Image Processing Based Path Finding Algorithm

Project Submitted By

Chayan Mahata

Aditya Prasad

Shubhasish Mondal

Bristi Ghosh

BACHELOR OF TECHNOLOGY (CSE)

DEPARTMENT OF COMPUTER SCIENCE

Sister Nivedita University, Newtown

Kolkata, West Bengal

2025

Image Processing Based Path Finding Algorithm

Project Submitted in Partial Fulfilment of the Requirements for Award of the Degree of

Bachelor of Technology (CSE)

Student Name	Reg. Number	Email ID
Chayan Mahata	220100205572	chayanmahata2003@gmail.com
Aditya Prasad	220100057893	adityaprasad072004@gmail.com
Shubhasish Mondal	220100478418	shubhasishmondal4@gmail.com
Bristi Ghosh	220100192883	bristighosh.khu@gmail.com

Submission Date: 21-November,2025

Under the supervision of

Basudev Saha

Assistant Professor

Department of Computer Science

Sister Nivedita University, Newtown

Kolkata, West Bengal

A Satyam Roychowdhury initiative

TECHNO INDIA GROUP | SNU SISTER NIVEDITA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE

Sister Nivedita University, Newtown

Kolkata, West Bengal

November,2025

Declaration

We hereby declare that this dissertation is the product of our own work, and we attest that it contains no material that resulted from collaboration, except where explicitly acknowledged in the text. Furthermore, we confirm that this project has not been previously submitted, either in part or in its entirety, to any other University or Institution for the purpose of obtaining any degree, diploma, or other qualification. All sources used and referenced in this dissertation are duly credited, and any borrowed ideas or information are appropriately cited in accordance with academic standards and guidelines.

Date: 21-11-2025
Place: SNU, WB

. . . .
Chayan Mahata
Registration Number: 220100205572

. . . .
Aditya Prasad
Registration Number: 220100057893

. . . .
Shubhasish Mondal
Registration Number: 220100478418

. . . .
Bristi Ghosh
Registration Number: 220100192883

Certificate

This is to certify that the project entitled "Image processing based path finding algorithm", submitted by Chayan Mahata, Aditya Prasad, Shubhasish Mondal, Brishti Ghosh to Sister Nivedita University, West Bengal for the award of the degree of Bachelor of Technology (CSE) is a bonafide record of the project work carried out by them under my supervision and guidance. The content of the project, in full or parts have not been submitted to any other institute or university for the award of any degree or diploma.

. . . .

(Basudev Saha)
Assistant Professor
Dept. of Computer Science
Sister Nivedita University

Date: 21-11-2025
Place: SNU, WB

Contents

List of Tables

List of Figures

Acknowledgement

We would like to express our heartfelt gratitude to our esteemed guide, **Basudev Saha**, Assistant Professor at Sister Nivedita University, WB, for his invaluable guidance, unwavering support, and encouragement throughout our project journey. His profound knowledge, insightful suggestions, and mentorship have been instrumental in shaping our project and academic pursuits.

We extend our sincere thanks to the members of the Project Committee at the Department of Computer Science , Sister Nivedita University, for their valuable feedback and constructive criticism during the evaluation of our work. Their expertise and insights have been crucial in refining the quality of our project.

We would like to express our heartfelt gratitude to our HOD, the esteemed Vice-Chancellor, the Class teachers and all the faculty members of our department for their unwavering support and guidance throughout our project journey. Your invaluable contributions have been instrumental in shaping this project endeavor.

We are deeply indebted to our family members whose constant love, encouragement, and support have been the driving force behind our academic pursuits. Their unwavering belief in us has given us the strength to overcome challenges and achieve milestones.

We also want to express our heartfelt gratitude to our friends, colleagues, juniors, and seniors for their camaraderie, encouragement, and support throughout this academic journey. Their presence and interactions have enriched us with diverse perspectives, making this journey all the more rewarding.

Lastly, we extend our sincere appreciation to all those who have been our pillars of support during this project endeavor. Your encouragement, motivation, and belief in our abilities have been invaluable, and we are truly grateful for your presence in our life.

Chayan Mahata
(Registration No.: 220100205572) ....

Aditya Prasad
(Registration No.: 220100057893) ....

Shubhasish Mondal
(Registration No.: 220100478418) ....

Bristi Ghosh
(Registration No.: 220100192883) ....

Abstract

This project focuses on creating a system that can automatically find the best route between two points by analyzing an image. It identifies all important nodes and the edges connecting them using simple color-based processing. These connections are then turned into a graph, and Dijkstra's algorithm is used to find the shortest and alternative paths. The system works without manual input and can be easily extended to real maps, navigation tools, and autonomous technologies.

Chapter 1

Introduction

In our project, pathfinding means determining the shortest and safest route between two points on the graph generated from the Voronoi image. The algorithm first detects nodes and edges from the image, converts them into a graph, and then calculates the optimal path using techniques like Dijkstra. It also identifies alternative routes so that even if one path is blocked or inefficient, other valid paths can still be used. This helps us analyze how movement can be optimized in complex, real-world environments.

1.1 Context of the Study

General domain of our project is Image Processing. .
Image processing is important today because it allows computers to automatically understand and analyze visual data. It enables faster and more accurate interpretation of images in fields like medical diagnosis, security surveillance, autonomous vehicles, industrial automation, and digital mapping. As images have become a major source of information, image processing helps extract meaningful patterns, detect objects, and convert visual information into usable digital data. This makes it essential for modern AI, automation, and intelligent decision-making systems.

1.2 Objectives

Objective of the project are listed below:
To detect nodes and edges from a given image and convert them into an accurate graph representation.
To compute the shortest path between any selected source and destination using efficient algorithms (e.g., Dijkstra).
To visualize all paths clearly on the image, helping users understand how the graph behaves.
To analyze and compare different paths based on distance, structure, and real-world relevance.

To create a tool that can be used for navigation, robotics, and autonomous systems research.

1.3 Contributions

This project contributes a practical and efficient solution for extracting graph structures from images and computing both shortest and alternative paths. It combines image processing with graph algorithms to create an automated path-finding system.
Key contributions are listed below:
Automated Graph Extraction
The project converts complex images into accurate node–edge graphs without manual intervention, reducing human effort and errors.
Reliable Path-Finding System By integrating Dijkstra and K-shortest path algorithms, the system not only finds the shortest route but also provides multiple alternative paths for better decision-making.
Real-Time Visualization
The project visually highlights paths directly on the input image, making it easier for users to understand how the algorithm operates.
Useful for Modern Applications
The approach can support autonomous vehicles, robotics, navigation tools, and research in graph-based image analysis.
Addresses a Real Gap
Many existing systems rely on manually drawn graphs or predefined datasets. This project bridges the gap by allowing graphs to be derived automatically from raw images, making the method more flexible and scalable.

1.4 Methodology Overview

To solve this problem, we first process the image to identify important features like nodes and connections. These are then converted into a graph structure so that algorithms like Dijkstra's can analyze all possible routes. Using this combination of image processing and graph theory, we can automatically detect valid paths and find the most efficient route between a chosen start and end point.

Chapter 2

Literature Review

2.1 Introduction to Literature Survey

The integration of unmanned vehicles with satellite and aerial imagery has revolutionized autonomous navigation systems across diverse applications including autonomous ground vehicles (UGVs), unmanned aerial vehicles (UAVs), and intelligent transportation systems. Path planning represents a critical component in autonomous navigation, enabling vehicles to determine optimal routes from source to destination while navigating complex environments and avoiding obstacles.

The convergence of remote sensing technology, image processing techniques, and algorithmic advances has created unprecedented opportunities for developing sophisticated path planning systems based on satellite imagery. Traditional path planning approaches often rely on pre-constructed high-precision maps or expensive commercial services, which limits accessibility and introduces dependency issues. In contrast, satellite imagery-based path planning leverages freely available geospatial data from services such as Google Maps, OpenStreetMap, and Bing Maps, providing cost-effective and readily accessible solutions for autonomous vehicle navigation in outdoor environments.

This literature survey examines the state-of-the-art in path planning for unmanned vehicles utilizing satellite images. The survey encompasses road detection methodologies, path planning algorithms, emerging deep learning approaches, and system integration challenges.

2.2 Existing Systems / Related Work

2.2.1 Traditional Road Detection Methods

Early approaches to road extraction from satellite imagery primarily relied on handcrafted features and image processing techniques. These methods typically employ a combination of color filtering, edge detection, and morphological operations to identify road regions. The foundational work in this area established that roads exhibit specific characteristics distinguishable from surrounding features, including constant width, low curvature, and relatively uniform spectral properties [1].

Classical techniques leverage multiple image processing filters including Difference of Gaussian

(DoG) filters to enhance road boundaries and emphasize edges between regions of different intensities. The Radon transform has been particularly effective for detecting linear structures characteristic of roads, especially when applied to grid sub-regions to identify local road segments [1]. These methods often incorporate color-based probability models trained on local regions to accommodate varying road surface materials such as asphalt, concrete, gravel, and unimproved earth surfaces.

A significant contribution in this domain was the hybrid approach combining road map image analysis with satellite image processing. This methodology exploits prior knowledge from online map services to simplify road detection while supplementing incomplete annotations through satellite image analysis [1]. The approach demonstrates that road network detection can be achieved with acceptable accuracy (approximately 95%) at lower computational costs compared to pure satellite image-based methods.

2.2.2 Deep Learning-Based Road Segmentation

The emergence of deep learning has fundamentally transformed road detection from satellite imagery. Convolutional Neural Networks (CNNs) have demonstrated superior capability in learning complex texture patterns and handling various environmental conditions without explicit feature engineering [2][3]. Seminal work by Mnih and colleagues established that neural networks trained on high-resolution aerial imagery could automatically detect roads with reasonable accuracy, introducing the Massachusetts Roads Dataset as a benchmark resource[4].

Fully Convolutional Networks (FCNs) represent a major architectural advancement, enabling end-to-end learning of pixel-wise classification tasks. FCNs eliminate the need for fully connected layers while maintaining the ability to process images of arbitrary size [5][6][7]. This architecture allows efficient semantic segmentation of road regions directly from satellite imagery.

The U-Net architecture, originally developed for biomedical image segmentation, has become the dominant framework for road extraction tasks. U-Net's symmetric encoder-decoder structure with skip connections effectively preserves low-level spatial details critical for accurate road boundary delineation. Recent studies demonstrate U-Net achieves approximately 92% overall accuracy in road surface classification from aerial imagery [6]. The architecture's ability to transfer low-level features to high-level processing through skip connections proves particularly valuable for maintaining road continuity and accurately delineating boundaries.

Advanced variants of U-Net incorporating residual blocks and attention mechanisms have further improved performance. Residual U-Net (RU-Net) combines advantages of U-Net, residual learning, atrous spatial pyramid pooling, and focal loss to handle class imbalance problems typical in road segmentation tasks [8]. These enhanced architectures achieve competitive performance on benchmark datasets while improving training stability through reduced network degradation in deeper layers.

DeepLabV3+ has emerged as another prominent architecture for road extraction, utilizing atrous convolution for multi-scale feature extraction. Recent research introduces Dense Depthwise Dilated Separable Spatial Pyramid Pooling (DenseDDSSPP) modules to enhance extraction of complex road structures, demonstrating superior performance over conventional Atrous Spatial Pyramid Pooling approaches [9].

2.2.3 Advanced Neural Network Architectures

Recent advances extend beyond traditional CNN architectures to incorporate attention mechanisms and transformer-based approaches. Vision Transformers (ViTs) represent a paradigm shift by replacing convolutional operations with self-attention mechanisms, enabling better capture of global context and long-range dependencies [10][11]. These approaches prove particularly effective for handling occlusions and complex road geometries that challenge conventional CNNs.

Hybrid architectures combining multiple neural network paradigms have shown promising results. Research introducing MambaVision-based frameworks with transformer capabilities demonstrates improved 3D lane and road detection even with elevation variations and complex road structures [11]. Attention-based mechanisms within semantic segmentation networks enable models to focus on relevant features while suppressing background noise and vegetation occlusions.

Graph Neural Networks (GNNs) introduce a novel paradigm for road extraction by directly predicting road graph structures. Rather than generating pixel-level segmentation maps followed by post-processing vectorization, GNN-based approaches combine Fully Convolutional Networks for identifying points of interest (intersections, turns, dead ends) with Graph Neural Networks for predicting connectivity between these points [12]. This end-to-end approach eliminates extensive post-processing while maintaining computational efficiency suitable for embedded devices.

Combination of CNN and GNN frameworks achieve regularized road surface extraction approaching human-level delineation quality. These methods formulate road surface extraction as two-sided width inference, utilizing CNN-based feature extractors and GNN models for global optimization of road width regularization [13]. Such integrated approaches demonstrate superior intersection over union scores and improved visual quality compared to pure CNN-based methods.

2.2.4 Path Planning Algorithms

Path planning algorithms form the computational backbone for autonomous navigation. Classical algorithms remain fundamental to contemporary systems despite recent deep learning advances.

Dijkstra's Algorithm provides guaranteed optimal shortest path detection in weighted graphs with non-negative edge costs. Its application to road network graphs derived from satellite imagery ensures minimal distance travel for autonomous vehicles [1]. While computationally efficient for moderate graph sizes, Dijkstra's algorithm exhibits increased computational complexity in large-scale urban environments.

A* Algorithm enhances Dijkstra's approach through heuristic guidance, reducing the search space by prioritizing nodes predicted closer to the goal. The algorithm combines the proven optimality of Dijkstra with heuristic functions, proving particularly effective for GPS-aided navigation systems and real-time applications [14][15][16].

Rapidly-exploring Random Trees (RRT) represent sampling-based planning approaches that generate feasible paths by exploring configuration space incrementally. RRT-based methods excel in high-dimensional spaces and complex obstacle environments, demonstrating capability to escape local minima more effectively than deterministic algorithms [3][17]. RRT variants including RRT* introduce asymptotic optimality through incremental rewiring, though at increased computational cost [17].

Probabilistic Roadmap (PRM) methods construct multi-query roadmaps enabling efficient path queries across diverse start-end configurations. Unlike RRT's single-query focus, PRM's roadmaps remain valid for multiple mission profiles, proving advantageous for fleet operations [18][19]. Temporal PRM extensions incorporate dynamic obstacle constraints by encoding time-dependent collision information within the roadmap structure [20].

Artificial Potential Field (APF) algorithms model obstacles as repulsive forces and goals as attractive forces, enabling real-time path computation. APF methods integrate naturally with collision avoidance frameworks and support dynamic environments, though susceptibility to local minima remains a recognized limitation [15].

Best-First Search (BFS) and **Depth-First Search (DFS)** provide foundational graph exploration strategies employed within hybrid planning frameworks. BFS guarantees shortest paths in unweighted graphs while DFS offers memory efficiency, with performance characteristics varying based on environmental structure [15][16].

2.2.5 Multi-Agent and Cooperative Path Planning

Multi-agent path planning addresses coordination among multiple autonomous vehicles in shared environments. Conflict Detection and Resolution (CDR) methods ensure collision-free trajectories across vehicle fleets, implementing pre-flight planning phases to resolve conflicts before mission execution [21].

Cooperative A* (CA*) extends single-agent A* planning to multiple agents through iterative conflict resolution and replanning. Enhanced Conflict-Based Search (ECBS) improves computational efficiency through batching mechanisms, proving more time-efficient for dynamic fleet management scenarios [21].

Multi-agent optimization techniques including clustering and task allocation use K-means or Fuzzy C-means methods to distribute targets among UAVs. Combined with route optimization algorithms and collision avoidance verification, these approaches enable efficient large-scale fleet coordination [22].

Heterogeneous multi-agent frameworks incorporate agents with distinct capabilities and priorities. Recent research demonstrates cooperative approaches where low-priority agents conduct exploration and threat localization while high-priority agents navigate to destinations with risk minimization, utilizing decentralized optimization and probabilistic constraints [23][24].

2.2.6 Integration of Road Detection with Path Planning

Integrated systems combine road detection outputs directly with path planning algorithms. The reference paper by Hoang et al. demonstrates a complete pipeline: road networks detected from map and satellite images are converted to Mercator coordinate systems and graph representations, then processed through Dijkstra's algorithm with BFS heuristics to compute optimal vehicle trajectories [1].

The extracted road network, represented as a directed graph with vertices corresponding to intersections and endpoints, and edges representing road segments, provides natural input to graph search algorithms. Euclidean distances between connected vertices define edge costs, enabling shortest path computation under the assumption of no dynamic obstacles [1].

This integration approach achieves significant improvements over pure satellite image-based methods or isolated online map services. By combining the strengths of both information sources prior knowledge from map services ensuring high confidence and satellite images detecting unmapped road segments—integrated systems construct comprehensive road networks suitable for real-world autonomous navigation [1].

2.2.7 Benchmark Datasets

Massachusetts Roads Dataset comprises 1,171 high-resolution aerial images (1,500 × 1,500 pixels) covering 2.25 square kilometers each, spanning over 2,600 square kilometers across Massachusetts with diverse urban, suburban, and rural environments. The dataset's training set contains 1,108 images, validation set 14 images, and test set 49 images, enabling comprehensive evaluation of road detection algorithms [4][25].

DeepGlobe 2018 Dataset provides high-resolution satellite imagery specifically designed for semantic segmentation tasks including road extraction. This dataset facilitates training and evaluation of deep learning models across diverse geographic regions and environmental conditions [26].

RoadTracer Dataset enables evaluation of road graph extraction methods through iterative graph construction approaches. Graph-based methods are benchmarked against baseline performance using this resource [12][27].

These benchmark datasets have proven instrumental in advancing road detection methodologies by providing standardized evaluation protocols and enabling direct comparison among competing approaches.

2.3 Limitations of Existing Systems

2.3.1 Road Detection Challenges

Despite significant advances, road detection from satellite imagery faces persistent technical challenges:

Occlusion Issues remain the most significant limitation. Dense vegetation, shadows from buildings and trees, tunnels, and overpasses frequently obscure road surfaces, rendering them invisible in satellite imagery. Spectral properties of vegetation and building shadows closely resemble some road surface characteristics, creating ambiguity in pixel-level classification [28][29][30].

Spectral Variability of road surfaces complicates detection algorithms. Different materials—asphalt, concrete, brick, gravel, and unpaved surfaces—exhibit distinct spectral signatures. Within-material variations due to age, weathering, and different manufacturing processes introduce additional complexity. Model adaptation across diverse road types and geographic regions requires substantial retraining effort [1][28].

Low Image Resolution and degraded image quality in certain regions limit detection accuracy. While high-resolution satellite imagery (sub-meter resolution) facilitates detection, many regions lack access to such data. Public satellite services provide varying resolutions depending on geographic location and temporal coverage [25].

Class Imbalance presents a persistent challenge in deep learning-based approaches. Road pixels constitute a small fraction of satellite images relative to background pixels (buildings, vegetation, water, non-road areas). This class imbalance biases models toward conservative predictions, reducing sensitivity to small or faint road segments [28][29].

Road Connectivity and Continuity errors frequently occur when detection models fail to maintain connection between road segments, particularly at intersections or when roads pass through tree cover. Short-distance discontinuities in detection results require sophisticated post-processing and geometric constraints to resolve [1][28].

2.3.2 Computational Complexity and Real-Time Constraints

High Computational Overhead characterizes many state-of-the-art deep learning approaches. Large CNN models with numerous parameters require substantial GPU resources for inference. While essential for maximum accuracy, this computational demand conflicts with real-time, on-board processing requirements for autonomous vehicles [31].

Embedding Constraints on autonomous vehicles limit deployable model complexity. Edge devices aboard UAVs and ground vehicles possess limited memory, processing power, and energy budgets. Balancing model accuracy against these resource constraints remains an open research challenge [31].

Real-Time Processing Latency requirements necessitate path planning computation within seconds rather than minutes. Complex sampling-based algorithms like RRT and PRM incur high computational costs in large-scale environments, while A* algorithm computational complexity increases significantly in highly populated graphs characteristic of detailed urban maps [16].

2.3.3 Path Planning Algorithm Limitations

Graph Size and Complexity in dense urban environments create computational bottlenecks. As road networks grow more complex with numerous intersections and branch connections, search-based algorithms experience increased memory requirements and extended computation times [16].

Local Minima Susceptibility affects potential field-based methods, where vehicles become trapped in local attractive-repulsive force equilibria without reaching actual destinations. While random sampling methods like RRT mitigate this issue, they introduce non-deterministic behavior and variable solution quality [15][32][33].

Dynamic Obstacle Handling presents challenges for static path planning approaches. Many classical algorithms assume static environments, requiring substantial modifications for incorporation of dynamic obstacles (moving vehicles, pedestrians) or time-dependent constraints [32][20][34].

Optimality Guarantees versus Computational Efficiency present fundamental tradeoffs. Algorithms guaranteeing optimal solutions (Dijkstra, A*) require extensive search, while approximate methods (RRT, heuristic-based approaches) compute faster with suboptimal results [15][16].

2.3.4 Environmental and Operational Constraints

GPS Denial and GNSS Availability limitations prevent reliance on satellite positioning in contested environments, tunnels, urban canyons, and other areas with signal blockage. Alternative localization approaches including visual odometry, SLAM, and inertial navigation systems incur additional computational overhead and calibration complexity [35][36][37].

Weather Conditions and Seasonal Variations affect satellite imagery quality and road detection accuracy. Cloud cover obscures underlying road networks, while seasonal vegetation changes alter spectral characteristics. Temporal resolution limitations restrict update frequency for dynamic road conditions [29][30].

Data Currency and Update Frequency issues plague map-based approaches. Road networks undergo constant change through construction, closures, and new development. Satellite imagery temporal resolution may lag actual ground conditions by weeks or months, introducing potential navigation hazards [1][29].

2.3.5 Integration and System-Level Challenges

End-to-End Optimization between detection and planning stages remains underdeveloped. Road detection accuracy directly impacts subsequent path planning results, yet optimization frequently proceeds independently. Detection errors propagate through planning stages without mitigation mechanisms [1].

Uncertainty Propagation from detection stages through planning produces cumulative errors in final trajectories. Confidence metrics and uncertainty quantification in detection outputs could inform more robust planning but are rarely integrated [38][28].

Transfer Learning and Generalization limitations restrict model applicability across diverse geographic regions and image sources. Models trained on specific datasets often experience significant performance degradation on new regions with different imaging characteristics, road construction standards, and environmental conditions [9][28].

Validation and Safety Assurance frameworks remain underdeveloped for safety-critical autonomous vehicle applications. Limited flight test validation exists for end-to-end systems, with most evaluations conducted in simulation environments that may not capture real-world complexity [39][17].

2.4 Comparative Study / Gap Analysis

2.4.1 Method Comparison: Handcrafted Features vs. Deep Learning

Handcrafted Feature-Based Approaches provide interpretable processing pipelines with well-understood behavior characteristics. These methods require limited training data, operate with low computational requirements, and enable direct incorporation of domain knowledge through carefully designed filters and processing sequences. However, they suffer from limited adaptability across diverse environments, require manual parameter tuning for different geographic regions, and typically achieve lower accuracy than deep learning methods, particularly in challenging conditions with occlusions and spectral variability [40][4][7][41].

Deep Learning Approaches automatically learn discriminative features from large training datasets, eliminating handcrafted feature engineering. These methods achieve substantially higher accuracy (90%+ for well-trained models) and demonstrate better generalization across diverse environmental conditions. However, they require extensive labeled training data, demand significant computational resources for training and inference, exhibit limited interpretability ("black box" behavior), and risk overfitting to dataset-specific characteristics with poor cross-dataset generalization [2][3][5][6][7].

Recent trends favor deep learning approaches for production systems despite higher complexity, as accuracy improvements often justify increased resource requirements. Hybrid approaches combining CNN feature extraction with traditional geometric constraints show promise in balancing accuracy and interpretability [7][28][13].

2.4.2 Path Planning Algorithm Comparative Analysis

Table 2.1: Path Planning Algorithm Comparative Analysis

Algorithm	Optimality	Real-Time Capability
Dijkstra	Guaranteed optimal	Moderate; scales poorly with graph size
A*	Guaranteed optimal (with admissible heuristic)	Good; faster than Dijkstra with effective heu
RRT	No guarantee	Good; real-time capable
RRT*	Asymptotically optimal	Moderate; slower than RRT
PRM	No guarantee (multi-query)	Good after learning phase
T-PRM	Maintains PRM properties with time	Moderate; searches temporal roadmap
APF	No guarantee	Excellent; $O(1)$ computation

2.4.3 Integration Gaps Between Detection and Planning

Current systems frequently operate detection and planning stages independently without joint optimization:

- Detection optimization targets pixel-level accuracy metrics (precision, recall, IoU) without considering downstream planning impacts
- Planning algorithms assume perfectly accurate road networks without modeling detection uncertainty
- End-to-end training frameworks jointly optimizing detection and planning components remain rare
- Detection errors propagate through planning with limited error mitigation mechanisms

Addressing these integration gaps could yield substantial improvements in overall system performance and robustness.

2.5 Summary of Literature Review

2.5.1 Key Findings

The literature review reveals a maturing field with accelerating progress driven by deep learning advances, expanding dataset availability, and increasing computational resources. Several key findings emerge:

1. Deep Learning Dominates Current Practice CNN-based approaches have substantially displaced handcrafted feature methods in production systems. U-Net and its variants achieve >90% accuracy on benchmark datasets, with Vision Transformers pushing performance toward 96

2. Hybrid Integration Approaches Show Promise Systems combining satellite images with online map services (e.g., Google Maps, OpenStreetMap) demonstrate superior performance compared to either source alone. This hybrid strategy leverages the strengths of each data source while mitigating individual limitations. Road detection accuracy of 95% combined with low computational costs validates this approach.

3. Classical Path Planning Algorithms Remain Fundamental Despite emergence of learning-based approaches, Dijkstra, A*, and probabilistic methods remain core to production systems. These algorithms provide optimality guarantees, well-understood behavior, and integration with existing infrastructure. Recent extensions incorporating time-dependent constraints (T-PRM) and dynamic obstacles demonstrate continued relevance.

4. Persistent Technical Challenges Constrain Performance Occlusion by vegetation and buildings remains the most significant obstacle to fully automatic road detection. Class imbalance in training data, spectral variability of road surfaces, and spectral similarity between roads and backgrounds continue to limit accuracy even for state-of-the-art methods.

5. Real-Time Processing Constraints Drive Research Embedding autonomous vehicle systems necessitate real-time processing on resource-constrained platforms. This requirement drives interest in lightweight architectures (ESANet, MobileNets) and classical algorithms despite their potential suboptimality compared to more complex methods.

6. Multi-Agent Coordination Frameworks Mature Cooperative planning methods for UAV fleets (ECBS, multi-agent optimization, conflict-based search) demonstrate capability for coordinating multiple vehicles in shared airspace. These approaches represent significant advancement for future autonomous systems.

7. Uncertainty Integration Remains Underdeveloped Few systems explicitly incorporate uncertainty from detection stages into planning algorithms. This gap represents a critical opportunity for improving robustness and safety through probabilistic approaches.

2.5.2 Conclusion

Path planning for unmanned vehicles based on satellite images has evolved into a mature research domain combining advances in remote sensing, computer vision, machine learning, and robotics. Deep learning methods have substantially improved road detection accuracy, while classical planning algorithms remain fundamental to production systems. However, persistent technical challenges—particularly occlusions and real-time processing constraints—continue to drive research innovation.

The convergence toward hybrid systems combining satellite imagery with online maps, deep learning perception with classical planning algorithms, and global path planning with local obstacle avoidance represents the current state-of-practice. Future advancement lies in addressing integration gaps between detection and planning, incorporating uncertainty throughout the system, and developing robust frameworks suitable for deployment in diverse, GPS-denied environments where autonomous vehicle systems must operate reliably under real-world constraints.

The field stands at an inflection point where mature individual components (detection algorithms, planning methods, processing platforms) are increasingly integrated into coherent systems. This integration represents both the greatest opportunity and most significant challenge for advancing practical autonomous vehicle systems capable of operating safely and efficiently in complex, real-world environments.

Chapter 3

Problem Identification

3.1 Problem Statement

The core problem addressed in this project is the automated extraction and computation of shortest paths between nodes within graph-based representations of navigable environments obtained from image data. Traditional pathfinding depends on manually created high-precision maps or costly commercial spatial databases, which are often outdated or unsuitable for real-world use. This project aims to bridge the gap between image processing and algorithmic navigation by automatically deriving navigable networks from visual sources (e.g., satellite images or corridor maps), applying efficient shortest-path algorithms, and enabling reuse of precomputed paths for future tasks. The main challenge is to achieve accuracy, computational efficiency, and adaptability across diverse environments while reducing reliance on external map resources.

3.2 Context and Background of the Problem

Path planning in complex environments is a foundational challenge in image processing, autonomous systems, and intelligent navigation. Accurate detection, interpretation, and traversal of spaces—from indoor layouts to expansive road networks—directly influence fields such as robotics, autonomous vehicles, transportation automation, and smart infrastructure. Modern systems increasingly require navigation solutions that operate even when traditional mapping resources are unavailable or impractical.

This project focuses on integrating computer vision with graph-based pathfinding. By extracting spatial information from visual inputs and converting it into computational graph structures, the system enables autonomous agents to make navigation decisions solely from image data. This capability is especially important for unmanned vehicles, mobile robots, emergency response operations, and environments where rapid adaptation to changing or previously unmapped spaces is required.

The broader relevance of this work extends beyond technical innovation. As urbanization and autonomous technologies continue to grow, the need for reliable, scalable, and cost-effective visual-based navigation solutions becomes increasingly significant. Automating the interpretation of visual environments and generating navigation paths directly supports real-world demands across transportation, security, and disaster-response applications.

3.3 Need and Importance of Solving the Problem

Solving the identified pathfinding challenges holds strong practical and theoretical importance. Developing automated navigation systems directly benefits several key areas:
Autonomous Systems: Unmanned vehicles, drones, and mobile robots require reliable navigation without depending on external maps. Automated path extraction from imagery increases autonomy, safety, and operational range.
Emergency Response: During disasters, existing maps may be outdated or infrastructure may be damaged. Systems that derive navigation data from current imagery enable faster, more accurate response and resource deployment.
Cost Reduction: Automating map creation and path planning reduces expenses associated with manual surveying, licensing, and database maintenance.
Operational Efficiency: Reusing previously computed paths lowers computational load, allowing systems to handle more queries or operate effectively on limited hardware.
From a broader perspective, improved autonomous navigation enhances transportation safety, accessibility, and smart-city automation. Theoretically, this work advances the integration of computer vision and algorithmic planning, demonstrating practical applications of image processing, graph theory, and optimization.

3.4 Scope and Project Objectives:

This project aims to develop and validate an integrated system for image-based path planning with clear, measurable objectives. The main objectives are:
Graph Extraction: Create robust image-processing pipelines to automatically extract node-and-edge graph structures from visual data such as satellite images and corridor maps. This includes preprocessing, feature detection, network construction, and coordinate transformation.
Shortest Path Implementation: Integrate efficient shortest-path algorithms to compute optimal routes between specified source and target nodes, prioritizing both accuracy and computational performance.
Path Storage and Reuse: Develop mechanisms to store previously computed paths in a structured format and determine when stored paths can be reused, reducing redundant calculations.
Validation and Testing: Assess system performance using real-world test data, including indoor layouts and satellite imagery. Evaluation will focus on path accuracy, computation time, and improvements achieved through path reuse.

Documentation and Analysis: Provide detailed documentation of the methodology, implementation, experimental results, and comparative evaluation against existing approaches.
The project scope focuses solely on algorithmic and computational aspects of path planning using standard visual data, excluding advanced sensing technologies or scenarios beyond typical imagery conditions.

3.5 Anticipated Challenges

The project involves several technical challenges that must be addressed for successful system development:
Image Processing Accuracy: Extracting reliable network structures from images requires handling noise, occlusions, illumination changes, and visual ambiguities. Ensuring that the generated graph accurately reflects actual navigable paths is essential.
Algorithm Integration: Connecting image-processing outputs with graph-based pathfinding algorithms demands careful alignment of data structures and coordinate systems. Any mismatch can introduce cumulative errors that reduce path quality.
Dynamic Environments: Real-world environments evolve over time as obstacles appear or routes change. The system must either adapt to such variations or clearly specify its operational limits.
Computational Performance: Achieving both accuracy and real-time speed is challenging. Advanced image processing and thorough path searches improve quality but may slow performance, requiring efficient optimization.
Path Storage Efficiency: Reusing shortest paths requires effective data structures, balanced storage requirements, and fast retrieval methods. The system must manage memory constraints while supporting efficient reuse.
Validation Complexity: Rigorous validation across diverse scenarios demands extensive datasets and well designed experiments. Ensuring that system performance generalizes to real-world settings requires comprehensive evaluation.

Chapter 4

Proposed Methodology

4.1 Introduction to Methodology

The core problem addressed in this project is the automatic construction of a complete and topologically correct road-network graph from visual map data, enabling reliable shortest-path computation in regions where conventional digital maps such as OpenStreetMap remain incomplete. A systematic and structured methodology is required because the input image contains noise, variable coloring, occlusions, and hand-drawn structures that cannot be directly converted into graph form without multiple stages of image processing, feature extraction, and graph validation.

To solve this, the project adopts a computer-vision-driven analytical pipeline that transforms a corridor-based image into a valid navigation graph. The methodology combines image preprocessing, color-based vertex extraction, corridor sampling for edge validation, and classical graph-search algorithms to ensure precise node detection and robust connectivity estimation. This approach uses deterministic, interpretable image-processing techniques paired with Dijkstra's shortest-path algorithm, resulting in a fully autonomous, explainable, and resource-efficient solution.

4.2 System Architecture / Conceptual Framework

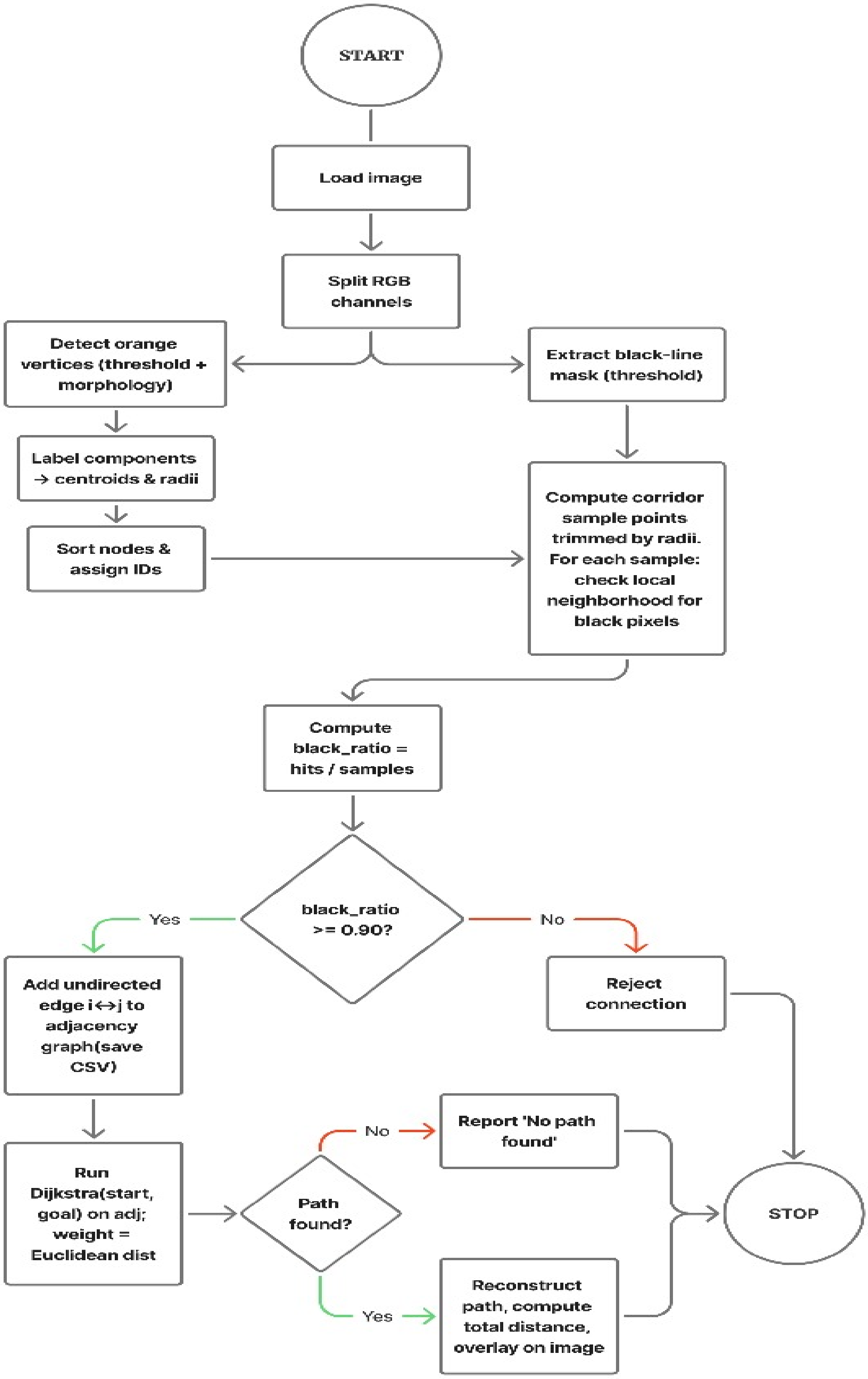

Figure 4.1: Flowchart of the methodology.

4.3 Algorithms / Techniques Used

4.3.1 Color-Based Image Segmentation (RGB Thresholding)

The system uses deterministic RGB thresholding to detect the orange junction points and black corridor lines in the input image. For orange vertex detection, pixel-level rules such as:
$R > 200$
$90 < G < 200$
$B < 90$
are applied to isolate orange-colored road intersections. Similarly, pixels with:
$R < 50$, $G < 50$, $B < 50$
are classified as black corridor pixels representing road segments. This approach was chosen because the input is a rendered map with high color consistency, making traditional thresholding more robust and faster than a CNN-based detector. Additionally, thresholding avoids the need for training data and ensures deterministic, explainable results.

4.3.2 Morphological Operations(Opening and Closing:)

Noise in the thresholded mask is removed using:
Binary Opening-removes small isolated pixels
Binary Closing-fills small gaps between connected orange pixels
These steps ensure that each junction becomes a single clean connected component, enabling accurate centroid estimation. Morphology is preferred over complex filtering because it preserves structural shape while cleaning noise.

4.3.3 Connected Component Analysis (Node Extraction)

After cleaning the orange mask, the system applies connected component labeling to group pixels belonging to the same node. For each component:
Centroid
Radius (max pixel distance from centroid)
are computed.
This algorithm ensures consistent detection of node locations regardless of small variations in drawing thickness or shape. It is chosen over clustering algorithms like K-means because the regions are already spatially connected, making connected-components the most efficient and accurate method.

4.3.4 Corridor Sampling for Edge Validation (Black Ratio Test)

For every pair of nodes (i, j):
1. A straight line is drawn between centroids.
2. The sample range is trimmed by the node radii (to avoid sampling inside nodes).
3. A series of n = 220 equally spaced sample points is generated.
4. Around each sample, a neighborhood of radius = 2 pixels is scanned for black pixels.
5. black ratio = hits / samples
6. If black ratio ≥ 0.75, the nodes are considered connected.

This corridor sampling method replaces alternatives like skeletonization or Hough transform because:
It is more reliable on hand-drawn, imperfect lines.
It allows detection of corridors even if slightly broken or irregular.
It minimizes false edges between nodes that are geometrically close but not connected.

4.3.5 Adjacency Graph Construction

All validated connections are stored in an undirected adjacency list:
adj[node] = list of neighbors
This representation is memory-efficient, supports fast lookups, and integrates seamlessly with shortest-path algorithms. The graph is also exported to a CSV file for verification and reproducibility.

4.3.6 Dijkstra's Algorithm for Shortest Path Computation

The system uses Dijkstra's Algorithm to compute the minimum-cost route between a selected start and goal node. Edge weights are computed using Euclidean distance between node centroids, ensuring that longer road segments incur higher traversal cost.
Dijkstra is chosen over A* because:
The map does not provide heuristic information in real-world coordinates
The graph is small ($\approx$ 30-40 nodes), making Dijkstra computationally efficient
Exact shortest-path guarantees are required

4.4 Expected Outcome

The proposed system is expected to successfully process a corridor-based map image and convert it into a structured, navigation-ready graph. By detecting the orange junction points, the system will accurately identify and localize all road intersections present in the image. Using the black corridor lines, it will then validate real connections between these junctions and construct a complete adjacency representation of the road network. Based on the generated graph, the system will be able to compute a shortest path between any selected start and destination node using Dijkstra's algorithm. The final output will be a visual representation of the optimal route overlaid on the original image, along with the corresponding node sequence that forms this path.

In future extensions, the proposed system can be expanded to work directly on real satellite or aerial images instead of manually prepared corridor diagrams. The node and corridor detection process can be adapted to handle more complex environments, including irregular road shapes, intersections, and varying lighting conditions. GPS coordinate mapping may be integrated so that detected nodes and paths correspond to real-world locations. Additionally, the graph construction step can be extended to include weighted information such as road width or surface type, making the system more useful for real-world autonomous navigation applications.

Chapter 5

Results and Discussion

graphicx

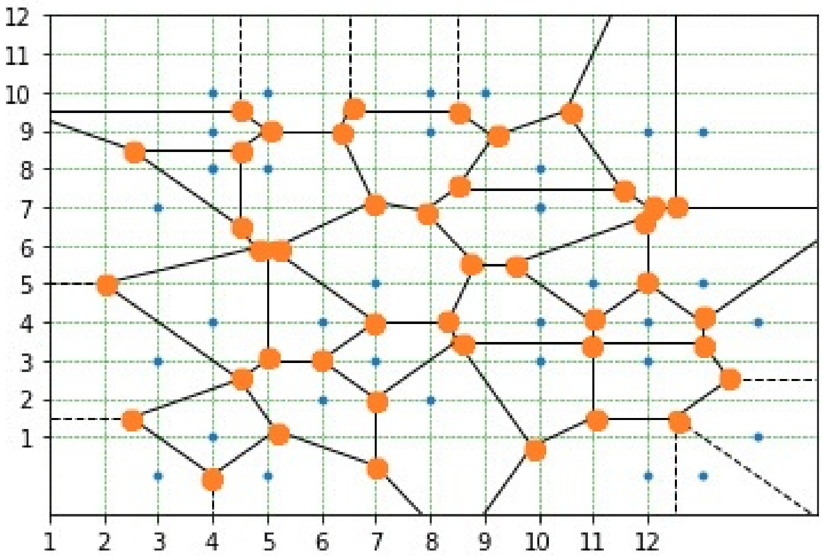

Figure 5.1: Input Image to test the algorithm

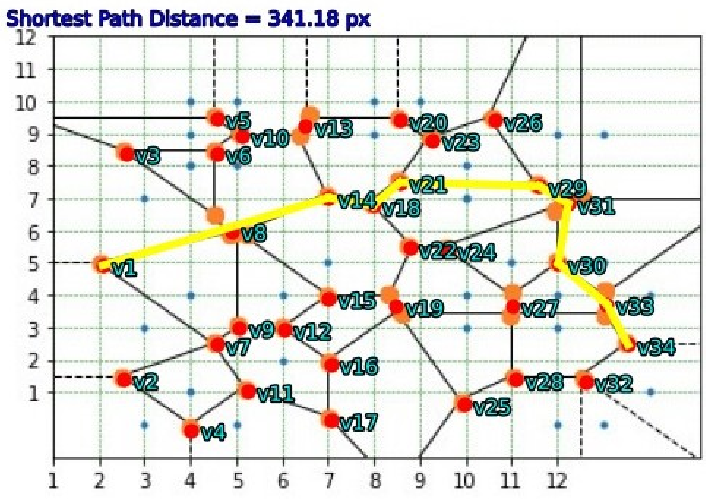

Figure 5.2: Output recieved

5.1 Experimental results and discussion

The results show that the algorithm successfully extracts the graph from the input image, detects nodes and edges using color segmentation and corridor sampling, and computes the shortest path between specified start (V1) and end(V34) points.This confirms the algorithm's capability to process the environment, validate connectivity, and deliver an effective route for navigation between input and output points as designed.

5.1.1 Comparison with existing state-of-the-art methods

This image-based pathfinding methodology is competitive because it constructs a graph directly from an RGB environment image using color segmentation and pixel-level corridor sampling to validate edges, ensuring paths correspond to actual traversable corridors. Unlike traditional algorithms that rely on predefined graphs, grids, or Voronoi diagrams, this method integrates image processing to dynamically extract navigation nodes and connections from visual data. While classical algorithms like A* or Dijkstra focus on weighted graphs or grid maps with heuristic guidance, this approach uses an unweighted graph derived from the image and applies Dijkstra's algorithm for shortest path computation. This makes it particularly suitable for complex, irregular environments represented visually rather than as explicit spatial graphs, offering a unique blend of image understanding and pathfinding compared to existing methods.

5.1.2 Novelty of the proposed methodology

The follow are the Novelty features of our methodology:

- **Image-Based Graph Construction:** The algorithm constructs a graph from an RGB image representing a spatial environment. It identifies vertices by detecting orange-colored areas (likely representing nodes) using color thresholding on the image and cleans this mask using morphological operations. Edges are derived by sampling along corridors (black areas in the image) between vertices, evaluating connectivity based on a "black pixel ratio" criterion.

- **Corridor Sampling for Edge Validation:** Instead of connecting nodes merely by distance or explicit graph data, the algorithm samples pixels along the corridor between nodes to ensure a high ratio (>= 95%) of black pixels (indicating free or traversable space). This effectively builds edges only along navigable paths extracted from the image.

- **Radius-Based Corridor Boundaries:** When sampling the corridor between two nodes, the algorithm uses the radius of each vertex to define the start and end boundaries of sampling, excluding areas inside the nodes themselves. This provides a more accurate corridor representation relative to node size.

Chapter 6

Conclusion and Future Work

In conclusion, this project successfully demonstrates an image-based pathfinding algorithm that extracts navigable graphs from environmental images and computes the shortest path between designated points. The algorithm integrates image processing with graph search techniques to handle complex, visually represented spaces. Future work will focus on refining the algorithm to enhance accuracy and efficiency, followed by prototyping a practical application—such as aiding a service robot in navigating autonomously within a defined perimeter—thereby extending its utility to real-world robotic navigation scenarios.

Bibliography

[1] V. Hoang. “D. Caceres Hernandez”. In: (2014). V. Hoang, D. Caceres Hernandez, A. Filonenko, K. Jo, "Path Planning for Unmanned Vehicle Motion Based on Road Detection Using Online Road Map and Satellite Image," in Computer Vision - ACCV 2014 Workshops, pp. 1-15, 2015.

[2] D. Patil et al. “"Road Segmentation in High-Resolution Images Using Deep Convolutional Neural Network”. In: (2022). D. Patil et al., "Road Segmentation in High-Resolution Images Using Deep Convolutional Neural Network," Engineering, Technology Applied Science Research, 2022.

[3] J. Li et al. “"Path-planning algorithms for self-driving vehicles based on improved RRT-Connect algorithm”. In: (2023). J. Li et al., "Path-planning algorithms for self-driving vehicles based on improved RRT-Connect algorithm," Transportation Systems and Engineering, 2023.

[4] V. Mnih. “G. Hinton”. In: (2010). V. Mnih, G. Hinton, "Learning to Detect Roads in High-Resolution Aerial Images," in ECCV 2010, pp. 210-223, 2010.

[5] D. Patil et al. “"Road Segmentation in High-Resolution Images Using Deep Convolutional Neural Networks with U-Network and Residual Blocks”. In: (2022). D. Patil et al., "Road Segmentation in High-Resolution Images Using Deep Convolutional Neural Networks with U-Network and Residual Blocks," Engineering, Technology Applied Science Research, vol. 12, 2022.

[6] O. Pesek et al. “"Convolutional neural networks for road surface classification on aerial data”. In: (2024). O. Pesek et al., "Convolutional neural networks for road surface classification on aerial data," Remote Sensing, 2024.

[7] "Deep Convolutional Neural Networks with Color Lines Model for Unmarked Road Segmentation. “" in Proceedings of CVPR”. In: (2018). "Deep Convolutional Neural Networks with Color Lines Model for Unmarked Road Segmentation," in Proceedings of CVPR, 2018.

[8] H. Wang et al. “"Building extraction from remote sensing images using Residual U-Net”. In: (2022). H. Wang et al., "Building extraction from remote sensing images using Residual U-Net," Journal of Hydro-environment Research, vol. 42, 2022.

[9] A. Mahara et al. “"Automated Road Extraction from Satellite Imagery Integrating Dense Depthwise Dilated Separable Spatial Pyramid Pooling with DeepLabV3+”. In: (2024). A. Mahara et al., "Automated Road Extraction from Satellite Imagery Integrating Dense Depthwise Dilated Separable Spatial Pyramid Pooling with DeepLabV3+," arXiv:2410.14836, 2024.

[10] S. Satti et al. “"Potholes and traffic signs detection by classifier with vision transformers”. In: (2024). S. Satti et al., "Potholes and traffic signs detection by classifier with vision transformers," Nature Scientific Reports, 2024.

[11] "Hybrid MambaVision and Transformer-Based Architecture for 3D Lane Detection. “" PubMed”. In: (2024). "Hybrid MambaVision and Transformer-Based Architecture for 3D Lane Detection," PubMed, 2024.

[12] G. Bahl et al. “"Road Extraction from Overhead Images with Graph Neural Networks”. In: (2021). G. Bahl et al., "Road Extraction from Overhead Images with Graph Neural Networks," arXiv:2112.05215, 2021.

[13] J. Yan et al. “"A Combination of Convolutional and Graph Neural Networks for Regularized Road Surface Map Extraction”. In: (2022). J. Yan et al., "A Combination of Convolutional and Graph Neural Networks for Regularized Road Surface Map Extraction," IEEE Transactions on Geoscience and Remote Sensing, 2022.

[14] "Path Planning for Self-Driving Cars. “" Sapien.io Blog”. In: (2025). "Path Planning for Self-Driving Cars," Sapien.io Blog, 2025.

[15] Y. Ming et al. “"A Survey of Path Planning Algorithms for Autonomous Vehicles”. In: (2024). Y. Ming et al., "A Survey of Path Planning Algorithms for Autonomous Vehicles," 2024.

[16] L. Shi. “"Research on Path Planning Method based on Graph Search Algorithm”. In: (2024). L. Shi, "Research on Path Planning Method based on Graph Search Algorithm," in MCEE 2024, vol. 97, pp. 267-280, 2024.

[17] H. Liu et al. “"An Autonomous Path Planning Method for Unmanned Aerial Vehicles”. In: (2020). H. Liu et al., "An Autonomous Path Planning Method for Unmanned Aerial Vehicles," IEEE Transactions on Aerospace and Electronic Systems, vol. 56, 2020.

[18] J. Jermyn. “"A Comparison of the Effectiveness of the RRT”. In: (2021). J. Jermyn, "A Comparison of the Effectiveness of the RRT, PRM, and Novel Hybrid RRT-PRM Path Planners," International Journal of Research and Analytical Reviews, vol. 8, 2021.

[19] "Probabilistic Roadmap Path Planning. “" Columbia University Course Notes”. In: (2015). "Probabilistic Roadmap Path Planning," Columbia University Course Notes, 2015.

[20] M. Hüppi et al. “"T-PRM: Temporal Probabilistic Roadmap for Path Planning in Dynamic Environments”. In: (2022). M. Hüppi et al., "T-PRM: Temporal Probabilistic Roadmap for Path Planning in Dynamic Environments," in Proceedings of IROS 2022, 2022.

[21] F. Ho et al. “"Multi-Agent Path Finding for UAV Traffic Management”. In: (2019). F. Ho et al., "Multi-Agent Path Finding for UAV Traffic Management," in Proceedings of AAMAS 2019, pp. 131-139, 2019.

[22] R. Nagasawa et al. “"Model-based analysis of multi-UAV path planning for large-scale disaster assessment”. In: (2021). R. Nagasawa et al., "Model-based analysis of multi-UAV path planning for large-scale disaster assessment," Nature Scientific Reports, 2021.

[23] G. Stagg et al. “"Cooperative Multi-Agent Path Planning for Heterogeneous UAVs in Contested Environments”. In: (2025). G. Stagg et al., "Cooperative Multi-Agent Path Planning for Heterogeneous UAVs in Contested Environments," in AIAA Journal of Guidance, Control, and Dynamics (in review), arXiv:2509.02483, 2025.

[24] "Cooperative Multi-Agent Path Planning for Heterogeneous UAVs in Contested Environments. “" arXiv”. In: (2008). "Cooperative Multi-Agent Path Planning for Heterogeneous UAVs in Contested Environments," arXiv, 2008.

[25] "Massachusetts Roads Dataset. “" GTS.ai”. In: (2025). "Massachusetts Roads Dataset," GTS.ai, 2025.

[26] "Road Extraction from Satellite Images Using UNet. “" GitHub”. In: (2023). "Road Extraction from Satellite Images Using UNet," GitHub, 2023.

[27] Y. He et al. “"Top-Down Road Network Extraction with Holistic Graph Convolutional Layers”. In: (2022). Y. He et al., "Top-Down Road Network Extraction with Holistic Graph Convolutional Layers," in Proceedings of ECCV 2022, pp. 553-569, 2022.

[28] H. Bai et al. “"A dynamic attention mechanism for road extraction from satellite imagery”. In: (2025). H. Bai et al., "A dynamic attention mechanism for road extraction from satellite imagery," Nature Scientific Reports, 2025.

[29] Z. Chen et al. “"Road extraction in remote sensing data: A survey”. In: (2022). Z. Chen et al., "Road extraction in remote sensing data: A survey," International Journal of Remote Sensing, vol. 43, 2022.

[30] "Integrated approaches for road extraction and de-noising from satellite imagery. “" AIP Advances”. In: (2025). "Integrated approaches for road extraction and de-noising from satellite imagery," AIP Advances, vol. 14, 2025.

[31] "Vision-Aided Online A* Path Planning for Efficient and Semantically Aware Navigation. “" arXiv:2511.06801”. In: (2025). "Vision-Aided Online A* Path Planning for Efficient and Semantically Aware Navigation," arXiv:2511.06801, 2025.

[32] W. Xiaofeng et al. “"Obstacle Avoidance Path Planning Strategy for Autonomous Vehicles Based on Genetic Algorithm”. In: (2024). W. Xiaofeng et al., "Obstacle Avoidance Path Planning Strategy for Autonomous Vehicles Based on Genetic Algorithm," Promet - Traffic Transportation, vol. 36, pp. 733-748, 2024.

[33] K. Katona et al. “"Obstacle Avoidance and Path Planning Methods for Autonomous Mobile Robots”. In: (2024). K. Katona et al., "Obstacle Avoidance and Path Planning Methods for Autonomous Mobile Robots," PubMed, 2024.

[34] M. Lashin et al. “"Real-time path planning in dynamic environments using predictive heuristics”. In: (2025). M. Lashin et al., "Real-time path planning in dynamic environments using predictive heuristics," ScienceDirect, 2025.

[35] A. Wu et al. “"Autonomous Flight in GPS-Denied Environments Using Monocular Vision and Inertial Sensors”. In: (2012). A. Wu et al., "Autonomous Flight in GPS-Denied Environments Using Monocular Vision and Inertial Sensors," IEEE Transactions on Aerospace and Electronic Systems, vol. 48, 2012.

[36] B. Allik et al. “"Localization for Aerial Systems in GPS Denied Environments”. In: (2022). B. Allik et al., "Localization for Aerial Systems in GPS Denied Environments," in Proceedings of AIAA Aviation Forum, 2022.

[37] "GNSS-Denied Navigation Kit. “" UAV Navigation Systems”. In: (2024). "GNSS-Denied Navigation Kit," UAV Navigation Systems, 2024.

[38] P. Yin et al. “"Towards Satellite Image Road Graph Extraction: A Global-Scale Dataset and a Novel Approach with SAM-Road++”. In: (2025). P. Yin et al., "Towards Satellite Image Road Graph Extraction: A Global-Scale Dataset and a Novel Approach with SAM-Road++," in Proceedings of CVPR 2025, 2025.

[39] J. Park et al. “"Deep Reinforcement Learning based Real-Time Path Planning for UAVs”. In: (2024). J. Park et al., "Deep Reinforcement Learning based Real-Time Path Planning for UAVs," in Proceedings of AIAA SciTech Forum, 2024.

[40] "Detecting Roads from Aerial Images using Deep Learning. “" TodaysSoftMag”. In: (2016). "Detecting Roads from Aerial Images using Deep Learning," TodaysSoftMag, 2016.

[41] "A Deep Learning Approach for Road Segmentation. “" ICTACT Journal on Image and Video Processing”. In: (2022). "A Deep Learning Approach for Road Segmentation," ICTACT Journal on Image and Video Processing, vol. 13, 2022.