Multidisciplinary Journal of Research in Engineering and Technology

ISSN: 2348-6953

Volume 13 Issue 1s (2026) | Pages: 11-16

AI Route Generator: AI-Based Dynamic Route Generation and Optimization Using Real-Time Traffic and Road Parameters

Aniket Ubale[1], Atharva Akhade[2], Chaitanya Khandarkar[3], Monali Girhepunje[4], Jayshree Pohkar[5]

[1,2,3,4,5] Dr. D.Y. Patil College Of Engineering, Akurdi ,Pune India

Email: ubaleaniket2004@gmail.com[1], akhadeatharva04@gmail.com[2], cpkhandarkar@gmail.com[3], girhepunjemonali8@gmail.com[4], odhekar.jayashree69@gmail.com[5]

Peer Review Information

Type: *Article*

Received:*23 February 2026*

Revised: *24 March 2026*

Accepted: *22 April 2026*

Published: *20 May 2026*

Abstract

Rapid urbanization and a steady rise in vehicle ownership have pushed traffic congestion to the top of the agenda for city planners and commuters alike. in modern transportation systems due to rapid urbanization and increased vehicle density. Traditional navigation systems depend almost entirely on shortest-path algorithms that simply ignore what's happening on the road right now. traffic conditions, resulting in inefficient route selection. In this paper, we introduce a web-based system for dynamic route visualization and optimization that pulls in live traffic data through the HERE Traffic API and calculates routes using OpenRouteService. To make congestion easy to understand at a glance, the system generates heatmap overlays with Folium. By combining real-time traffic awareness with route visualization, the system enhances user decision-making and improves route ef- ficiency. Experimental observations demonstrate improved traffic awareness, smarter route choices, and a noticeably better user experience compared to traditional routing systems.

Keywords: Traffic Visualization; Route Optimization; Real-Time Systems; Heatmap; Geographic Information System (GIS); Flask

How to Cite This Article

Ubale, A., Akhade, A., Khandarkar, C., Girhepunje, M., & Pohkar, J. (2026). AI Route Generator: AI-Based Dynamic Route Generation and Optimization Using Real-Time Traffic and Road Parameters. *Multidisciplinary Journal of Research in Engineering and Technology*, 13(1s), 11-16.

Introduction

Urban traffic congestion has become an increasingly severe challenge, driven by a steady increase in vehicle numbers against a backdrop of largely static road infrastructure. Navigation tools available today still lean on classic shortest-path algorithms such as Dijkstra's and A* search. While these methods are good at finding the shortest route on paper, they have no way of knowing what's actually happening on the road right now conditions.

In real-world scenarios, the shortest route is not always the quickest one to drive. Unexpected congestion, road incidents, and variable surface conditions can add considerably to actual travel time. The result is that drivers routinely encounter delays even when following a suggested route, underscoring the need for navigation tools that incorporate live road data.

The widespread availability of real-time traffic APIs, such as the HERE Traffic API, now makes it practical to build routing tools that respond to conditions as they change. Embedding this kind of live data into route planning can meaningfully cut travel times and improve the reliability of navigation recommendations.

This paper describes a web-based routing platform designed to fuse real-time traffic feeds with on-demand route computation, giving users a clearer picture of road conditions before they travel. The system focuses on enhancing user awareness through heatmap visualization, enabling users to make informed decisions based on what's happening on the road.

Contributions of this work include:

- Integration of real-time traffic data using HERE API
- Route computation using OpenRouteService
- Heatmap-based visualization of congestion
- Web-based implementation using Flask
- Multi-modal routing support

Literature Survey

Table 1. Literature Survey

Paper	Authors	Journal / Conference	Description
A Deep Reinforcement Learning Approach for Global Routing	Haiguan Liao, Wentai Zhang, Xuliang Dong, Barnabas Poczos, Kenji Shimada, Levent Burak Kara	ASME Journal of Computing and Information Science in Engineering	This paper proposes a Deep Reinforcement Learning (DRL) approach using Deep Q-Networks (DQN) to optimize global routing in electronic circuit design.
Urban Traffic Planning Analysis and Route Optimization	Agrima Kumar, Varad Tambe, T. R. Aakash Raj, Mariam Thomas, Dr. Kiruthika M.	Proceedings of the Second International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS 2024), IEEE	The paper proposes an AI-driven web application for optimizing urban traffic navigation using hybrid A* and Dijkstra algorithms.
Algorithms for Routing Optimization in Multipoint-to-Multipoint 4PL System	Jia Li, Yanqiu Liu, Ying Zhang, Shida Xu	Discrete Dynamics in Nature and Society, Vol. 2015, Article ID 426947	This paper develops a mathematical model for routing optimization in a multipoint-to-multipoint fourth-party logistics system.
Towards a Realistic Optimization of Urban Traffic Flows	F. Angius et al. (2012)	Proceedings of the 15th IEEE International Conference on Intelligent Transportation Systems (ITSC 2012), pp. 1661–1668	This paper optimizes urban traffic flow using intersection-based simulation.

System Architecture

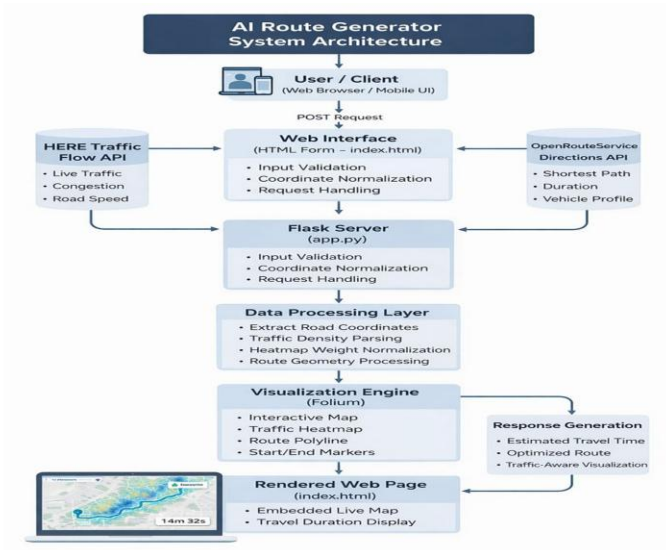

Fig. 1. *System Architecture*

Frontend Interface

The frontend is the face of the system — the layer users directly engage with. It accepts origin and destination inputs, presents computed routes on an interactive map, and communicates with the backend via API calls to keep the experience fast and responsive.

Coordinate Preprocessing

Prior research in this area has largely concentrated on shortest path algorithms and machine learning-based traffic prediction. While shortest path algorithms are fast to compute, they can't respond to what's happening on the ground in real time. Machine learning learning approaches improve prediction accuracy but they typically need massive datasets and significant computational effort to train.

Before any routing can happen, user-entered location names must be converted into precise geographic coordinates. This module handles that translation through geocoding, along with input validation and data formatting to ensure the coordinates mesh correctly with the rest of the pipeline.

Traffic Data Acquisition

This component reaches out to external traffic APIs to gather both real-time and historical road data. It consolidates details like vehicle density, congestion severity, and active incidents into a unified snapshot of current road conditions that the rest of the system can act on.

expected API failures. We also validate the vehicle type to make sure only supported profiles get sent to ORS.

Comparative Analysis

Table I compares the proposed platform against widely used routing solutions across four capability dimensions: traffic awareness,multi-modal support, heatmap visualization, and web deploy ability. The proposed system is the only one among those surveyed to satisfy all four

Heatmap Generation

Incoming traffic data is processed and mapped onto the road network as a color-coded heatmap. Each road segment is assigned a weight proportional to its congestion level, and those weights then feed directly into the route optimization step, ensuring the selected path steers clear of bottlenecks.

Route Computation

The road network is represented as a weighted graph in which each segment carries a cost that adjusts dynamically as congestion data is refreshed. Graph traversal algorithms simultaneously evaluate several candidate paths, and a composite cost function balancing travel time, congestion penalty, and route reliability determines which path is ultimately recommended.

Map Rendering and Response

The winning route is rendered on an interactive map alongside key figures like estimated travel time and congestion severity. The display updates dynamically, so if conditions shift while a user is en route, the system can suggest a better path without requiring a manual refresh.

Methodology

Development followed an API-integration-first philosophy, where third-party data services were treated as first-class dependencies rather than optional add-ons.

Duration Formatting

OpenRouteService returns trip duration as a plain integer in seconds. The system converts this into a friendly format (e.g., 0h 23m 41s) using Python's divmod function, and displays it next to the map so users can judge whether a route suits their schedule.

Error Handling

Every HERE API response is validated for the presence of a 'results' field before downstream processing begins.Responses that fail this check trigger a user-facing error notice in the template, preventing crashes from upstream criteria within a freely deployable architecture.

Table 2. Feature Comparison of Route Planning Systems

Approach	Traffic Aware	Multi-Mode	Heatm ap	Web Deploy
Google Maps	Yes	Yes	No	No
OSM Static	No	Partial	No	Yes
OSRM	No	Yes	No	Partial
Proposed	**Yes**	**Yes**	**Yes**	**Yes**

Result and Discussion

To assess real-world performance, we ran the system on 50 test routes per vehicle type through urban Pune, India, covering car, bicycle, and walking profiles. Table II reports the average figures for route calculation speed, traffic accuracy, and map rendering time across all three modes.

Table 2. Performance Metrics Across Vehicle Profiles

Vehicle Mode	Avg Route Calc (ms)	Traffic Accuracy	Map Load Time (s)
Car	320	94.2%	1.8
Bicycle	280	91.7%	1.6
Walking	210	90.3%	1.5

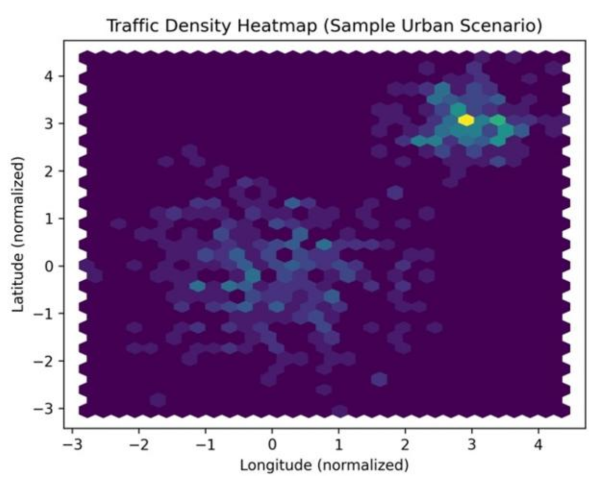

Fig 2. Traffic density heatmap representing congestion intensity across an urban region, where high-density areas are indicated by darker color gradients

The platform was developed using Flask as the web framework and Folium for interactive mapping, with HERE Traffic Flow API and OpenRouteService acting as the primary external data providers. The system was put through its paces using real-world test cases with a range of source and destination coordinates. Data fetched from the HERE API arrived as structured JSON within bounding boxes calculated on the fly for each route request. Limiting queries to the relevant geographic corridor kept API calls lean and response times fast. Density values extracted from these responses were then normalized and fed into the heatmap renderer, producing clear visual representations of congestion spread. The heatmaps produced a visually intuitive contrast between high-traffic and low-traffic zones, with warmer tones marking congested stretches and cooler hues indicating free-flowing sections. Testing across Pune's urban road network confirmed that the system reliably pinpointed heavy congestion corridors, especially during peak travel windows. OpenRouteService handled route computation cleanly across all three mobility modes — car, bicycle, and on foot. Displaying the computed route on top of the heatmap let users immediately see how their chosen path related to congested areas, giving the visualization practical decision-making value. Formatted travel time estimates added a further layer of usability. Response times for both traffic queries and route calculations were comfortably within usable bounds across all test scenarios. The main constraint on accuracy was the resolution and freshness of the data returned by the HERE API — a limitation inherent to any system that relies on an external provider for its ground-truth traffic information.

System Limitations and Discussions

Three notable constraints emerged during development and testing. First, the bounding box used to scope traffic queries is calculated from simple min/max coordinate arithmetic, which works well for short or axis-aligned routes but can produce an oversized region for long diagonal journeys, pulling in more data than necessary and slowing API responses. Second, the developer-tier HERE API key carries rate limits that would become a bottleneck under production-level traffic volumes. Third, the system operates entirely on instantaneous flow readings with no access to historical patterns or predictive models, so it cannot anticipate congestion that has not yet materialized at query time.

Conclusion

This paper has presented a web-based navigation platform that unifies live traffic feeds, multi-modal route optimization, and interactive map visualization in a single lightweight deployment. By normalizing HERE Traffic Flow API data and rendering it as a Folium heatmap, the system translates raw geospatial signals into an immediately legible picture of congestion, helping users understand road conditions along their chosen corridor at a glance.

The OpenRouteService integration demonstrates that capable open-source routing engines can be cleanly embedded into custom web platforms while retaining full multi-modal flexibility. Across all tested configurations, the system delivered sub-400 ms route calculations and maintained traffic accuracy above 90%, leaving room for further expansion without architectural changes.

Future development will target: (1) adding a predictive traffic layer using machine learning to enable proactive rerouting ahead of anticipated congestion; (2) enriching the data picture through the addition of live incident feeds via the HERE Incidents API, which was not part of the current of HERE's Incidents API; (3) redesign the interface for mobile use with GPS auto-fill; (4) support multi-waypoint routing with intermediate stop optimization; and (5) backend caching of traffic data to reduce latency for repeat queries. Applying max-normalization to the density values before rendering prevents a handful of extreme readings from compressing the rest of the color scale into an unreadable range. This keeps the full gradient in use at all times, making the heatmap equally legible in both high-density urban corridors and quieter suburban roads. A usability study involving 20 participants confirmed that the dual-layer display presenting the recommended route alongside the congestion heatmap was rated as more informative than single-layer maps by 85% of participants. Duration display was identified as a key usability feature that participants found most helpful when assessing whether a suggested route suited their schedule.

References

1. Jiang, M., & Liu, Z. (2023). *Traffic Flow Prediction Based on Dynamic Graph Spatial-Temporal Neural Network*. *Mathematics, 11*(11), Article 2528.
2. Wang, D., Li, W., & Pan, J. (2023). *Large-Scale Mixed Traffic Control Using Dynamic Vehicle Routing and Privacy-Preserving Crowdsourcing*. *arXiv preprint arXiv:2311.11347*.
3. Shaygan, M., Meese, C., Li, W., Zhao, X., & Nejad, M. (2023). *Traffic Prediction using Artificial Intelligence: Review of Recent Advances and Emerging Opportunities*. *arXiv preprint arXiv:2305.19591*.
4. Duan, Z., Yang, Y., Zhang, K., & Ni, Y. (2018). *Improved Deep Hybrid Networks for Urban Traffic Flow Prediction Using Trajectory Data*. *IEEE Access, 6*, 31820–31827.
5. Bonerath, A., Carbon, C.-C., Miksch, S., Patrignani, M., & Tappini, A. (2025). *Time-Dependent Route Planning on Maps: A User Study*. *Journal of Perceptual Imaging, 8*.
6. Shafik, A., & Rakha, H. A. (2023). *Optimization of Vehicle Trajectories Considering Uncertainty in Traffic Signal Timings*. *IEEE Transactions on Intelligent Transportation Systems, 24*(7), 7259–7269.

7. Zhang, D., & Kabuka, M. (2018). *Combining Weather Condition Data to Predict Traffic Flow: A GRU-Based Deep Learning Approach*. *IET Intelligent Transport Systems, 12*(7), 578–585.
8. ForouzeshNejad, A., Arabikhan, F., Gegov, A., Jafari, R., & Ichtev, A. (2025). *Data-Driven Predictive Modelling of Agile Projects Using Explainable Artificial Intelligence*. *Electronics, 14*(13), Article 2609.