data Visualization*: Researchers are developing visualization tools and dashboards that provide users and traffic management authorities with intuitive and informative views of traffic conditions [9]. These tools help users make informed decisions and authorities manage traffic more effectively.

## METHODOLOGY

1. *Data Collection and Analysis*: Utilize sensor networks, GPS data, and connected vehicle networks to gather real-time information on traffic flow, density, and vehicle positions. Analyze historical traffic data to identify patterns, peak hours, and congestion-prone areas.

2. *Artificial Intelligence and Machine Learning*: Develop AI algorithms that can predict traffic conditions and congestion based on the collected data. Implement machine learning models to adaptively adjust traffic routing and prioritize alternative routes to avoid congestion.
3. *Decentralized Traffic Management*: Enable vehicles to communicate with each other using V2V (vehicle-to-vehicle) and V2I (vehicle-to-infrastructure) communication. Implement negotiation protocols that allow vehicles to coordinate their movements at intersections without centralized signal control.
4. *Reinforcement Learning for Intersection Control*: Utilize reinforcement learning algorithms to optimize intersection management in real-time. Train models to dynamically adjust the timing of vehicle movements at intersections based on current traffic conditions.
5. *Swarm Intelligence-Inspired Navigation*: Design navigation algorithms inspired by swarm intelligence, where vehicles collectively adjust their speeds and trajectories to maintain smooth traffic flow. Implement self-organizing behaviors that enable vehicles to navigate through intersections without the need for fixed signals.
6. *Testing and Simulation*: Create simulation environments to test the proposed methodologies in controlled settings. Use historical data and realistic traffic scenarios to validate the effectiveness of signal-free navigation approaches.
7. *Integration with Urban Infrastructure*: Collaborate with urban planners and transportation authorities to integrate the developed methodologies into existing urban infrastructure. Implement the proposed approaches in selected areas to observe their impact on traffic flow and congestion reduction.
8. *Continuous Optimization and Learning*: Continuously update AI models and algorithms based on real-world performance and feedback. Incorporate adaptive learning mechanisms to improve the accuracy of traffic predictions and navigation decisions over time.

### Libraries

1. *Data Analysis and Processing*: NumPy-pandas-Matplotlib-Seaborn.
2. *Artificial Intelligence and Machine Learning*: Tensor-Flow or PyTorch-scikit-learn-XGBoost or LightGBM.
3. *Decentralized Communication*: Depending on the specific communication protocols you are using (V2V, V2I), you might need to work with standards like DSRC or C-V2X. Implementing these protocols could involve custom code rather than pre-existing libraries.
4. *Reinforcement Learning*: OpenAI Gym-Stable Baselines.
5. *Simulation*: SUMO (Simulation of Urban MObility)-AnyLogic.
6. *Swarm Intelligence*: Since this concept is more algorithmic in nature, you might not need specific libraries. You could implement the basic Boids Algorithm using standard programming tools.

### Installation Process

1. *NumPy, pandas, Matplotlib, Seaborn*: You can install these libraries using the following commands in your terminal or command prompt:  
`pip install numpy pandas matplotlib seaborn`
2. *TensorFlow*: For TensorFlow installation, use: `pip install tensorflow` For GPU support (if you have an NVIDIA GPU and want to utilize it for faster training): `pip install tensorflow-gpu`
3. *PyTorch*: PyTorch installation commands depend on your CUDA version and operating system. For a basic CPU-only installation, use: `pip install torch torchvision`
4. *Scikit-learn*: Install scikit-learn with: `pip install scikit-learn`
5. *XGBoost and LightGBM*: You can install these gradient boosting libraries with: `pip install xgboost lightgbm`
6. *OpenAI Gym and Stable Baselines*: For OpenAI Gym and Stable Baselines, you can use: `pip install gym stable-baselines3`

## RESULT

### Car Detection

Libraries like TensorFlow or PyTorch can be utilized for this. Using Haar Cascades (illustration for ambulances): Import OpenCV (cv2) library. Load a pre-trained Haar Cascade classifier particular to ambulances. Read the video file. Convert each outline to grayscale for way better detection [10]. Detect ambulances utilizing cv2.CascadeClassifier.detectMultiScale. Draw bounding boxes around recognized ambulances in each frame. Display the coming about the video with crisis vehicles highlighted. Limitations and Improvements: This approach might not work well for other crisis vehicles (fire trucks, police cars). Accuracy can be made strides through: Pre-processing outlines (commotion lessening, color correction). Adjusting location parameters (scale figure, least neighbors) (Figure 1).

There are two primary approaches: (1) Pre-trained Classifiers (Haar Cascades): Faster and less difficult, perfect for beginners; leverages pre-trained models to recognize car features; and accuracy might be lower in complex scenarios (lighting varieties, car types). (2) Deep Learning: More precise but requires more exertion and resources. Train a demonstration on a tremendous dataset of labeled car images/videos. Libraries like TensorFlow or PyTorch can be utilized for this.

Using Haar Cascades: Import OpenCV (cv2) and NumPy libraries. Load the pre-trained car classifier. Read the video/image. Convert the outline to grayscale (pre-processing). Detect cars utilizing cv2.CascadeClassifier.detectMultiScale. Draw bounding boxes around recognized cars. Display the coming-about frame. For superior accuracy: Pre-process the picture (clamor decrease, color correction). Adjust discovery parameters (scale figure, least neighbors).

### Emergency Vehicle Detection

Identify crisis vehicles in recordings with OpenCV, advertising two primary approaches: (1) Pre-trained Haar Cascade Classifiers: It is simpler and quicker, but with limitations. Uses pre-trained models (like car classifiers) to discover particular features. It might battle with non-standard crisis vehicles or complex lighting. (2) Deep Learning: It is more exact but requires critical assets and expertise. Train a profound learning show on a huge dataset of labeled videos/images of crisis vehicles (Figure 2).

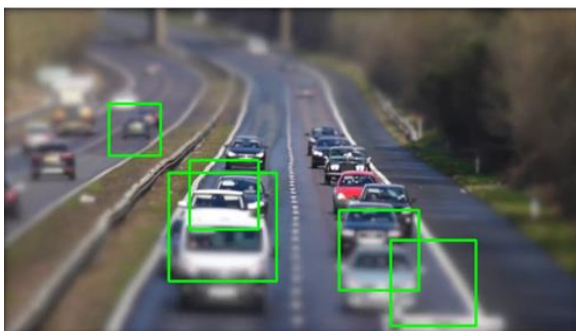


Figure 1. Car Detection.



Figure 2. Emergency Vehicle Detection.

### Distance Detection

Python can estimate distance between vehicles in videos, but it is not a perfect science. Here is the gist:

1. *Vehicle detection*: First, you need to find the vehicles. This can be done with: Pre-trained Classifiers (Haar Cascades): Easier but less accurate, especially for distance. Relies on car features, not size for distance estimation. Deep Learning: More accurate but requires training on data with distance labels (complex).
2. *Distance estimation*: Once vehicles are spotted, estimate the gap: Monocular Vision with Calibration: Uses camera properties and known car size to estimate distance based on vehicle image size. Requires camera calibration for accuracy. Stereo Vision (2 Cameras): Like human eyes, it uses two cameras for depth perception and calculates distance directly. Requires more hardware. Important Note: Cameras have limitations. Lighting and angle variations affect accuracy. Deep learning with distance data can help. Beyond Python: LiDAR sensors directly measure distance with high precision. While not done in Python alone, LiDAR data can be integrated for further analysis.
3. *Libraries*: OpenCV (vehicle detection, calibration), TensorFlow/PyTorch (deep learning with distance labels).
4. *The bottom line*: Python offers tools, but deep learning with Li-DAR is the most promising approach for real-world accuracy (Figure 3).

### Figma Map Integration

Figma, a popular design tool, does not inherently support direct map integration within its platform (Figure 4). However, you can design map interfaces within Figma using static images or by creating custom map components using shapes and icons. Here is how:

1. *Static Map Images*: You can use screenshots or exported map images from map services like Google Maps or OpenStreetMap. Import these images into Figma and design around them, overlaying markers, routes, or annotations as needed.
2. *Custom Map Components*: Create map components using Figma's shape and vector tools. Design a base map with shapes representing landmasses, roads, and water bodies. Add markers, pins, or icons to denote specific locations. Utilize Figma's layering and grouping features to organize and customize the map elements.

```
PS S:\swaraj\swaraj workspace> python  
Distance between vehicles: 15
```

**Figure 3.** Distance detection.



**Figure 4.** Figma map integration.

3. *Prototyping*: While Figma does not support live map interactions, you can simulate map interactions using Figma's prototyping capabilities. Link map elements to different frames to simulate navigation or interaction flows within your map interface.
4. *Export and Integration*: Once your map design is complete, export your designs as images or SVGs. You can then integrate these designs into your web or mobile applications alongside actual map APIs like Google Maps or Leaflet to achieve live map functionality.

While Figma does not directly integrate with map APIs, you can design map interfaces within Figma and then implement them using code in your actual projects.

## CONCLUSION

In a world where automated traffic navigation systems guide us through the labyrinth of roads, offering not just directions but a digital co-pilot that ensures efficiency, safety, and convenience, the conclusion is clear: we stand at the crossroads of a transportation revolution. These systems are not just about getting from point A to point B; they are the architects of a new era in mobility, reshaping how we move, interact, and experience the world. Yet, with their power comes a responsibility to navigate the ethical, regulatory, and social landscapes that lie ahead. As we yield to the guidance of our digital navigators, let us ensure that they guide us towards a future that is safer, more sustainable, and more connected than ever before.

1. *Improved Traffic Efficiency*: These systems have the potential to significantly improve traffic flow and reduce congestion. They can suggest optimal routes, taking into account real-time traffic data, accidents, and road closures, thereby reducing travel times and fuel consumption.
2. *Enhanced Safety*: Automated traffic navigation systems can contribute to safer roads. They provide warnings about accidents, road hazards, and traffic conditions ahead, allowing drivers to make informed decisions. Furthermore, V2V and V2I communication can help vehicles avoid collisions and improve overall road safety.
3. *Economic Benefits*: Enhanced traffic flow and safety can lead to economic benefits. Businesses can optimize their delivery routes, and cities can reduce the costs associated with traffic management and accident response.
4. *Challenges and Concerns*: While automated traffic navigation systems have many benefits, there are also challenges and concerns. These include data privacy issues, potential over-reliance on technology, cybersecurity risks, and the need for constant updates and maintenance to keep the systems accurate and up to date.
5. *Integration and Standardization*: The successful implementation of automated traffic navigation systems requires collaboration and standardization among vehicle manufacturers, technology companies, and government bodies. This can be a complex process, and achieving uniformity is vital for their effectiveness.
6. *Legal and Regulatory Frameworks*: Governments worldwide are still working on developing comprehensive legal and regulatory frameworks to govern the use of automated traffic navigation systems. Issues related to liability, insurance, and accountability need to be addressed.
7. *Technological Advancements*: As technology continues to advance, we can expect even more sophisticated automated traffic navigation systems. This may include full automation of vehicles, increased accuracy in real-time data, and further integration with smart infrastructure.

In conclusion, automated traffic navigation systems have the potential to transform transportation by enhancing traffic efficiency, safety, and environmental sustainability. However, their widespread adoption also comes with challenges related to privacy, regulation, and labor implications. As technology evolves and the infrastructure adapts, these systems will continue to play a significant role in shaping the future of transportation.

## REFERENCES

1. Dikshit S, Atiq A, Shahid M, Dwivedi V, Thusu A. The use of artificial intelligence to optimize the routing of vehicles and reduce traffic congestion in urban areas. EAI Endorsed Trans Energy Web. 2023 Dec 15; 10: 1–13.

2. He W, Li H, Zhi X, Li X, Zhang J, Hou Q, Li Y. Overview of V2V and V2I wireless communication for cooperative vehicle infrastructure systems. In 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). 2019 Dec 20; 127–134.
3. Li Z, Yu H, Zhang G, Dong S, Xu CZ. Network-wide traffic signal control optimization using a multi-agent deep reinforcement learning. *Transp Res Part C: Emerg Technol.* 2021 Apr 1; 125: 103059.
4. Smith S, Razo M. Using traffic microsimulation to assess deployment strategies for the connected vehicle safety pilot. *J Intell Transp Syst.* 2016 Jan 2; 20(1): 66–74.
5. Wang N, Wang X, Palacharla P, Ikeuchi T. Cooperative autonomous driving for traffic congestion avoidance through vehicle-to-vehicle communications. In 2017 IEEE Vehicular Networking Conference (VNC). 2017 Nov 27; 327–330.
6. Helbing D, Schreckenberg M. Cellular automata simulating experimental properties of traffic flow. *Phys Rev E.* 1999 Mar 1; 59(3): R2505.
7. Shah PM, Gupta N. Analysis of speed parameters of mixed traffic flow on the sections of arterial streets (Jalandhar and Chandigarh Cities). *Indian J Sci Technol.* 2016 Dec; 9(47): 1–6.
8. Chao Q, Bi H, Li W, Mao T, Wang Z, Lin MC, Deng Z. A survey on visual traffic simulation: Models, evaluations, and applications in autonomous driving. *Comput Graph Forum.* 2020 Feb; 39(1): 287–308.
9. Tsouros I, Polydoropoulou A, Tsirimpa A, Karakikes I, Tahmasseby S, Mohammad AA, Alhajyaseen WK. From Raw Data to Informed Decisions: The Development of an Online Data Repository and Visualization Dashboard for Transportation Data. *Procedia Comput Sci.* 2024 Jan 1; 231: 161–7.
10. Tumiatti A. (2019 Jul 26). Object detection with non-traditional sensors. [Online]. Webthesis. Politoit. Available from: <https://webthesis.biblio.polito.it/11704/>