

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

An Embedded System Exploiting Multi-Domain Urban Maps to Support Safer Cycling

Tiago Henrique Almeida de Sousa



Mestrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Prof. Daniel Gouveia Costa

September 15, 2025

Resumo

Num contexto de crescimento acentuado da população urbana e de limitada capacidade de resposta dos sistemas de transporte público tradicionais, tem-se verificado um aumento significativo da utilização da bicicleta como meio de transporte principal. Em resposta a esta tendência, as cidades inteligentes têm procurado soluções inovadoras para reforçar a atratividade e viabilidade do ciclismo, promovendo simultaneamente a eficiência e a sustentabilidade ambiental.

Não obstante o aumento verificado na adoção da bicicleta, os riscos enfrentados pelos ciclistas permanecem substanciais. A escassez de infraestruturas adequadas, a insegurança no tráfego, o congestionamento urbano e a poluição continuam a comprometer a prática do ciclismo, evidenciando a necessidade de analisar e de otimizar as condições de mobilidade urbana para este grupo.

Com o intuito de responder a estes desafios, a presente dissertação propõe o desenvolvimento de um sistema embebido que, recorrendo a tecnologias de Internet of Things (IoT), localização geoespacial em tempo real e técnicas de mineração de dados, permite calcular e apresentar um índice de risco unificado ao ciclista. Através de notificações imediatas, o sistema sinaliza a entrada em zonas potencialmente perigosas, promovendo uma percepção mais acentuada do ambiente urbano.

Este trabalho constitui, portanto, a implementação de uma solução inovadora, fundamentada num sistema embebido de baixo custo e operação autónoma (sem dependência de redes externas), com o propósito principal de otimizar a experiência e a segurança dos ciclistas em percursos urbanos.

Palavras chave: Sistemas Embebidos, Segurança de Ciclistas, Mobilidade Urbana, Avaliação de Risco, Cidades Inteligentes, Internet das Coisas

Abstract

In a context of sharp growth in the urban population and limited capacity of traditional public transport systems, there has been a significant increase in the use of bicycles as the main means of transportation. In response to this trend, smart cities have been looking for innovative solutions to enhance the attractiveness and viability of cycling, while promoting efficiency and environmental sustainability.

Despite the increase in bicycle adoption, the risks faced by cyclists remain substantial. The lack of adequate infrastructure, traffic insecurity, urban congestion, and pollution continue to compromise the practice of cycling, highlighting the need to analyze and optimize urban mobility conditions for this group.

In order to respond to these challenges, this dissertation proposes the development of an embedded system which, using Internet of Things (IoT) technologies, real-time geospatial location and data mining techniques, makes it possible to calculate and present a unified risk index to cyclists. Through immediate notifications, the system signals entry into potentially dangerous areas, promoting a better perception of the urban environment.

This work is therefore the implementation of an innovative solution, based on a low-cost embedded system and autonomous operation (without dependence on external networks), with the main purpose of optimizing the experience and safety of cyclists on urban routes.

Keywords: Embedded Systems, Cyclist Safety, Urban Mobility, Risk Assessment, Smart Cities, IoT

UN Sustainable Development Goals

Table 1: Sustainable Developments Goals (SDG) 11 and 13.

SGD	Target	Contribution	Performance Indicators and Metrics
11	11.2	Promotes safer urban mobility by encouraging the use of bicycles as a sustainable mode of transport and by reducing traffic-related risks.	Decrease in the number and severity of cycling related traffic accidents (Indicator 11.2.1).
13	13.2	Supports climate action through the promotion of low-emission transport, contributing to improved air quality and reduced reliance on fossil fuels.	Measurable reduction in CO ₂ from urban mobility emissions (Indicator 13.2.2).

Acknowledgements

The completion of this dissertation would not have been possible without the support and contributions of several individuals and institutions.

I would like to begin by thanking my supervisor, Professor Daniel Costa, for his guidance and encouragement throughout this project. His insights and technical feedback were instrumental in helping me stay focused and improve the quality of the work.

I thank the Faculty of Engineering of the University of Porto (FEUP) for providing the academic environment and infrastructure necessary to develop this project. Their support enabled the exploration of real-world mobility challenges and the application of embedded and geospatial technologies in a practical context.

On a more personal note, I would like to thank my family and close friends for their continuous support, patience, and motivation throughout the process. Their understanding made the long development hours and field testing phases considerably more manageable.

Finally, I hope that this work serves as a useful contribution to the fields of urban mobility and embedded systems and that it can inspire further developments toward safer and smarter cycling infrastructure in our cities.

Tiago Henrique Almeida de Sousa

“The best way to predict the future is to invent it.”

Alan Kay

Contents

1	Introduction	1
1.1	Context	1
1.2	Goals	2
1.3	Dissertation Structure	2
2	Background and Fundamental Concepts	4
2.1	Internet of Things	4
2.1.1	Architectures	6
2.1.2	Embedded platforms	7
2.2	Smart Cities	8
2.2.1	Smart Mobility	10
2.2.2	Smart cycling	11
2.2.3	Advancing sustainable cycling worldwide	12
2.3	Data Mining	13
2.4	Geospatial data and analysis	14
2.5	State of the Art	15
3	Proposed Approach	18
3.1	The System's Architecture	18
3.2	Pre-processing Unit	20
3.3	Cycling Unit	23
3.3.1	GPS Module	24
3.3.2	LED Bar Visualization Logic	25
3.3.3	Buzzer Module	26
4	Implementation and configuration	27
4.1	Pre-processing data sources	27
4.1.1	PoI-driven cycling risks based on CityZones data	28
4.1.2	Traffic-influenced cycling risks based on OpenStreetMap data	31
4.1.3	Final Normalization Process	33
4.2	Cycling Unit prototype	34
5	Discussion and Results	37
5.1	Experimental settings	37
5.2	Assessing cycling risks in Porto	40
5.2.1	Case Studies	41
5.2.2	Discussion	44
5.3	Performance issues	44

5.4	Practical Implications in Smart Cities	46
6	Conclusions	48
6.1	Main Findings and Achievements	49
6.2	Limitations and Future Work	49

List of Figures

2.1	Applications of IoT for supporting smart cities [49].	5
2.2	Most common IoT architectures.	6
2.3	Raspberry Pi 4 and Arduino UNO.	8
2.4	Smart Cycling Services.	11
2.5	Typical processing flow in Data Mining.	13
3.1	Flowchart of the system architecture, including the PU and CU.	19
3.2	Proposed Pre-processing Unit flowchart.	20
3.3	Example of a generated grid with a red point in the coordinates of each center. . .	23
3.4	Proposed Cycling Unit flowchart.	24
3.5	Typical \$GPGGA sentence retrieved from the GPS module.	25
4.1	Exported risk map from Porto, Portugal.	29
4.2	Example of output in a CSV file.	31
4.3	Intersections near to bus stops in Paranhos, Porto, Portugal.	33
4.4	Hardware chosen and connections between each component of the Cycling Unit. . .	35
4.5	3D-printed case containing the electronics and internal wiring.	36
4.6	Cycling Unit prototype mounted on the bicycle frame using the custom 3D printed case.	36
5.1	Real-time risk feedback displayed through the integrated LED bar.	41
5.2	Cycling Route 1.	42
5.3	Cycling Route 2.	42
5.4	Cycling Route 3.	43
5.5	Uni-T UT658C used to measure current during operation of the system.	46

List of Tables

1	Sustainable Developments Goals (SDG) 11 and 13.	iii
2.1	Characteristics of Smart City Categories.	9
2.2	Comparative analysis of related works in smart cycling and urban mobility. . . .	17
4.1	Hardware costs of the developed prototype, with supplier links.	36
5.1	Summary of cycling routes and corresponding risk data.	43
5.2	Average memory and CPU usage at different stages of system operation.	45
5.3	Timing metrics for risk evaluation during simulation.	45
5.4	Average current consumption at different stages of system operation.	45

List of Acronyms

AoI	Area of Interest
BSPs	Bicycle-Sharing Programs
CO ₂	Carbon Dioxide
CSV	Comma Separated Values
CU	Cycling Unit
DM	Data Mining
eHMI	External Human-Machine Interfaces
GIS	Geographic Information Systems
GPS	Global Positional Systems
IOT	Internet of Things
IEEE	Institute of Electrical and Electronics Engineers
LED	Light-Emitting Diode
ML	Machine Learning
MZ	Mitigation Zone
OSM	OpenStreetMap
PoC	Proof-of-Concept
PoI	Point of Interest
PU	Pre-processing Unit
R&D	Research and development
SoA	Service-oriented Architecture

Chapter 1

Introduction

1.1 Context

When observing the evolution of cities, it is easy to understand that the development of transport technology has significantly impacted the evolution of mobility patterns and has influenced changes in urban space [17]. However, cities are still very centered around cars as the main mode of transport, with the car being the most used transport in general. This trend has led cities to experience high levels of congestion and a significant carbon footprint resulting from vehicles powered by internal combustion engines [26].

Currently, the global community faces the pressing challenge of climate change. According to the World Bank, transportation is responsible for approximately 15% of global carbon dioxide (CO₂) emissions [38]. In response to the environmental and infrastructural challenges posed by urbanization, the concepts of smart and sustainable cities have gained increasing relevance. These initiatives, which are still evolving, aim primarily at reducing dependence on private vehicles and encouraging more sustainable forms of mobility. In this context, the integration of digital technologies, particularly big data analytics, into urban transport systems has emerged as a promising strategy to improve travel efficiency and overall mobility experiences [11].

In response to this evolving paradigm in urban mobility, bicycles and electric bicycles are progressively recognized as vital alternatives for short to medium distance urban mobility. Promotion of active mobility is supported by public policies that encourage the development of dedicated cycling infrastructure and the implementation of bicycle-sharing programs (BSPs). However, research by van Marsbergen et al. [48] reveals that not all BSPs function as intended. In certain instances, these services have been observed to replace public transit rather than complement it, potentially subverting the overarching objectives of modal integration. Although, when integrated into a well-coordinated urban mobility system, BSPs can play a complementary role by expanding transit coverage and reducing reliance on automobiles.

Despite these efforts, urban environments continue to present significant safety challenges for cyclists, including hazardous intersections, high traffic density, and limited access to emergency assistance. According to data from the European Commission, 57% of all cyclist deaths in Europe

occur on urban roads, compared to 40% for all combined road users [19]. This statistic underscores the disproportionate vulnerability of cyclists in urban areas and reinforces the urgency to implement targeted safety measures and technological interventions.

This dissertation addresses the need to improve cycling safety through the development of an embedded system that integrates Global Positioning Systems (GPS) and Geographic Information Systems (GIS) to provide real-time risk assessments and alerts. Integrating multi-domain urban map data with GPS tracking, the system generates a unified risk index based on factors such as proximity to hospitals, accident-prone areas, and overall safety of the route. These real-time alerts inform cyclists of potential hazards along their route, enabling them to make safer navigational choices.

By providing a technological solution to the issue of urban cycling safety, this system aims to reduce accidents and encourage more people to consider cycling as a viable and secure alternative to traditional forms of transport, thus contributing positively to sustainability, active mobility, and public health.

This project aligns with the goals of sustainable development and smart city promotion, while also demonstrating the technical feasibility of combining embedded systems with real-time geographic data, and contributes to the broader objective of designing safer and more accommodating cities for cyclists. The delivery of a cost-effective, scalable, and adaptable solution lays the foundation for future smart mobility initiatives that prioritize sustainability and urban resilience.

1.2 Goals

The primary goal of this research is to develop an embedded system aimed at enhancing the experience of urban cycling. To achieve this, the following specific objectives were defined:

- Leverage multi-domain urban data to support intelligent decision-making for safer cycling;
- Design and structure data formats capable of handling large geospatial maps;
- Implement a low-cost embedded solution that provides real-time alerts to cyclists;
- Evaluate the system through real-world case studies conducted in the city of Porto.

1.3 Dissertation Structure

In addition to the introduction, this dissertation contains five more chapters.

Chapter 2 focuses on the background and fundamental aspects of this research, providing an overview of key concepts and technologies such as the Internet of Things (IoT), smart cities, data mining, geospatial data and analysis, and the state of the art in the field. The chapter discusses the definitions, architectures and applications of IoT, the defining dimensions and advances in

smart cities, and the critical role of data mining and geospatial analysis in addressing urban challenges. The chapter lays the foundation necessary to understand the subsequent developments in the dissertation.

Chapter 3 presents the preliminary work and the proposed work plan. It introduces the architecture of the embedded system developed in this research, detailing its two main components: the Pre-processing and Cycling Units. The Pre-processing Unit is responsible for handling large geospatial datasets, while the Cycling Unit focuses on real-time risk assessment and visualization for cyclists. The chapter also describes the system's flexibility in processing geospatial files, its hardware schematic, and the flowcharts that guide its implementation, providing a clear roadmap for the research.

Chapter 4 reflects on the broader implications of this research and its potential to improve urban cycling safety, highlighting the integration of IoT and geospatial technologies as a crucial step towards achieving sustainable and safer mobility solutions for smart cities. It includes the data normalization pipeline, hardware integration, and software logic used to build the prototype. Both data sources and embedded components are described to illustrate the functional design.

Chapter 5 provides a discussion of the results and effectiveness of the system. It includes experimental setups, case studies conducted in the city of Porto, a comparative analysis of cycling routes with varying risk levels, and a performance assessment with respect to computational resources and power consumption.

Finally, Chapter 6 reflects on the contributions of the proposed system to urban mobility and smart city initiatives and concludes the dissertation by summarizing the main achievements and limitations of the project, while proposing directions for future work.

Chapter 2

Background and Fundamental Concepts

This chapter provides a comprehensive overview of the key concepts and technologies that underlie this work. It begins with an exploration of the Internet of Things, covering its definitions, characteristics, architectures, and the role of embedded platforms. IoT is highlighted as a cornerstone technology for modern smart systems, including its applications in smart cities. The discussion then extends to smart cities, focusing on their defining dimensions, applications, and specific advances in smart mobility and cycling.

In addition, the chapter explores the importance of Data Mining and geospatial analysis in addressing urban challenges, highlighting their role in enabling data-driven decision-making and improving urban planning. The integration of these technologies into smart cities illustrates their transformative potential to improve efficiency, sustainability, and safety in urban environments. Together, these fundamental aspects set the stage for understanding subsequent advances and applications in the context of bicycle safety and intelligent transportation systems.

2.1 Internet of Things

IoT is not a single technology, but a framework that integrates multiple technologies [2]. This section begins by exploring some definitions of IoT, then introduces three IoT architectures, and finally examines the embedded platforms.

The IEEE has provided two definitions of IoT [9]. The first refers to simple IoT networks, describing them as systems that connect uniquely identifiable devices, or "things", to the Internet. These things have sensing, actuation, and programmability capabilities. Through unique identification and sensing, information can be gathered about these devices and their states can be changed remotely at any time and from any location. The second definition applies to larger, more complex networks, such as those used in smart cities. It envisions the IoT as a self-configuring adaptive network that connects devices to the Internet using standardized communication protocols. These interconnected devices have a physical or virtual representation in the digital world,

with sensing, actuation, and programmability features. Each device is uniquely identifiable and carries information such as identity, status, location, or other relevant data. The system enables services with or without human interaction by leveraging unique identification, data capture, communication, and actuation. These services are accessible anytime, anywhere and for any purpose, with security considerations in place.

From these definitions, several key characteristics of the IoT can be highlighted:

1. IoT involves a system of interconnected "things".
2. These devices monitor or sense their environment.
3. Devices are connected to the Internet for communication.
4. Each device is uniquely identifiable.
5. The system processes, stores, or transmits data as needed.
6. Information is presented to one or more users.
7. The system responds to inputs from connected devices or users.

IoT has a significant impact on the daily lives of citizens, particularly in fields such as automation, manufacturing, logistics, and business, as well as in the development of intelligent transportation systems [4]. Its importance is especially pronounced in smart cities, where IoT facilitates interconnected infrastructures, enhanced public services, and refined governance models through e-Government initiatives [21]. Figure 2.1 illustrates various IoT applications that support smart cities.

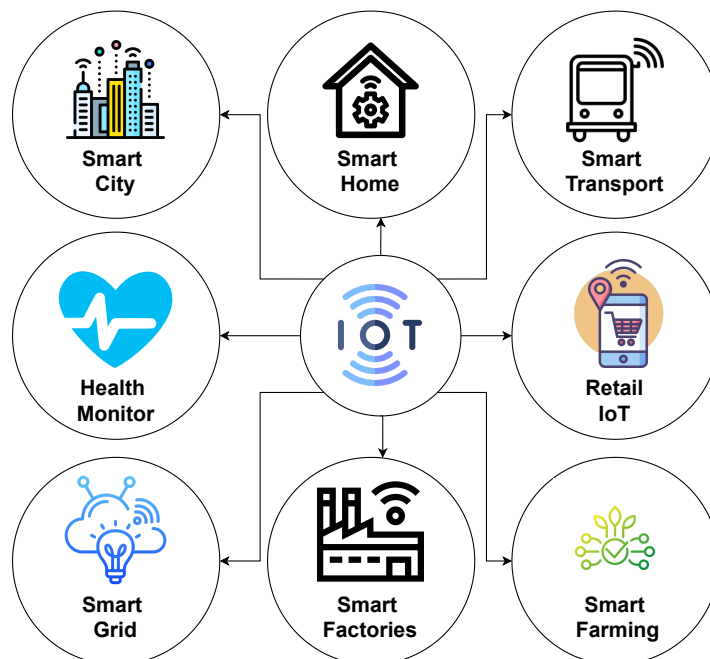


Figure 2.1: Applications of IoT for supporting smart cities [49].

2.1.1 Architectures

A technical architecture provides a structured framework that enables designers and developers to consider a system holistically while also breaking it down into manageable layers. Currently, a universally accepted reference architecture for the Internet of Things is lacking, and the development of one remains challenging despite numerous standardization efforts. This challenge stems from the inherent fragmentation of IoT applications, each characterized by distinct variables and design specifications [34].

Figure 2.2 illustrates some of the most commonly used IoT architectures.

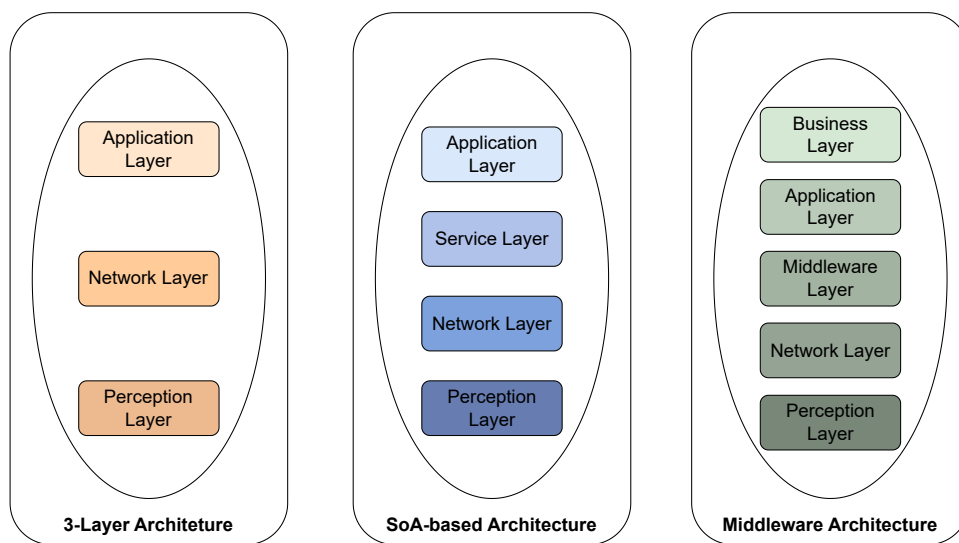


Figure 2.2: Most common IoT architectures.

In the three-layer architecture model that has gained wide acceptance, the perception layer functions at the physical level, engaging with the external environment through the collection and processing of data. Devices in this layer are frequently equipped with sensing and computational capabilities, rendering them "smart" from a technological standpoint and "intelligent" from a functional perspective. The network layer is responsible for transmitting the collected data to the application layer. It encompasses all the technologies and protocols necessary for communication and should not be confused with the network layer of the ISO/OSI model, which exclusively handles routing [47]. The application layer, located at the apex of the information technology infrastructure, is responsible for the processing, storage, and analysis of received data. This layer ultimately delivers services to end users.

In order to enhance interoperability and modularity, Service-Oriented Architecture (SoA) is frequently integrated into IoT frameworks. The SoA orchestrates the integration of services through the implementation of standardized interfaces and protocols, thereby fostering the reusability of software and hardware components. When incorporated into IoT systems, SoA introduces

an intermediate service layer between the network and application layers. This additional layer comprises the following elements:

- The process of *service discovery* is the procedure of locating available services;
- The *service composition* process integrates and orchestrates these elements;
- The establishment of trust and control mechanisms is imperative for effective *service management*;
- The implementation of *service interfaces* is instrumental in enabling seamless interactions across components;

Another notable model is the middleware-based or five-layer IoT architecture, which has gained traction in recent years due to its flexibility and ability to integrate legacy technologies [40]. This architecture introduces two additional layers to the original three: the middleware layer, which filters, aggregates, and manages access to data; and the business layer, which interprets processed data in the context of organizational goals or business intelligence. Middleware, in particular, functions as an abstraction layer that obscures technological intricacies and furnishes standardized interfaces, thereby enabling developers to concentrate on application logic without being encumbered by infrastructure compatibility concerns.

In response to the limitations of traditional architectural models Al-Awami, Mahfud Al-Aty, and Al-Najar [5] proposes a unified seven-layer IoT architecture. The model under consideration evaluates and structures IoT systems based on seven critical characteristics: scalability, interoperability, reliability, security, context awareness, real-time performance, and heterogeneity. By offering a comparative analysis of existing frameworks through these essential dimensions, their work reinforces the need for more robust and adaptive architectures, particularly in domains such as smart cities, where system diversity and performance are paramount.

2.1.2 Embedded platforms

A significant challenge in urban planning is the selection of devices capable of adapting to the complexity and scale of urban systems. The implementation of effective solutions that promote sustainable living for both people and the environment requires the careful integration of high-performance embedded systems. As mentioned in the seminal work [12], a multifaceted approach is imperative when selecting the most suitable open source platform for a particular Internet of Things application. These include cost, processing capabilities, programming flexibility, energy efficiency, physical footprint, and the availability of input or output interfaces.

The increasing popularity of the Internet of Things is attributable not only to technological advancements in the field, but also to an elevated demand for embedded hardware within urban environments. In this context, the role of embedded platforms within IoT infrastructures must be evaluated, particularly in terms of the design trade-offs involved. Devices such as the Raspberry Pi and Arduino, as illustrated in Figure 2.3, offer numerous advantages. These include support

for multiple programming languages, substantial computational capabilities, compact size, and affordability. The programmability and versatility of these systems makes them applicable to a wide range of domains, including environmental and urban monitoring tasks [12, 36].

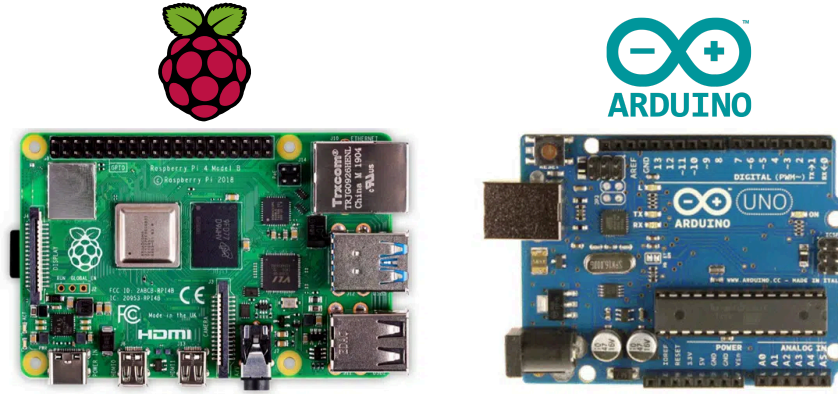


Figure 2.3: Raspberry Pi 4 and Arduino UNO.

In summary, embedded platforms are the hardware and software environments that are designed to run embedded systems, which are specialized computing devices dedicated to performing specific tasks or functions. These platforms provide the necessary resources to design, develop, and deploy embedded systems in various applications, from consumer electronics to industrial automation. A notable example is the integration of the Raspberry Pi Zero 2 W into an IoT-driven robotic surveillance system for smart environments, demonstrating its suitability for real-time monitoring and autonomous decision making in urban scenarios [27].

2.2 Smart Cities

Currently, a universal definition of what constitutes a smart city remains elusive. This phenomenon is primarily attributable to the fact that the concept under discussion varies depending on the specific characteristics, priorities, and development context of each city. However, according to recent reviews [28, 35], the existing literature is increasingly converging around six core dimensions that serve to structure and define the components of a smart city. A comprehensive overview of these dimensions is provided in Table 2.1.

Table 2.1: Characteristics of Smart City Categories.

Category	Characteristic
Smart economy	The emphasis is placed on the implementation of innovative and adaptable strategies to enhance the productivity and competitiveness of local economies. Key indicators include levels of entrepreneurialism (for example, number of startups), R&D investment, labor market adaptability, and international engagement.
Smart mobility	The focus of this initiative is to leverage advanced technologies and creative approaches to improve transportation and digital connectivity. Metrics generally encompass transport efficiency, adoption of sustainable modes, accessibility, and the development of infrastructure such as digital mobility services.
Smart environment	The promotion of sustainable environmental practices and efficient resource management is of paramount importance. Indicators encompass environmental quality, the use of renewable energy, green infrastructure, pollution control, and urban ecological assets such as green spaces and clean water.
Smart people	It focuses on human capital and civic engagement, where educated, creative, and technologically connected citizens are key drivers of progress. The following indicators are considered: education levels, digital inclusion, migration trends, and participation in community and public initiatives.
Smart living	The concept of quality of life is addressed through equitable provision of public services, including but not limited to housing, culture, entertainment, healthcare and personal safety. It is evident that the initiative has also produced positive results, as evidenced by the investments made to enhance the general well-being and life satisfaction of the residents.
Smart governance	This concept pertains to participatory and transparent decision-making processes that are supported by digital tools. Such tools include e-Governance platforms, open data initiatives, and citizen-driven budgeting. The initiative encompasses strategic urban development plans and inclusive governance practices.

A city can be regarded as "smart" when it effectively integrates social and human capital with traditional and digital infrastructure, comprising transport and communication systems, within a framework of sustainable development and inclusive governance. This integration aims to improve the quality of life of its inhabitants and promote their well-being.

Nevertheless, the development of truly smart and sustainable cities continues to face significant

challenges. As emphasized by Castro Paes et al. [7], these include technological integration, socioeconomic inequalities, institutional coordination, and the need for long-term environmental resilience.

In parallel, a separate set of pressing concerns pertains to cybersecurity and data governance. Many intelligent systems remain susceptible to emerging digital threats, which underscores persistent concerns about privacy and information security [32]. The widespread use of sensors and data collection technologies has led to critical inquiries on the storage, access, and ethical use of personal data [13]. Furthermore, effective implementation of smart city technologies frequently requires considerable investments in digital infrastructure and robust collaboration among government entities, private companies, and civil society.

2.2.1 Smart Mobility

The concept of smart mobility is gaining traction in the scholarly literature. Its definition is still lacking and differs depending on the context of developing, emerging, and developed economies. However, it is generally agreed that smart mobility aims to use technology to improve efficiency, safety, sustainability, and accessibility [38]. A critical aspect of intelligent transportation systems is the integration of maintenance strategies into the planning and design of transportation infrastructure, particularly with regard to cycling lanes and facilities [51].

Using data on road surface conditions, such as the presence of potholes or accumulated debris, urban planners can identify critical areas that need repair. This transformation in urban mobility is indicative of a change in value systems, where collaborative, sustainable, and data-driven solutions are prioritized over conventional transport planning models [1]. This data-driven preventive approach enhances the safety and comfort of cyclists while also contributing to the overall reliability of the transportation system. Leveraging statistical analyses to guide maintenance priorities allows cities to minimize accidents associated with poor infrastructure conditions.

In addition, the substantial availability of urban data facilitates the anticipation of future safety risks, allowing authorities to implement timely improvements and optimize the responsiveness of the cycling network. The integration of digital tools and predictive models has been shown to improve the operational quality of bicycle infrastructure while simultaneously strengthening the promotion of cycling as a viable and sustainable mode of transportation. A practical illustration of this phenomenon can be found in the work of Eguiluz et al. [18], in which an open source urban mobility web service was developed to integrate environmental quality metrics with bicycle mobility data. Consequently, the implementation of smart mobility strategies is imperative in the development of safe, effective, and environmentally conscious urban transportation ecosystems.

A recent case study in Amsterdam offers a pragmatic evaluation of the implementation of smart mobility through six key objectives: sustainability, efficiency, safety, accessibility, convenience, and cost-effectiveness [45]. The findings indicate that the system demonstrates high performance in terms of safety, convenience, and cost. However, the study also highlights the existing tensions between efficiency and safety, particularly in the context of low-speed zones that reduce travel speed while improving safety. The study emphasizes the inherent trade-offs in smart mobility

policies, underscoring the necessity for integrated strategies, such as intelligent traffic management and investment in active mobility, to achieve a balanced and inclusive urban mobility system.

2.2.2 Smart cycling

The concept of Smart Cycling has emerged as a new step to improve the experience of cyclists, their safety while moving, and the perception of bikes as an active element of the urban technological ecosystem [42]. This evolution reflects an increasing effort to bridge the gap between real-world cycling practices and the digital services that support them, as discussed by Carvalho and José [6].

Smart cycling represents a fundamental change in the role of cycling-related Information Technology, from its current supporting role, mostly focused on basic awareness and process optimization, to a leading role, where the new opportunities emerging from a fast-changing digital world become the key drivers for innovation and market competition in urban cycling.

In general, smart cycling is associated with some basic services, as expressed in Figure 2.4.

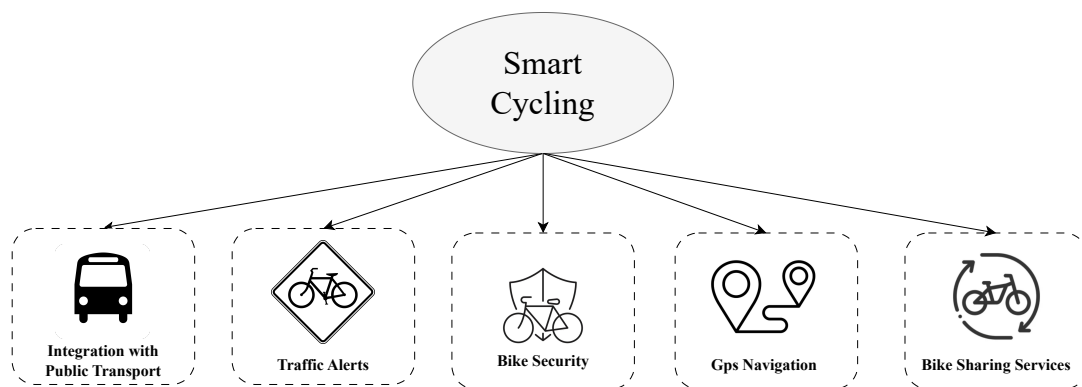


Figure 2.4: Smart Cycling Services.

Among recent advances in the realm of smart cycling, real-time safety systems have garnered significant attention due to their potential to reduce injuries and improve the efficiency of emergency response. For example, Munisamy et al. [39] proposed an embedded IoT-based system capable of detecting when a cyclist has fallen and immediately sending emergency notifications to relevant contacts or services. This solution exemplifies the evolution of smart cycling from a passive data collection model to an active safety intervention framework. It uses real-time data processing, embedded intelligence, and automated communication to ensure the protection of cyclists in urban environments.

A recent discourse analysis conducted in seven global cities (Helsinki, Montreal, Moscow, Tokyo, Dublin, Boston, and Vancouver) has highlighted that smart cycling is not only a technological solution, but also a sociopolitical strategy embedded in broader goals of urban development [24]. The study revealed that cities adopt divergent narratives and priorities regarding smart cycling, balancing technical innovation and social inclusion. This sociotechnical framework underscores the versatility of smart cycling initiatives in addressing a variety of urban challenges.

These challenges include, but are not limited to, sustainability, safety, civic engagement, and public health.

In parallel, the growing recognition of subjective safety perceptions has led to novel analytical approaches in smart cycling research. For example, Ye, Zhong, and Suel [53] proposed a three-level cycling safety indicator for London that combines injury and collision risk with street view imagery. Using interpretable machine learning, the researchers identified the characteristics of the roads most associated with perceived danger, such as intersections and low levels of greenery. Their findings emphasize that improving smart cycling requires not only technological interventions, but also a deep understanding of how cyclists interpret and respond emotionally to the environment.

2.2.3 Advancing sustainable cycling worldwide

Advancing sustainable cycling worldwide requires a deep understanding of how urban mobility can evolve to meet the demands of a more sustainable future. Cycling, as a mode of transportation, plays a vital role in reducing emissions, alleviating traffic congestion, and promoting healthier lifestyles. By examining successful examples of cycling initiatives globally, urban planners and policymakers can identify strategies that work and adapt them to their local contexts.

This review of the literature focuses on Copenhagen and Amsterdam as case studies to analyze the role of cycling and smart mobility initiatives [41]. Both cities are prominent players in the European and global field of smart cities, receiving awards and delegations from different parts of the world eager to learn about their smart technology, cycling planning, and culture.

Copenhagen has implemented a variety of smart cycling innovations, such as bicycle counters, cycling messaging boards, green waves, and smart data collection projects. Geospatial analysis is a crucial component in planning and implementing cycling routes. This analysis ensures that the routes are safe, accessible, and connected to key urban areas, contributing to the development of sustainable transportation infrastructure [52].

In addition, the city of Copenhagen has approached the parking problem by collecting data from different sources, most of which were related to parking and traffic situations. These historical data and real-time data have then been combined through algorithms, and machine learning makes it now possible to predict where to find vacant parking spots. The goal of the smart parking initiative is to minimize parking search time, which will reduce emissions and thus be another step toward becoming a CO₂ neutral city.

Compared to Copenhagen, smart cycling innovation in Amsterdam plays a somewhat less significant role in branding the city as a smart city. Amsterdam Smart City platform has a distinctive decentralized character as citizens, entrepreneurs, and public servants can register their initiatives on the website.

The priority in Amsterdam is a more efficient use of existing space and infrastructure, enabling more cycling and walking. Cycle policy is based on five pillars, such as road safety, infrastructure, parking, education, and promotion. The policy has resulted in a network of separate bike lanes, as well as a grid of protected cycle routes throughout the city, which ensures that planners take extra

predictions when redesigning a street within the network. Moreover, Amsterdam has implemented a cycling traffic system, informed by the combination of spatial design and traffic policy, where the minimum amount of time to be green is only six seconds.

In conclusion, the experiences of Copenhagen and Amsterdam show that promoting sustainable cycling globally requires both innovative strategies and a strong commitment to integrating cycling into urban infrastructure.

2.3 Data Mining

Data Mining (DM) is a robust analytical approach that is used to identify latent patterns, relationships, and trends within voluminous datasets, effectively transforming raw data into actionable insights. It encompasses a wide range of techniques and methodologies, drawing on fields such as statistics, Machine Learning (ML), and information theory to extract meaningful information from complex data sources.

In the context of big data, DM must extend across both front- and back-end processes. The data involved may originate from various repositories, including databases, data warehouses, online platforms, and various digital infrastructures. A distinguishing strength of DM is its capacity to process and analyze high-volume, high-dimensional datasets, which are often referred to as "big data."

A plethora of tools and platforms, both free and commercial, have emerged to support the development and deployment of DM techniques. As emphasized by Dol and Jawandhiya [16] in their seminal work, the advent of such tools has considerably reduced barriers to the implementation of DM in a variety of domains, extending beyond the confines of education. This development offers a promising prospect for the integration of DM in smart city contexts.

Using advanced computational techniques, DM can manage and interpret data that exceed the capabilities of traditional analysis tools. This quality makes it especially valuable in domains such as finance, healthcare, marketing, and social sciences, where data are generated continuously at scale.

Many smart cities are supporting big DM technologies, such as Smart Grids, which are expected to greatly boost intelligent communication technologies [14], and the prediction of electricity consumption, which is a vast field that attracts the attention of researchers increasingly [10]. In Figure 2.5, the whole process of Data Mining is highlighted.

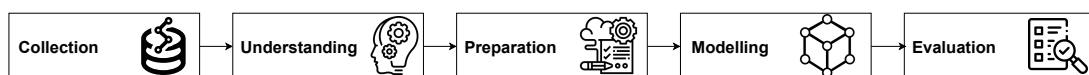


Figure 2.5: Typical processing flow in Data Mining.

The advent of the Internet of Things and big data has profoundly transformed the role of the Department of Management (DM) in smart cities. The widespread implementation of connected devices and sensors has led to the continuous generation of data, enabling real-time analysis and

expediting data-driven urban decision making. This integration improves the ability to address key urban challenges, such as traffic congestion, hazardous intersections, and cyclist safety.

However, the practical implementation of DM in urban contexts still faces notable challenges. The issues surrounding data privacy, security, and the integration of heterogeneous sources remain pressing concerns in the advancement of smart city technologies [13].

2.4 Geospatial data and analysis

Geospatial data are information that describes objects, events, or other features with a location on or near the surface of the Earth. Geospatial data typically combine location information and attribute information (the characteristics of the object, event, or phenomena concerned) with temporal information. The location provided may be static in the short term, like the location of a piece of equipment, or dynamic, such as a moving vehicle or pedestrian.

The spatial data produced by remote sensing technologies and IoT devices can be represented in various formats, including text, raster, or vector. These data types frequently manifest intricate spatial relationships that require the incorporation of topological characteristics such as distance, direction, and spatial structure [44]. A fundamental concept in this domain is spatial autocorrelation, defined as the tendency for nearby spatial values to exhibit greater similarity than distant values. This poses a significant challenge in the realm of geospatial data mining, as it deviates from the underlying assumptions of numerous conventional data mining techniques [44].

Beyond the realm of data mining, geospatial analysis employs Geographic Information Systems (GIS) and spatial datasets to interpret and visualize urban dynamics. In the context of smart mobility, geospatial tools are essential to identify mobility patterns, optimize transportation routes, and evaluate the effectiveness of different urban mobility strategies. A substantial corpus of research has demonstrated that geospatial analysis significantly enhances decision making in urban planning by revealing spatial dependencies, trends, and distributions [8].

The application of geospatial analysis to smart mobility and urban planning has garnered increasing interest in recent years. Notable examples include cities such as Barcelona and Copenhagen, which have successfully integrated geospatial intelligence into their mobility systems. These cities have implemented forward thinking strategies to improve traffic flow, air quality, and reduce urban congestion [52].

Recent advances have also highlighted the need for interoperable platforms that integrate both geospatial data and analytical models for online analysis and real-time decision support. A notable example is the development of centralized geospatial data and model centers, which enable scalable cloud-based spatial processing and facilitate integration with smart city infrastructures [15].

2.5 State of the Art

Cycling has emerged as a key mode of sustainable urban transportation, promoted as part of smart city initiatives to address climate change, urban congestion, and public health. However, according to the European Commission [19], cycling remains one of the most vulnerable modes of transportation, particularly in urban areas. As cities around the world strive for more intelligent and resilient infrastructure, the integration of IoT, embedded systems, and geospatial data has become essential to smart mobility efforts [7, 21, 51]. European examples, such as the Polish transition to smart cities, illustrate the growing policy and technological interest in sustainable mobility [28].

Smart city development increasingly relies on the synergy between technological platforms and urban governance, with particular attention to how mobility services are co-designed with citizens. In this context, smart cycling systems are gaining traction as a research focus and a practical solution to promote safe and efficient urban cycling [24, 42]. Cities such as Copenhagen and Amsterdam have become exemplary cases by adopting geospatial technologies and real-time monitoring to support route safety and cyclist infrastructure [41, 45, 52].

The IoT architecture is a pivotal component in the development of these systems. Several reviews have delved into the evolution, middleware design, and communication protocols of IoT [2, 4, 9, 34, 40, 47, 49]. Open-source urban mobility services, such as those proposed by Eguiluz et al. [18], illustrate the adaptability of IoT frameworks to incorporate data from sensors, GPS, and user feedback. This adaptability enables informed decision-making processes that guide both individual and collective mobility choices. However, challenges related to privacy, cybersecurity, and data governance persist in urban environments enhanced by IoT [13].

Geospatial analysis plays a fundamental role in these systems, allowing risk mapping, spatial clustering, and route optimization. The potential of spatial data mining for the analysis of road networks and infrastructure weaknesses is well documented [15, 44]. Tools such as CityZones [43] and methods for GIS-based urban zoning [50] highlight how embedded systems can draw on spatial information to generate real-time alerts and safety indexes. Studies such as Charnock, March, and Ribera-Fumaz [8] and Wolniak [52] further reflect on how smart mobility planning requires a nuanced understanding of the urban form and governance structures. The use of platforms such as OpenStreetMap to simulate road networks and urban features further strengthens the capabilities of geospatial data analysis [31, 46].

Several studies have also examined the application of predictive analytics in the context of smart cities. The application of data mining has been demonstrated in various fields, including the prediction of energy consumption [10], the anticipation of material use in smart grid systems [14], and the extraction of information from urban education [16]. The application of these techniques to the domain of cycling safety is a growing area of research, as evidenced by studies that examine visual perception of road safety using imagery and accident data [53].

Embedded systems represent a pivotal technological foundation, furnishing the hardware infrastructure essential for real-time analysis and actuation. The Raspberry Pi and Arduino platforms have achieved widespread adoption due to their cost effectiveness and adaptability [12, 36]. The

use of low-power embedded devices for road surface condition detection [22], has further underscored their aptitude for handling sophisticated AI tasks. Furthermore, novel contributions to the optimization of embedded systems and cybersecurity are emerging, particularly in the context of IoT integration [27, 32].

In the domain of mobility-focused research, the importance of GPS modules and route computation strategies, such as those based on Haversine distance calculations, has been thoroughly documented [3, 23]. These technologies are based on the foundational work in geodesy and satellite positioning [25]. Advancements in cloud-based smart city platforms [11, 35], and data management strategies, exemplified by CSV-based workflows, as outlined in [20], facilitate the development of location-aware, scalable systems.

From a service and social perspective, recent work has examined the role of participatory design [17, 38], e-Governance integration [21], and bicycle-sharing programs [48] in transforming urban mobility. Various studies discuss the disconnect between current mapping services and actual cycling needs, indicating the need for a better alignment between infrastructure and user experience [6]. Others have proposed cycling-specific HMI solutions and real-time emergency response systems [29, 39].

The smart mobility paradigm demands reflection on values and long-term impacts. The works of Ali-Vehmas and Casey [1] and International Energy Agency [26] assess the environmental and systemic implications of smart transportation systems, while Al-Awami, Mahfud Al-Aty, and Al-Najar [5] offers comparisons of IoT architecture models to guide sustainable development.

In summary, this dissertation builds on a rich body of research in IoT [2, 9, 49], embedded platforms [12, 36], geospatial analysis [15, 43], and smart mobility [41, 51]. It seeks to bridge existing gaps by proposing a bike-oriented system that combines real-time risk estimation, spatial intelligence, and embedded hardware into a unified low-cost solution. Through this contribution, the work aligns with broader efforts to improve sustainable urban mobility and safety in the context of smart cities.

To provide a more detailed illustration of the differences and commonalities among the most relevant contributions in the literature, Table 2.2 provides a comparative overview of selected works related to smart mobility and cycling safety. The comparison is centered on the objectives, technologies, contributions, and limitations of the selected works. This analysis underscores the manner in which previous studies have approached topics such as integration of urban mobility, fall detection, subjective safety perception, and geospatial analysis. However, none of these works combine lightweight embedded hardware with The present study proposes a real-time geospatial risk assessment specifically designed for cyclists. This discrepancy was the impetus for the development of the proposed system, which integrates GPS-based localization, geospatial risk data, and embedded feedback mechanisms to improve cyclist safety in urban environments.

Table 2.2: Comparative analysis of related works in smart cycling and urban mobility.

Author / Year	Focus of the Work	Technologies / Data	Contributions	Limitations
Eguiluz et al. (2022)	Urban mobility web service integrating environmental and cycling data.	IoT, Web services, Environmental data, Bicycle mobility.	Demonstrates integration of mobility and environmental quality.	Does not address cyclist safety.
Munisamy et al. (2024)	Real-time system for fallen cyclist detection and emergency notification.	IoT, Embedded sensors, Communication systems.	Automatic notification in case of accidents.	Acts only after accident occurs.
Ye, Zhong, and Suel (2024)	Cycling safety indicator using street view imagery and accident data.	Machine Learning, Street view imagery, Accident data.	Combines subjective safety perception with accident data.	Not applicable in real time.
Charnock, March, and Ribera-Fumaz (2021)	Geospatial analysis of urban planning models (Barcelona case).	GIS, Spatial clustering, Policy analysis.	Reveals spatial dependencies in mobility planning.	Does not implement embedded prototype.
Wolniak (2023)	Smart mobility analysis in Copenhagen and Barcelona.	GIS, Smart mobility indicators.	Provides comparative study of sustainable mobility strategies.	Case study, not cyclist-specific.
Carvalho and José (2022)	Gap between cycling practices and mapping services.	IoT, Mapping platforms, Smart cycling services.	Highlights challenges in integrating digital cycling services.	Focuses on usability, not risk.
van Marsbergen et al. (2022)	Role of bicycle sharing programs in urban transit.	Public policy, Transport data.	Shows influence of bike-sharing on mobility systems.	Does not directly address cyclist safety.
Proposed Work (2025)	Embedded system for cyclist risk assessment.	IoT, GPS, CityZones [43], OSM.	Real-time risk evaluation with LED and buzzer feedback.	Validated only in Porto case study.

As summarized in Table 2.2, existing contributions have addressed specific aspects of smart mobility and cycling safety, but have lacked an integrated real-time embedded solution. The proposed work fills this gap by combining geospatial risk data, GPS-based localization, and immediate feedback through lightweight hardware.

Chapter 3

Proposed Approach

This chapter presents the methodological steps and fundamental elements of the proposed approach, which is centered on an embedded system that can be attached to bicycles. The main objective of such a system is to monitor and assess cycling conditions in real time based on geospatial risk indicators. Since it was developed to provide immediate feedback to cyclists through a visual interface and audio alerts, quick warning "notifications" are at the core of the proposed approach, potentially improving safety during urban travel.

The architecture of the proposed system is divided into two modules: the Pre-processing Unit (PU) and the Cycling Unit (CU). The PU, a program that can be executed on a regular computer, is responsible for the management of large-scale geospatial datasets, with a particular emphasis on the organization and indexing of information to facilitate efficient data processing and retrieval. This process involves the generation of a spatial grid, risk normalization, and the export of structured files that are compatible with lightweight embedded hardware. On the other hand, the CU, which is the actual code that will be executed on an embedded device attached to bikes, functions in real time and was designed to operate on a Raspberry Pi Zero 2 board, due to its reduced size, low energy consumption, affordable price, and reasonable processing and memory capabilities. This module integrates GPS input, compares the current location with previously processed risk zones, and provides the user with a real-time risk level using a 10-segment LED bar and a buzzer. In fact, this unit was optimized for low memory usage, fast response times, and independence from external connectivity, achieving one of the goals of this dissertation.

The subsequent sections of this chapter provide a detailed exposition of the design rationale underpinning each module. This exposition covers input/output requirements, flowcharts, hardware configuration, and implementation strategies.

3.1 The System's Architecture

The proposed model integrates multiple components to handle geospatial data containing risk zone information and to present the corresponding risk levels in real time. The system architecture is

structured into two main processes, each of which is described in detail in the following subsections.

The initial component, designated as the Pre-processing Unit, is responsible for managing data from any geospatial file type. In this stage, each coordinate is assigned a unique index, which enables efficient storage and rapid lookup during real-time operation.

The second component, designated as the Cycling Unit, involves the acquisition of the cyclist's current GPS position. This capability enables the system to determine the pertinent urban area and provide context-sensitive safety information to the user. The cyclist is provided with a unified risk index, displayed in green LED color, while the emitted audible alerts play a crucial role in ensuring the safety of the cyclist on the road.

Figure 3.1 illustrates the approach proposed for the implementation of the embedded system.

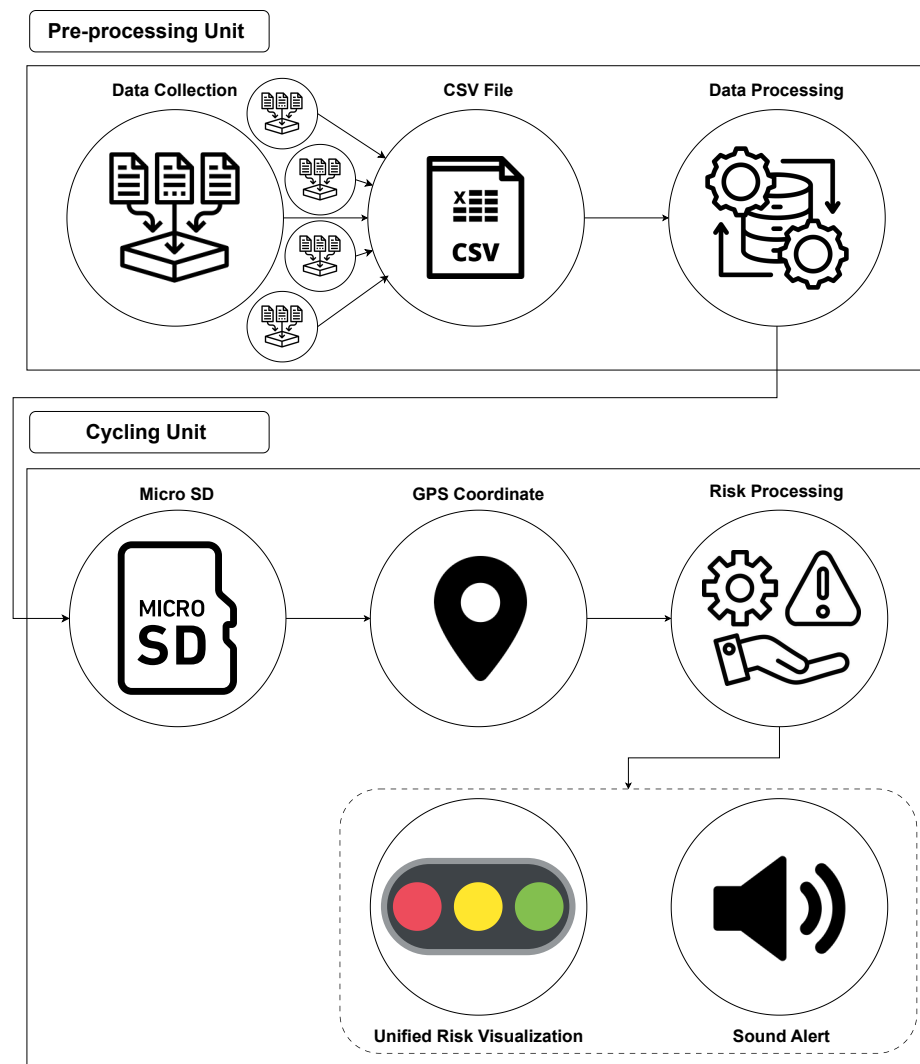


Figure 3.1: Flowchart of the system architecture, including the PU and CU.

This system is characterized by its flexibility, allowing the use of any geospatial CSV file as input. Regardless of the specific structure of the provided file, the system can efficiently process it, assign unique indices to the coordinates, generate the unified risk index, and emit the necessary audible alerts when the cyclist enters the risk areas. This adaptability ensures that the system works consistently in different contexts and locations, significantly improving cyclist safety by providing clear, real-time information on the level of risk along their route.

3.2 Pre-processing Unit

The PU serves as the foundation for managing geospatial data input in the system. Its primary function is to process any geospatial CSV file provided as input. This unit is designed to be highly flexible, allowing the manipulation of files with different structures and formats. The main goal of this unit is to organize and index the data in a way that facilitates efficient and rapid searches during real-time operation.

Figure 3.2 illustrates the workflow of the PU. The procedure commences with the delineation of the geographical boundary box, which is defined by two input coordinates. This is followed by the specification of the desired grid cell size. Consequently, a standard spatial grid is generated and each cell is assigned a unique identifier.

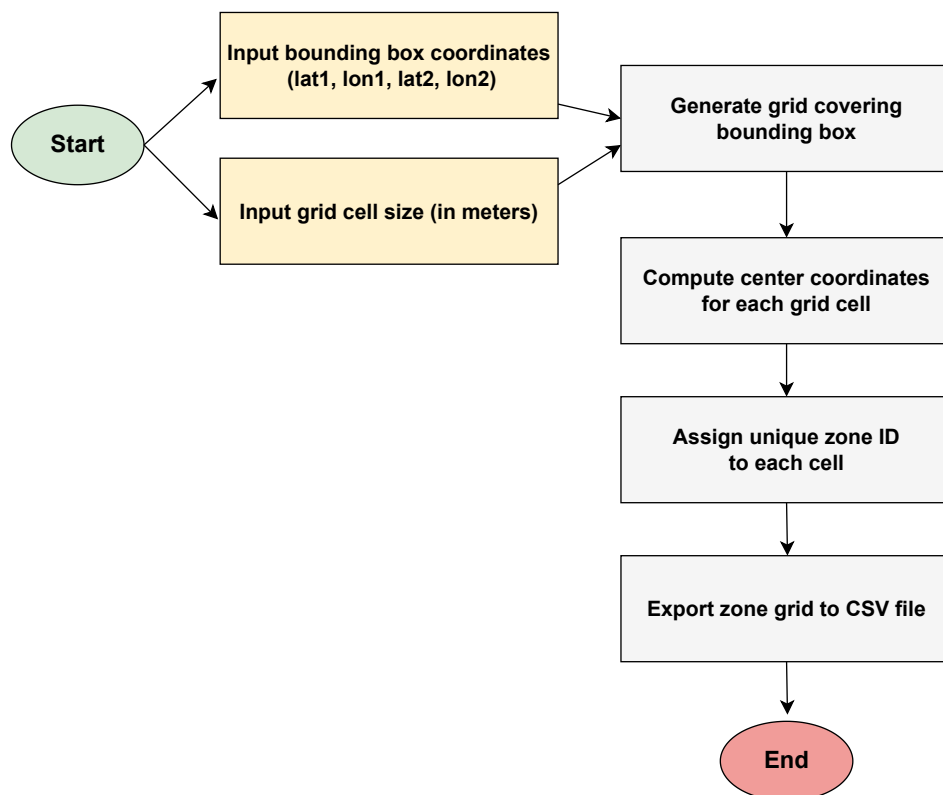


Figure 3.2: Proposed Pre-processing Unit flowchart.

The initial phase of the PU involves the division of the geographical area of interest into a regular grid of square zones. This spatial decomposition facilitates efficient risk attribution and look-up operations during runtime. In support of this assertion, a range of geospatial utility procedures are employed.

A fundamental technique employed is the Haversine approximation, which facilitates the calculation of the great-circle distance between two points on the Earth's surface by leveraging their latitude and longitude. This approach is widely used in geospatial analysis due to its simplicity and precision over short distances [23].

The algorithm 1 delimits the procedure used to compute the distance in meters from the real world between GPS coordinates.

Algorithm 1: Compute geodesic distance between two GPS coordinates (Haversine distance).

Input: Two coordinate pairs (ϕ_1, λ_1) and (ϕ_2, λ_2)

Output: Distance in meters between the two points

- 1 Use the geodesic formula to compute the great-circle distance between the two coordinates;
 - 2 Return the distance in meters;
-

In order to facilitate the conversion of spatial dimensions from meters to degrees, a procedure is required in order to define grid steps in geographic coordinates.

The algorithm 2 offers a pragmatic estimate for the conversion of distances in meters to degrees in geographic coordinates, which accounts for the curvature of the Earth relative to latitude. This method is based on the assumption that one degree of latitude is approximately equal to 111,320 meters, while the longitudinal distance represented by one degree varies with the cosine of the latitude. This approach is frequently used in geospatial applications to delineate uniform grid steps across geographic regions [25, 50].

Algorithm 2: Convert distance in meters to geographic degrees.

Input: Distance in meters, reference latitude

Output: Approximate degree values for that distance in latitude and longitude

- 1 Compute latitude degree step as $\frac{\text{meters}}{111320}$;
 - 2 Compute longitude degree step as $\frac{\text{meters}}{111320 \cdot \cos(\text{latitude in radians})}$;
 - 3 Return both steps;
-

Utilizing these two fundamental components, the entire grid is generated through the iterative creation of non-overlapping square zones within the delineated bounding box. The process is described in Algorithm 3, which generates the polygonal representation of each cell and the coordinates of its center.

The grid centers thus determined are then saved to a CSV file for future reference and usage in the downstream processing, as illustrated in Algorithm 4.

Algorithm 3: Divide geographic area into equally sized square zones

Input: Bounding box coordinates: (Ax, Ay) (top-left), (Bx, By) (bottom-right), grid size in meters**Output:** List of square polygons and their geographic centers

- 1 Compute total width and height of the bounding area in meters using `haversine_distance`;
 - 2 Compute number of columns (Zw) and rows (Zh) based on grid size;
 - 3 Convert grid size to degrees using `meters_to_degrees`;
 - 4 **foreach** *cell in grid* **do**
 - 5 Compute corner coordinates of the square;
 - 6 **if** *square exceeds bounding box* **then**
 - 7 Skip to next iteration;
 - 8 **end**
 - 9 Create a polygon from corner coordinates;
 - 10 Compute geographic center of the square;
 - 11 Append square and center to the output lists;
 - 12 **end**
 - 13 Return the list of square polygons and their centers;
-

Algorithm 4: Save grid zone centers to CSV.

Input: List of grid cell centers, filename**Output:** CSV file with latitude, longitude, and zone width

- 1 Create a data table with columns: Latitude, Longitude, Zone Width;
 - 2 Write the data to CSV file using the specified filename;
 - 3 Print confirmation message;
-

The resulting CSV file, which is the product of this PU, contains the geographical coordinates of the centers of each grid zone, along with their corresponding size. This zonal structure serves as a foundational framework for the subsequent stages of the system, during which risk levels are assigned.

Figure 3.3 illustrates a grid generated from an area near the Faculty of Engineering of the University of Porto.

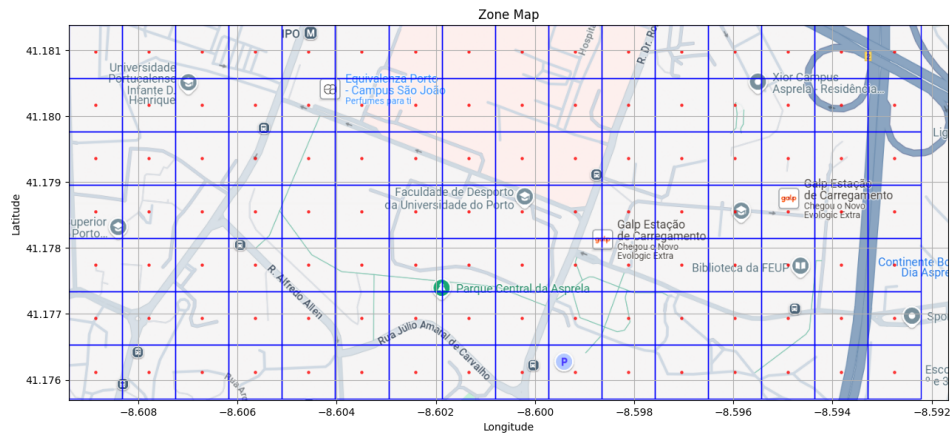


Figure 3.3: Example of a generated grid with a red point in the coordinates of each center.

The use of this grid facilitates the future association of each incoming GPS coordinate with a designated zone. This capability enables localized analysis and efficient mapping of risk information.

3.3 Cycling Unit

The function of the Cycling Unit is to deliver real-time risk assessments and immediate feedback to the cyclist during the movement. The design of this system prioritizes safety, responsiveness, and non-intrusiveness. That is, it ensures that the cyclist remains informed without compromising attention or control.

This module operates by continuously acquiring the current GPS coordinates of the cyclist and comparing them against a previously generated geospatial grid produced by the PU. For each new location fix, the system identifies the closest grid cell (the center of the nearest zone) and retrieves the corresponding assigned risk level.

In order to facilitate the effective transmission of this information to the user, the CU integrates the following hardware components:

- **Led Indicators:** A 10-segment Led bar is employed to visually represent the current risk level. The quantity of illuminated segments is directly proportional to the normalized risk value within the range of [1, 100].
- **Auditory Alerts:** An electric buzzer is an electronic device that emits warning sounds when a significant increase in risk is detected. For example, it would be activated if the user transitioned from a low-risk zone to a high-risk zone. This mechanism functions as a redundant alert system, a particularly valuable asset when visual attention is otherwise occupied.

- **Battery and Power Management:** The system is powered by a battery and includes a physical on/off switch for manual activation. Power consumption is minimized through the implementation of efficient software loops and sensor management.

The real-time logic is implemented as a continuous monitoring loop. At each iteration, the system retrieves the most recent GPS fix, identifies the nearest grid zone, retrieves the associated risk level from the local CSV file, and updates the visual and auditory output accordingly.

As illustrated in Figure 3.4, the CU employs an internal logic that is optimized for low latency performance and capable of autonomously operating independently of external systems.

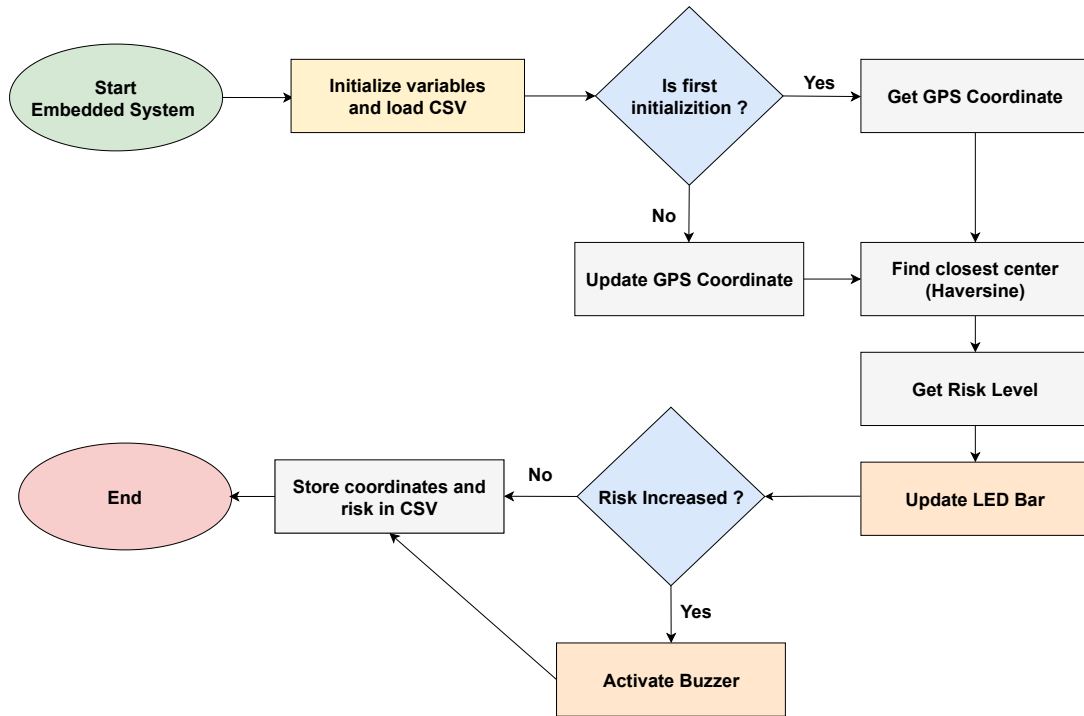


Figure 3.4: Proposed Cycling Unit flowchart.

The integration of geospatial awareness, modular feedback mechanisms, and efficient processing within the CU has been shown to improve situational awareness and promote safer urban mobility for cyclists.

3.3.1 GPS Module

The CU is initiated by acquiring the current geographic position through the GPS subsystem, which is implemented using the Grove GPS module (Air530). The GPS module is periodically polled for an interval of 1 second, thereby ensuring that the positional tracking is kept up to date. A similar approach to GPS-based localization is described by Ariffin [3], who highlights the use of NMEA-compatible modules and real-time tracking strategies for embedded systems, particularly in low-power scenarios.

The data retrieved from the GPS module generally align with the NMEA 0183 standard, a protocol defined by the National Marine Electronics Association (NMEA) that establishes a communication protocol and message structure that is widely adopted by navigation equipment. This format consists of comma-separated ASCII sentences, where each field represents specific navigational data such as time, latitude, longitude, altitude, fix status, and the number of satellites in view.

The most pertinent sentence for positional data acquisition is the one prefixed with \$GPGGA, known as the Global Positioning System Fix Data sentence.

```
$GPGGA,163447.000,4123.4567,N,00836.1234,W,1,08,1.0,100.0,M,50.0,M,,*68
```

Figure 3.5: Typical \$GPGGA sentence retrieved from the GPS module.

Figure 3.5 shows a typical message provided by the GPS module. The following itemization details the interpretation of each field in this sentence.

- Timestamp (HHMMSS.SSS): 163447.000 corresponds to 16:34:47 UTC;
- Latitude and Direction: 4123.4567,N corresponds to 41°23'45.67" North;
- Longitude and Direction: 00836.1234,W corresponds to 8°36'12.34" West;
- Fix Quality: 1 indicates a valid GPS fix;
- Number of Satellites: 8 satellites were used in the position fix;
- Horizontal Dilution of Precision (HDOP): 1.0, indicating high accuracy;
- Altitude: 100.0 M above mean sea level.

3.3.2 LED Bar Visualization Logic

The system incorporates a 10-segment LED bar that is designed to provide cyclists with intuitive visual feedback on the level of environmental risk in real time. The management of this feedback is entrusted to a dedicated Python control module, which determines the number of segments to be illuminated based on the calculated risk score.

To ensure proportionality, the function `writeRiskLevel(level)` maps a normalized risk score (1 to 100) to a corresponding number of illuminated segments. Each segment of the graph represents a 10-point increment, and the underlying logic can be outlined as follows.

$$\text{num_leds} = \left\lfloor \frac{\text{level} - 1}{10} \right\rfloor + 1 \quad (3.1)$$

where `level` represents the current risk score.

The activation of the LED segments is initiated sequentially from the base to the apex, ensuring that an increased number of illuminated segments corresponds to a greater perceived risk. This

approach presents a method for alerting the cyclist that is easily visible and screen-free, thus ensuring that the attention of the cyclist and mobility are not hindered.

3.3.3 Buzzer Module

In addition to the visual LED bar, the prototype incorporates an auditory feedback mechanism implemented through a buzzer. The objective of this module is to provide immediate auditory alerts to the cyclist when a change in the detected risk level occurs. This is of particular importance in situations where the attention of the rider may not be on the visual indicators or during low-light conditions.

The buzzer is controlled through a dedicated GPIO pin and activated for a brief duration to ensure that alerts are both noticeable and not disruptive. The integration of this technology ensures that cyclists receive prompt notifications regarding changes in urban risk conditions, thereby enhancing their general awareness of the route.

The implementation of the buzzer logic is encapsulated in a simple class, as demonstrated in the following code:

Listing 3.1: Example of a buzzer activation class.

```
from gpiozero import Buzzer
import time

class BuzzerHandler:
    def __init__(self, buzzer_pin):
        self.buzzer = Buzzer(buzzer_pin)

    def playBuzzer(self):
        self.buzzer.on()
        time.sleep(1)
        self.buzzer.off()
```

The module is designed to start only in the event of a change in risk level. The system has been designed to provide cyclists with timely and accessible warnings by offering complementary auditory feedback, regardless of their current focus or ambient conditions.

Chapter 4

Implementation and configuration

This chapter details the implementation of a Proof-of-Concept (PoC) for the system architecture that was introduced in the preceding chapter. The objective is to validate the feasibility and functional integration of the two core modules, the Pre-processing Unit and the Cycling Unit, in a realistic embedded setup.

The Pre-processing Unit was developed and tested in a desktop computing environment using Python-based scripts and geospatial data files. This component is responsible for preparing and structuring the data prior to deployment on the embedded device, and thus is expected to be executed on a regular computer platform before the target bicycle is ready to be ridden.

The Cycling Unit was implemented on a Raspberry Pi Zero 2 W, integrating a GPS module, a 10-segment LED bar, and a piezoelectric buzzer. The program logic was implemented in Python, with the primary objective of minimizing resource consumption and ensuring responsive behavior.

The implementation process underscores the flexibility of the system and its ability to integrate various risk indicators derived from multiple data sources, including CityZones and OpenStreetMap. The software design is also notable for its use of a modular approach, a feature that facilitates real-time execution on a low-power embedded platform. Furthermore, this chapter presents the physical integration of all components in a compact bicycle-mounted prototype designed for field testing.

This chapter also delineates the configuration of each module in detail, including hardware connections, software configuration, and the integration of peripheral components.

4.1 Pre-processing data sources

To enable a real-time assessment of cycling risks, it is essential to preprocess the data sources that are used to create the final risk map. This involves transforming raw geospatial datasets into a structured, consistent format that is suitable for execution in resource-constrained embedded systems.

The Pre-processing Unit was carried out in a standard desktop environment with Python scripts. Two types of data were processed: the risk level of urban zones, derived from the proximity

of the infrastructure using the CityZones methodology, and the structural risk of traffic intersections, calculated using OpenStreetMap data.

It is important to note that although our solution was designed to process any number of data sources when computing a unified perception of cycling risks, we decided to incorporate in our experiments only two realistic data sources to demonstrate the practical adoption of our solution in real scenarios. Nevertheless, other data sources could be easily employed since the proper proposed normalization processing is applied.

4.1.1 PoI-driven cycling risks based on CityZones data

CityZones is an online geospatial analysis instrument developed to assess urban risk zones for emergency response planning in the context of smart cities [43]. The platform utilizes real-world geographic data to evaluate the effectiveness of emergency response capabilities within a specified urban region. This objective is achieved by delineating Areas of Interest (AoIs) and subsequently categorizing them into three distinct risk categories: low (green), medium (yellow), and high (red) based on their proximity to critical infrastructures such as hospitals, police stations, and fire departments.

It is possible for users to define an Area of Interest and segment it into uniform mitigation zones (MZs). These zones are typically square areas of fixed dimensions. Subsequently, the tool calculates the relative risk for each mitigation zone based on the spatial distribution of the emergency resources.

The resulting output is presented as a heatmap, which visually encodes the level of risk in the specified area. The heatmap can be exported in standard formats, such as Comma-Separated Values (CSV), for integration into other analytical workflows or systems.

The risk heatmap shown in Figure 4.1 presents the output of the CityZones tool when applied to the city of Porto, Portugal. The generation of this map involves a two-step geospatial data processing pipeline, which transforms raw risk level (RL) values into an interpretable zone-based heatmap.

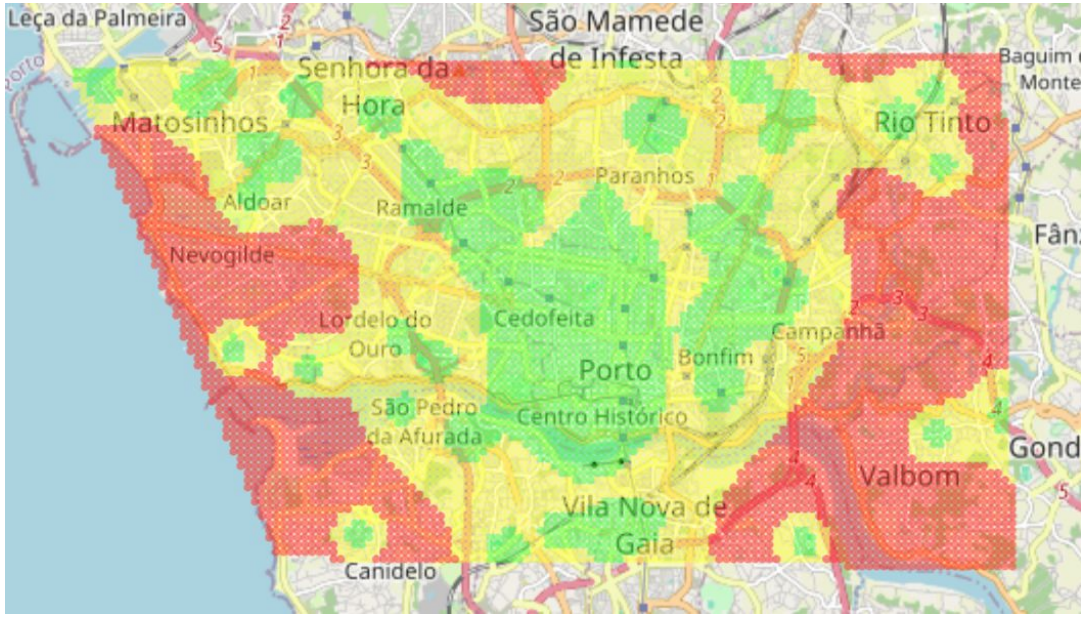


Figure 4.1: Exported risk map from Porto, Portugal.

In the initial stage, the RL values obtained for different coordinates within the AoI are normalized to a constant scale ranging from 1 to 100. This approach is designed to ensure uniform representation of the risk intensity in various regions or datasets. The normalization procedure is formally described in Algorithm 5.

The RL rescaled is calculated for each point based on the minimum and maximum values in the dataset, and a linear transformation is applied to map all values to the interval [1, 100]. This approach is designed to prevent distortion in color gradients due to outlier values and facilitate comparisons between zones.

Algorithm 5: Normalize RL values to a fixed range between 1 and 100.

Input: DataFrame *df* containing a column with RL values

Output: DataFrame with normalized RL values in the range [1, 100]

```

1 Determine the minimum (old_min) and maximum (old_max) RL values from the
  column;
2 foreach value v in the RL column do
3   if old_max > old_min then
4     | Compute normalized value as:  $\text{round}(1 + \frac{(v - \text{old\_min}) \cdot 99}{\text{old\_max} - \text{old\_min}})$ 
5   else
6     | Assign value 1
7   end
8 end
9 Update the RL column with the normalized values;
10 Return the modified DataFrame;

```

Subsequent to the normalization of the RL values, the second stage entails the aggregation of the data in a spatial domain. Specifically, each risk point is assigned to the closest mitigation zone (MZ), which functions as a spatial bin. These zones are predefined as a uniform grid over the AoI. The aggregation process is delineated in Algorithm 6, where the RL of each point is assigned to the corresponding zone.

Following this, the tool calculates the average risk level (RL) for each zone, generating a single representative risk value per mitigation zone. In the absence of associated points, a default RL value of 1 is assigned to ensure full coverage.

Algorithm 6: Assign zones to points and compute average RL per zone.

Input: `points_file`: CSV file with points and RL values; `zones_file`: CSV file with zone coordinates (Latitude, Longitude)

Output: CSV file with zone coordinates and average RL per zone

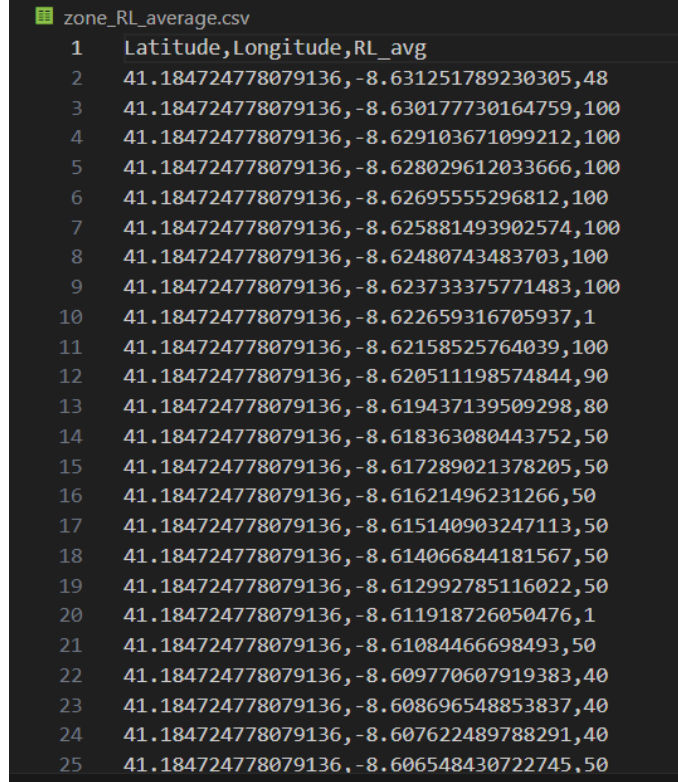
```

1 Load zones_df from zones_file;
2 Load points_df from points_file;
3 Extract zone coordinates: zone_coords = [Latitude, Longitude];
4 Extract point coordinates: point_coords = [lat, lon];
5 Build KD-tree: tree = cKDTree(zone_coords);
6 Query nearest zone for each point: dists, indices =
  tree.query(point_coords);
7 Assign zone_id to each point using indices;
8 Initialize RL_sum and RL_count columns in zones_df with 0;
9 foreach row in points_df do
10   Let z = row[zone_id];
11   Let rl = row[RL];
12   Increment zones_df[RL_sum][z] by rl;
13   Increment zones_df[RL_count][z] by 1;
14 end
15 foreach row r in zones_df do
16   if r[RL_count] > 0 then
17     Compute RL_avg = max(1, round(RL_sum / RL_count));
18   else
19     Set RL_avg = 1;
20   end
21 end
22 Save zones_df with columns Latitude, Longitude, and RL_avg to
  output_file;
```

The result of this pipeline is a structured dataset in which each mitigation zone is associated with its geographic center and an average normalized RL value. This output is then stored in a

CSV file that contains the latitude, longitude, and a computed risk level for each designated zone.

The resulting dataset is prepared for integration into external systems or further processing in urban analytics pipelines. This enables consistent and scalable use across various applications, including emergency planning, resource allocation, and mobility analysis. Figure 4.2 illustrates the output of zone assignment with the normalized RL in a CSV file.



1	Latitude,Longitude,RL_avg
2	41.184724778079136,-8.631251789230305,48
3	41.184724778079136,-8.630177730164759,100
4	41.184724778079136,-8.629103671099212,100
5	41.184724778079136,-8.628029612033666,100
6	41.184724778079136,-8.62695555296812,100
7	41.184724778079136,-8.625881493902574,100
8	41.184724778079136,-8.62480743483703,100
9	41.184724778079136,-8.623733375771483,100
10	41.184724778079136,-8.622659316705937,1
11	41.184724778079136,-8.62158525764039,100
12	41.184724778079136,-8.620511198574844,90
13	41.184724778079136,-8.619437139509298,80
14	41.184724778079136,-8.618363080443752,50
15	41.184724778079136,-8.617289021378205,50
16	41.184724778079136,-8.61621496231266,50
17	41.184724778079136,-8.615140903247113,50
18	41.184724778079136,-8.614066844181567,50
19	41.184724778079136,-8.612992785116022,50
20	41.184724778079136,-8.611918726050476,1
21	41.184724778079136,-8.61084466698493,50
22	41.184724778079136,-8.609770607919383,40
23	41.184724778079136,-8.608696548853837,40
24	41.184724778079136,-8.607622489788291,40
25	41.184724778079136,-8.606548430722745,50

Figure 4.2: Example of output in a CSV file.

In summary, the CityZones tool applies a combination of normalization (Algorithm 5) and spatial aggregation (Algorithm 6) to transform heterogeneous geospatial risk data into a structured dataset.

4.1.2 Traffic-influenced cycling risks based on OpenStreetMap data

OpenStreetMap (OSM) is a collaborative project that provides freely accessible geographic data, including detailed representations of road networks, intersections, and transportation infrastructure. Although OSM does not provide real-time traffic information, it serves as a reliable spatial basis for estimating traffic-influenced risks based on the static configuration of the urban infrastructure.

For instance, Saha et al. [46] demonstrated the generation of synthetic distribution feeders using OSM data, and Li [31] explored the use of OSM for procedural modeling of urban night scenes. These studies highlight the versatility of OSM in supporting geospatial applications, ranging from technical infrastructure planning to virtual urban environments.

In this thesis, the OSM data were used to extract the road network of a specified urban area and to identify intersections that may pose increased risk to cyclists. The risk index assigned to each intersection is not based on dynamic traffic flow but rather on static structural characteristics. Specifically, two factors were considered in the analysis: the complexity of the intersection, determined by its degree (the number of connected streets) and its proximity to public transportation, namely, the presence of a bus stop within 50 meters of the intersection.

The methodology is initiated by filtering the road network to retain only intersections with a degree greater than two, as these junctions are typically more complex and potentially more hazardous. This process is described in Algorithm 7, in which each selected intersection is evaluated according to the following:

- $I = \max(\text{degree} - 2, 0)$: a structural risk component;
- $B = 1$ if a bus stop exists within 50 meters of the intersection, otherwise $B = 0$;
- Risk Index = $I + B$: the total estimated risk score assigned to the intersection.

Algorithm 7: Compute intersection risk index using structural and spatial criteria

Input: Set of intersections and bus stop geometries, distance threshold

Output: List of intersections with risk index and coordinates

```

1 foreach intersection do
2   Retrieve the intersection's geometry;
3   Compute the intersection's degree in the road graph;
4   Let  $I = \max(\text{degree} - 2, 0)$ ;
5   Find the nearest bus stop to the intersection;
6   if distance between intersection and nearest bus stop < threshold then
7     | Set  $B = 1$ 
8   else
9     | Set  $B = 0$ 
10  end
11  Compute Risk =  $I + B$ ;
12  Extract latitude and longitude from intersection geometry;
13  Save {latitude, longitude, degree,  $I$ ,  $B$ , Risk} to results list;
14 end
15 Export results to a CSV file;
```

The final result is a structured dataset in which each row corresponds to a road intersection and includes its geographic coordinates, the calculated degree, the structural score I , the proximity factor B , and the total risk index. This dataset is exported as a CSV file and can be used independently or integrated into larger urban risk models.

As illustrated in Figure 4.3, the intersections in Paranhos, Porto, that connect more than two streets and are located within 50 meters of the nearest bus stop are of particular interest. Although

these data do not represent real-time traffic conditions, they reflect estimated structural risk based on static infrastructure characteristics available in OpenStreetMap.

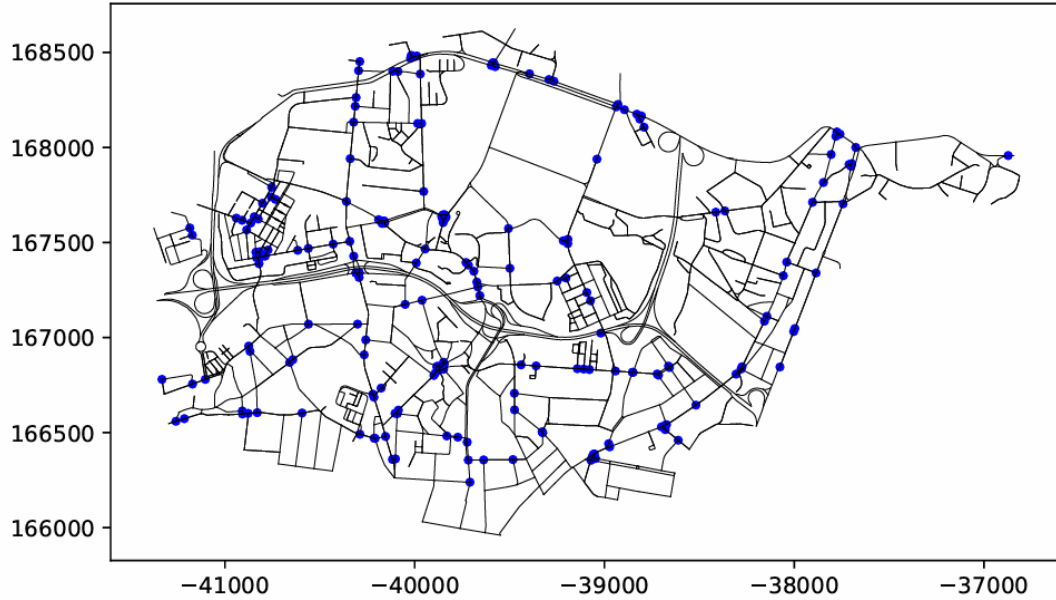


Figure 4.3: Intersections near to bus stops in Paranhos, Porto, Portugal.

Normalization of the resulting risk values is imperative to ensure consistency with other system components. The method defined in Algorithm 5 undertakes this task by rescaling the scores to a standard range between 1 and 100. This process maintains uniformity and enables direct comparability between all modules.

4.1.3 Final Normalization Process

The final step of the risk processing workflow is presented in this section, where the risk values derived from two independent methodologies are combined to generate a unified geospatial dataset. Specifically, the approach merges the following:

- The zone-based risk computed using the CityZones data, which evaluates risk based on proximity to emergency infrastructure within spatial mitigation zones (stored as `RL_avg`);
- The intersection-based risk calculated using OpenStreetMap data, which assigns a risk score to each road intersection based on topological complexity and proximity to public transportation stops (stored as `risco_norm`).

The process begins by normalizing the intersection risk values to a common scale from 1 to 100 using the same normalization procedure described in Algorithm 5.

The two datasets are converted to geospatial formats and re-projected into a metric coordinate system (EPSG:3763), ensuring accurate distance-based spatial operations. A nearest-neighbor

spatial join is performed to associate each zone with its closest intersection, as long as the distance does not exceed 50 meters.

If an intersection is found within this distance, the final risk value is calculated as an average between the zone risk and the normalized intersection risk. If no intersection is found, the zone risk is used as the default value. This procedure is detailed in Algorithm 8.

Algorithm 8: Combine zone-based (CityZones) and intersection-based (OSMnx) risk to compute final risk score

Input: Zone file with `RL_avg`, intersection file with `risco_norm`, threshold distance

Output: CSV file with final risk per location

```

1 Normalize intersection risk values to range [1, 100], store in risco_norm;
2 Convert both datasets to GeoDataFrames using latitude and longitude;
3 Reproject both datasets to metric CRS (e.g., EPSG:3763);
4 foreach zone do
5     Find nearest intersection within 50 meters;
6     if intersection is found then
7         | Compute final risk as average:  $\text{risco} = \text{round}\left(\frac{\text{RL\_avg} + \text{risco\_norm}}{2}\right)$ 
8     else
9         | Set final risk to zone value: risco = RL_avg
10    end
11    Save {Latitude, Longitude, risco} to results
12 end
13 Export results to CSV file;
```

The final output is a structured CSV file in which each row contains the geographic coordinates of the assessed location along with the final risk level.

This consolidated dataset captures both the broader contextual risk associated with infrastructure availability and the local structural risk associated with traffic intersections, making it suitable for visualization, decision-making, and integration into smart city systems. The adoption of the Comma Separated Values (CSV) format aligns with the prevailing practices in data processing workflows for embedded systems and analytical pipelines [20].

Designed for real-time applications, this file will serve as input to the embedded system that provides cyclists with continuous risk feedback through visual and auditory signals.

4.2 Cycling Unit prototype

The primary objective of the developed cycling unit is to assess the risk level of the area where the cyclist is located in real time and to provide clear alerts without affecting the cycling experience. The system is designed to remain energy efficient, compact, and financially accessible, both in terms of acquisition and maintenance.

In order to achieve these objectives, hardware design places high priority on the utilization of small, low-power, and cost-effective components. These components are integrated into a custom 3D printed enclosure. This enclosure not only protects the electronics, but also ensures easy mounting and stability during cycling. The architecture integrates a Global Positioning System (GPS) module for geolocation, a visual Light Emitting Diode (LED) bar to indicate the risk level, and a buzzer to generate auditory alerts.

The effectiveness of visual and auditory alerts in urban cycling scenarios has been supported in prior studies that involved the interaction of cyclists with automated vehicles. In particular, Kaß et al. [29] demonstrated that well-designed external human-machine interfaces (eHMI) significantly improved cyclist behavior and decision making when faced with ambiguous vehicle maneuvers. Although the present system does not involve vehicle interaction, the principle of providing timely and unambiguous feedback to cyclists remains crucial to enhancing safety in complex traffic environments.

Figure 4.4 illustrates the architecture of the system and the connections between each of its components.

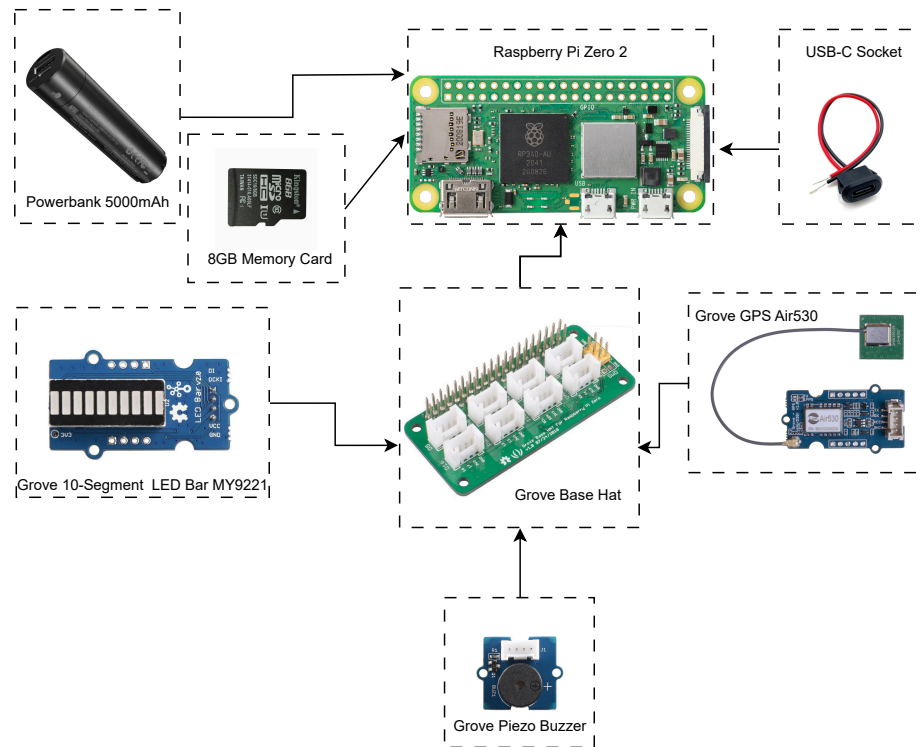


Figure 4.4: Hardware chosen and connections between each component of the Cycling Unit.

As demonstrated in Table 4.1, a comprehensive cost analysis of the hardware components is provided. This selection ensures that the prototype remains financially accessible to a broad range of consumers while also maintaining minimal energy consumption and adaptability.

Table 4.1: Hardware costs of the developed prototype, with supplier links.

Component	Cost (€)	Qty	Supplier Link
Raspberry Pi Zero 2 W	17.99	1	Mauser Portugal
Grove GPS (Air530)	11.90	1	Mauser Portugal
Grove LED Bar (MY9221)	6.10	1	Mauser Portugal
Grove Buzzer	1.80	1	Mauser Portugal
Grove Hat	7.50	1	Mauser Portugal
USB-C Socket	4.45	1	Mauser Portugal
USB-C Cable	5.00	1	Mauser Portugal
Rocket switch	1.36	1	Mauser Portugal
3D Case	10.00	2	Custom Printed
Total	76.10 €	10	

The physical integration of the system was achieved through the implementation of a custom designed 3D-printed case, which accommodates all electronic components and connectors. As illustrated in Figure 4.5, the assembled case and its internal connections are shown. Figure 4.6 presents the case mounted on a bicycle frame, highlighting its compact form factor and unobtrusive placement during operation.



Figure 4.5: 3D-printed case containing the electronics and internal wiring.



Figure 4.6: Cycling Unit prototype mounted on the bicycle frame using the custom 3D printed case.

Chapter 5

Discussion and Results

This chapter presents the validation results and performance analysis of the proposed system, based on real-world testing scenarios conducted in the city of Porto. The objective of this study is twofold: first, to evaluate the functional effectiveness of the prototype in detecting and signaling environmental risk, and second, to evaluate its computational efficiency on hardware with limited resources.

The chapter is divided into three sections. The first section outlines the experimental setup, including the configuration of sensors and risk mapping strategies. The second section evaluates the behavior of the system during active cycling through several case studies, comparing different urban routes in terms of risk exposure. The third section analyzes performance indicators such as memory consumption, CPU load, and energy usage.

This chapter offers a comprehensive evaluation of the system's reliability, responsiveness, and suitability for urban cycling applications. The chapter presents a balanced integration of quantitative and qualitative evidence, providing a multifaceted perspective on the system's performance and potential applications.

5.1 Experimental settings

The prototype was deployed on a Raspberry Pi Zero 2 W, which served as the embedded computing platform responsible for coordinating GPS, risk calculation, LED feedback, and buzzer activation. The selection of this hardware was driven by its compact size, minimal energy requirements, and adequate computing power to execute Python-based risk detection algorithms in real time.

The integrated system comprises three primary components:

- The Grove GPS Air530 module is used to determine the location of the cyclist at one-second intervals.
- The MY9221 LED bar, which is composed of ten segments, is employed to visually represent the current level of cycling risk;

- A Grove piezoelectric buzzer has been developed to provide an audible warning to cyclists whenever they enter a riskier zone than the one they were in previously.

The system operates continuously while the cyclist is in motion, with updates occurring at a rate of once per second (although this could easily be configured to a different time frequency). At each iteration, the current GPS coordinates are captured and compared against a preloaded CSV file containing geospatial risk data. The closest known risk point is determined using the Haversine distance, and the corresponding risk level is extracted. The system utilizes this value to regulate the LED bar and, if necessary, to trigger the buzzer.

Algorithm 9 presents the core logic executed on the Raspberry Pi in a looped cycle. The system loads the precomputed risk values and continuously monitors the location of the cyclist. The primary control flow guarantees real-time responsiveness to changes in the environment of the cyclist.

Algorithm 9: Main control loop of the system.

```

1 Load risk data from CSV;
2 previous_risk ← 0;
3 while system is active do
4   (lat, lon) ← Get GPS coordinate;
5   current_risk ← FindClosestRisk(lat, lon);
6   UpdateLEDBar(current_risk);
7   if current_risk > previous_risk then
8     ActivateBuzzer();
9   end
10  previous_risk ← current_risk;
11  Wait 1 second;
12 end

```

Algorithm 10 details the subroutine `FindClosestRisk()`, which identifies the risk value of the zone closest to the current position of the cyclist. It iterates through all preloaded coordinates and returns the highest risk value found among the nearest matches. This conservative approach ensures that transitions to risky areas are never underestimated.

Algorithm 10: FindClosestRisk(lat, lon).

```

1 min_distance  $\leftarrow \infty$ ;
2 max_risk  $\leftarrow 0$ ;
3 foreach (point_lat, point_lon, risk) in CSV do
4   d  $\leftarrow$  HaversineDistance(lat, lon, point_lat, point_lon);
5   if d < min_distance then
6     min_distance  $\leftarrow$  d;
7     max_risk  $\leftarrow$  risk;
8   else
9     if d = min_distance then
10      max_risk  $\leftarrow$  max(max_risk, risk);
11    end
12  end
13 end
14 return max_risk;

```

Algorithm 11 delineates the logic employed to translate a risk value into the corresponding LED pattern. The 10-segment bar illuminates in proportion to the risk index, which is normalized between 1 and 100. Finally, Algorithm 12 controls the operation of the buzzer, which is triggered when the current risk level exceeds the previous. This logic establishes an audible signal to alert the cyclist to an increase in environmental risk.

Algorithm 11: UpdateLEDBar(risk).

```

1 risk  $\leftarrow$  Clamp(risk, 1, 100);
2 num_leds  $\leftarrow$  floor((risk - 1) / 10) + 1;
3 led_pattern  $\leftarrow$  Bitmask with num_leds LEDs ON;
4 Set LED bar to led_pattern;

```

Algorithm 12: ActivateBuzzer()

```

1 Turn buzzer ON;
2 Wait 1 second;
3 Turn buzzer OFF;

```

The implementation of these algorithms was modularized in Python, with each hardware component encapsulated in a dedicated script. The primary control loop is contained in the file named `main.py`, while the acquisition of GPS signals, the LED control, and the buzzer activation are managed through the files `GPS.py`, `Ledlib.py`, and `Buzzer.py`, respectively. The utilization of a modular structure facilitates the establishment of a clear abstraction and the simplification of maintenance procedures for the codebase.

This software architecture was designed to ensure that all hardware modules interacted seamlessly with minimal delay. During field experiments, the system demonstrated a consistent 1-second refresh cycle, which facilitated the timely update of visual and auditory alerts in response to GPS-based risk evaluation.

5.2 Assessing cycling risks in Porto

To validate the functionality and real world applicability of the proposed system, bicycle field tests were conducted in different zones of the city of Porto. The primary objective of this study was to assess the response of the system to georeferenced risk information during periods of active mobility. These tests confirmed that the GPS module acquired coordinates at a rate of once per second and that the embedded algorithms updated the LED and buzzer outputs reliably in real time.

During the experimental phase, the cyclist navigated predefined routes that traversed areas with varying risk levels. Upon reaching a new location, the system evaluated its proximity to known risk zones and provided visual feedback through the LED bar. In the event that the risk level increased compared to the previous reading, an audible alert was initiated.

The subsequent output of the code (Listing 5.1) demonstrates a simulated execution of the system using authentic coordinates derived from one of the cycling sessions. In this simulation, the fluctuation in risk levels is manifested through the activation of the buzzer and the LED bar.

Listing 5.1: System output during the simulation of coordinates with associated risk levels. The buzzer is activated whenever a higher risk level is detected compared to the previous reading.

```
>>> Run main.py

Simulando coordenada: 41.237496, -8.612972
Risco atual: 50
Risco aumentou! Ativando buzzer...

Simulando coordenada: 41.233016, -8.608568
Risco atual: 40

Simulando coordenada: 41.236338, -8.617299
Risco atual: 50
Risco aumentou! Ativando buzzer...

Simulando coordenada: 41.237832, -8.617297
Risco atual: 50

Simulando coordenada: 41.235529, -8.616097
Risco atual: 50

Simulando coordenada: 41.232121, -8.608806
Risco atual: 40
```

As illustrated in Figure 5.1, two moments from the same cycling session are shown, with varying levels of risk indicated by the number of LED segments illuminated on the device. In both images, the LED feedback system provides an immediate and visible context for the cyclist about their environment.



(a) Device operating at risk level of 50.



(b) Device operating at risk level of 40.

Figure 5.1: Real-time risk feedback displayed through the integrated LED bar.

5.2.1 Case Studies

In order to evaluate the efficacy of the system in diverse urban contexts, three different cycling routes were selected throughout the city of Porto. Each route was chosen to exemplify a distinct risk profile and a range of traffic conditions.

As illustrated in Figures 5.2, 5.3 and 5.4, three different urban routes were selected for the real-world validation of the system. Each route is characterized by a distinct combination of traffic density, infrastructure quality, and environmental complexity, thus offering a diverse set of cycling conditions for analysis.



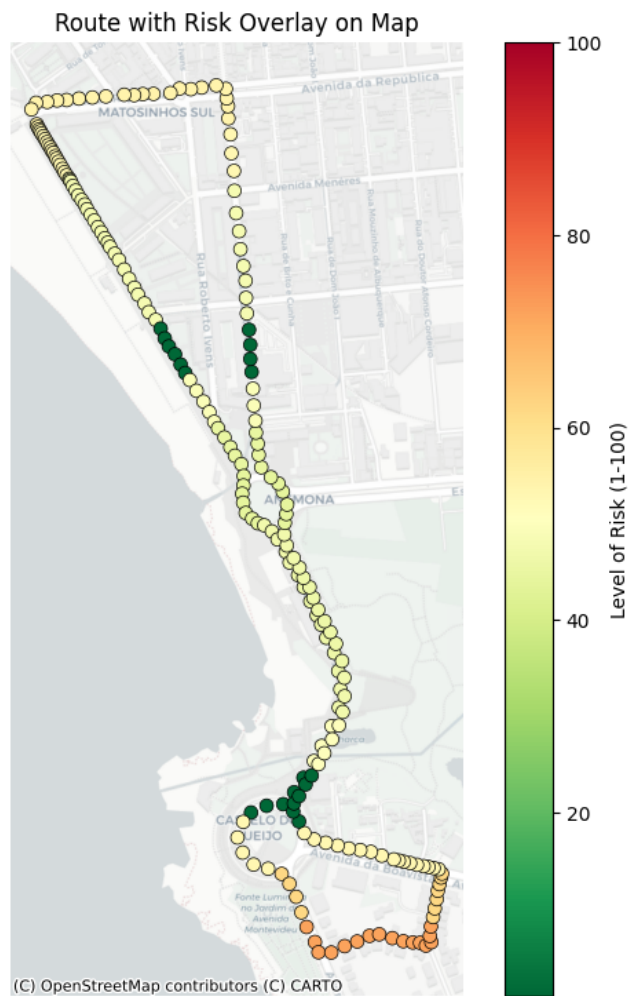


Figure 5.4: Cycling Route 3.

Table 5.1: Summary of cycling routes and corresponding risk data.

Route	Distance (km)	Risk Zones Detected	Average Risk	Max Risk
Route 1	1.84	102	22.14	50
Route 2	3.54	274	37.12	50
Route 3	4.21	213	53.04	72

Table 5.1 presents a summary of the salient characteristics and risk metrics derived from each of the three routes selected for analysis. The selection of these routes was meticulous, with the objective of exemplifying the heterogeneity in traffic and infrastructure conditions in diverse urban contexts within the city of Porto.

5.2.2 Discussion

The route delineated in Figure 5.2, although it is the shortest (1.84 km), crosses more than 100 zones, indicating a route with a high degree of segmentation and frequent environmental changes. Despite this observation, the average risk level remains at its lowest point (22.14), suggesting that the majority of these segments correspond to relatively safe areas, characterized by calmer streets, reduced vehicular traffic, or the presence of cycling infrastructure.

Figure 5.3 introduces a scenario of medium length 3.54 km, covering 274 detected zones and an average risk value of 37.12, which is considered moderate. It is plausible that this route traverses a diverse array of environments, encompassing moderately busy thoroughfares and critical intersections. The increased number of detected zones may signify a heightened frequency of transitions between low- and high-risk areas, thereby posing a challenge to the system's capacity to provide consistent feedback.

The third route, illustrated in Figure 5.4, was selected to represent a complex and high-risk cycling scenario. This route was identified as having the highest average risk (53.04) and the highest single risk value encountered (72). This route traverses denser, high-traffic areas and includes several complex intersections or proximity to risk-inducing infrastructures, such as bus stops and multilane roads. The system was able to detect and inform the cyclist about these high-risk conditions in real time.

These three case studies demonstrate the ability of the system to discriminate between different levels of environmental risk and adapt its feedback accordingly. By analyzing the average and maximum risk values in a variety of urban settings, it becomes clear that the system can support cyclists in identifying safer routes, avoiding hazardous areas, and making informed decisions during navigation. This comparative analysis also validates the robustness of the risk model used, which successfully captures the diversity and dynamic nature of real-world cycling conditions.

5.3 Performance issues

In the context of embedded systems designed for real-time decision-making, particularly in mobility and safety applications, performance constraints assume a pivotal role in ensuring the viability of the system. The proposed system was developed to run on the Raspberry Pi Zero 2, a low power, resource-constrained platform. This requires careful consideration of memory usage, processing load, and energy consumption.

This section is devoted to the evaluation of the performance of the system in these dimensions, with a particular emphasis on its ability to function efficiently without compromising responsiveness. A variety of metrics were measured at different stages of operation in order to assess the system's scalability under realistic usage conditions. These metrics included average memory usage, CPU load, and electrical current draw. The results are presented in Table 5.2 and Table 5.4, and are discussed in detail below.

Table 5.2: Average memory and CPU usage at different stages of system operation.

Stage of Operation	Memory Used (MB)	CPU Used (%)
Before operation	189.54	1.50
During operation	195.45	9.23
After operation	184.20	2.80

As illustrated in Table 5.2, the mean usage of memory and CPU is documented in three critical phases of system execution: before, during, and after active operation. As anticipated, the most substantial resource consumption occurs during the active operation phase, where both GPS readings and real-time risk evaluations are performed. The utilization of the central processing unit (CPU) increases from an unloaded state of approximately 1.5% to 9.23% during operation, reflecting the computational demands of real-time processing, coordinate matching, and alert management. Memory usage also increases marginally, from 189.54 MB to 195.45 MB, which remains well within the limitations of the Raspberry Pi Zero 2 with 512 MB RAM. Subsequent to the operation, resource consumption stabilizes, thereby confirming that the system appropriately releases memory and processing resources upon the termination of execution.

Table 5.3: Timing metrics for risk evaluation during simulation.

Metric	Measured Time (s)
CSV load time	0.0752
Average time of risk calculation per coordinate	0.1929

In addition to resource usage, execution timing was also measured in order to assess the real-time responsiveness of the system. As shown in Table 5.3, the time required to load the risk dataset from a CSV file was approximately 0.0752 seconds, ensuring fast initialization even in resource-constrained environments. During operation, the system maintained an average time for risk calculation of 0.1929 seconds per coordinate, covering all processing stages including matching, retrieval, and alert triggering. These values validate the suitability of the system for real-time applications in dynamic urban cycling environments.

Table 5.4: Average current consumption at different stages of system operation.

Stage of Operation	Current (A)
Before operation	0.39
During operation	0.55
After operation	0.46

This increase in computational workload is reflected in the electrical consumption profile of the system, presented in Table 5.4. The system exhibited a clear increase in current draw during the active monitoring phase. Before operation, the system consumed an average of 0.39 A. During operation, when GPS data acquisition, real-time risk assessment, and alert mechanisms were

activated, the current increased to an average of 0.55 A. After the operation was completed, the current consumption decreased and stabilized at approximately 0.46 A, indicating the return to a low-power state. Measurements were performed using a USB-C voltage and current tester with memory functionality (Uni-T UT658C), which allowed precise tracking of power draw. Figure 5.5 illustrates an example of a measurement performed for the present evaluation.



Figure 5.5: Uni-T UT658C used to measure current during operation of the system.

These results confirm that the additional load introduced by the GPS module, the LED bar, and the buzzer remains within acceptable limits. Despite the increased activity, the system remains highly energy efficient and suitable for use with portable power supplies. This reinforces its applicability for real-time urban cycling safety applications.

5.4 Practical Implications in Smart Cities

The proposed embedded system contributes to smart city development by improving urban planning, bicycle safety, and sustainable mobility. As urban populations continue to expand and cities seek to implement efficient and livable environments, cycling is a critical element in the development of sustainable transport strategies. However, the widespread adoption of this technology is contingent not only on the development of adequate infrastructure but also on the implementation of technological innovations that guarantee safety, usability, and public trust. Smart cities, as defined in the literature, encompass multiple dimensions, including governance, mobility, environment, and digital infrastructure [30].

A significant contribution of the system is its ability to detect and notify risks in real time. Using IoT technologies and geospatial analytics, the device provides cyclists with immediate awareness of hazardous zones, including high-risk intersections or areas with limited access to emergency services. This situational awareness empowers users to make informed routing decisions, reinforcing a proactive approach to the safety of urban mobility.

From a governance perspective, the anonymized and aggregated data collected by the system offers valuable insights for city planners and policymakers. For example, heat maps of frequently flagged danger zones can support data-driven decisions such as prioritizing infrastructure upgrades or enforcing targeted traffic regulations. This initiative aims to address the disparity between the individual user experience and urban planning, thus facilitating a responsive, evidence-based feedback loop. This feedback loop is a fundamental component of smart city ecosystems.

Furthermore, the low-cost, autonomous design of the system ensures accessibility and scalability, especially in cities with limited budgets for large-scale infrastructure projects. Unlike solutions reliant on constant wireless connectivity or cloud services, this system performs all decision making locally, maintaining full functionality. This resilience makes it particularly suitable for implementation in developing or resource-constrained urban contexts.

It is imperative to note that the system has been meticulously engineered to accommodate a multitude of input databases, irrespective of their provenance or configuration. This modular and data-agnostic architecture facilitates the seamless integration of heterogeneous datasets, including accident records, geospatial indicators, traffic data, and user-generated feedback. Consequently, the solution is in line with the tenets of data-driven smart cities, facilitating continuous enhancement of the decision-making process and real-time updates without necessitating structural modifications to the system. This scalability guarantees that the platform will remain relevant and adaptable as urban data ecosystems undergo evolution.

Beyond its technical capabilities, the solution fosters a cultural shift toward data-aware and safety-oriented cycling. Incorporating risk intelligence into daily commuting routines has been shown to foster user confidence and promote the adoption of active transportation modes. This phenomenon can potentially instigate a virtuous cycle characterized by reduced congestion, reduced emissions, and improved public health. These objectives are of paramount importance in the context of global smart city initiatives. Recent empirical studies further underscore the need for balanced smart city development strategies that prioritize not only technological advancement, but also social equity, privacy, and environmental sustainability [33].

However, the successful implementation of such technologies is contingent on broader institutional capacities. As Mills et al. [37] have shown in their work, sustainable smart city innovation is contingent not only on technical efficiency, but also on the ability to leverage technology, foster innovation, and manage diverse stakeholder interests. Their research emphasizes the necessity of developing these capacities with practical frameworks that account for institutional complexity and real-world constraints. The incorporation of this perspective serves to reinforce the relevance of the proposed system as both a technological solution and a catalyst for building systemic capacity in smart city contexts.

Chapter 6

Conclusions

This dissertation has addressed the pressing challenges facing urban cyclists, particularly those related to navigating hazardous areas, maintaining real-time awareness of environmental risk, and aligning cyclist safety with the broader vision of smart cities. To address these issues, a modular and cost-effective embedded system was proposed, integrating multi-domain urban datasets, geospatial analysis, and IoT-based technologies to provide cyclists with actionable risk feedback.

The investigation carried out as part of this project has confirmed that urban environments present significant risks to cyclists, due to factors such as inadequate infrastructure, unpredictable traffic conditions, and the general absence of accessible real-time risk assessment tools. These observations underscore the pressing need for intelligent systems that can effectively enhance situational awareness. Numerous studies and smart mobility initiatives, particularly in cities such as Copenhagen and Amsterdam, demonstrate the benefits of combining urban planning with digital technologies [41]. However, there remains a clear gap in the availability of adaptable, affordable, and bicycle-oriented solutions capable of operating autonomously in real time.

The system delineated in this thesis addresses this gap through a two-part architecture. The Pre-processing unit offers a versatile data pipeline that can transform heterogeneous geospatial CSV files into a structured format, irrespective of the original structure of the dataset. This adaptability capacity is critical for allowing the system to effectively address the diverse urban contexts and risk indicators that characterize different regions. In turn, the primary function of the Cycling Unit is to facilitate run-time operation, integrating GPS input and local risk assessment mechanisms to ensure immediate feedback through a 10-segment LED bar and a buzzer. This modular structure ensures the system's adaptability to different cities while maintaining a low hardware footprint and high responsiveness.

Although large-scale deployment and extended validation are reserved for subsequent phases, the contributions of this work are nonetheless substantial. The proposed system offers a novel and cost-effective solution based on affordable hardware such as the Raspberry Pi Zero 2 W, capable of operating independently of external connectivity. The embedded system promotes safer cycling behavior by informing users of potential risks as they move through urban areas, while also contributing to the larger goals of sustainable mobility and intelligent transportation

systems.

In conclusion, this dissertation lays the foundation for a practical and innovative contribution to urban cycling safety. The proposed solution prioritizes flexibility, scalability, and autonomy, fundamental principles for integration into real-world smart city infrastructures. The insights and tools developed in this study present novel opportunities to improve cyclist safety through digital means, paving the way for future research at this promising intersection of urban mobility, the IoT, and geospatial intelligence. This work represents a first step toward the development of intelligent and cyclist-centered infrastructures within smart cities, where mobility solutions are not only technologically advanced but also inclusive, adaptive, and resilient by design.

6.1 Main Findings and Achievements

The primary achievement of this work lies in developing a dual-module architecture that can transform raw geospatial data into meaningful real-time risk feedback for cyclists.

The proposed system successfully integrates a Pre-processing Unit, which is responsible for organizing arbitrary CSV geospatial files and generating structured spatial grids, and a Cycling unit. The Cycling unit operates on a low-power embedded device to process GPS input and provide intuitive on-the-fly alerts.

The Pre-processing workflow demonstrated considerable flexibility by allowing datasets with varied formats to be consistently normalized and indexed, supporting smooth adaptation to different urban scenarios. Furthermore, implementing the prototype on a Raspberry Pi Zero 2 W proved the feasibility of real-time risk detection under strict hardware constraints, offering a scalable, cost-effective solution.

The case studies in Porto, Portugal, validated the ability of the system to alert cyclists through visual (LED bar) and auditory (buzzer) signals, revealing a clear behavioral alignment between the output of the system and the risk levels associated with each route tested. In addition, performance evaluations confirmed that the solution maintained low CPU and memory usage, ensuring reliability in continuous operation.

6.2 Limitations and Future Work

Despite these achievements, certain limitations were identified that will guide future work. First, the current system relies on static preprocessed data and does not incorporate dynamic sources, such as live traffic feeds, weather conditions, and temporary road events. This restricts the system's ability to adapt to unpredictable or rapidly changing urban conditions in real time.

Second, the current risk classification and feedback logic are based on fixed thresholds and discrete intervals. Although functional, this approach may oversimplify complex environments. Future implementations could benefit from adaptive or fuzzy logic models that better reflect gradual transitions in risk perception and environmental complexity.

Furthermore, the system was validated only in the city of Porto, which may not fully capture the diversity of urban layouts, cycling infrastructures, and risk factors present in other cities. Broader testing across different regions is essential to assess the generalizability and robustness of the system.

In consideration of future efforts, it is imperative to prioritize the integration of real-time data feeds to improve context awareness. In addition, the risk estimation model needs to be refined by incorporating more nuanced urban features, such as lighting, pavement quality, and temporary hazards. Future iterations of the cycling support system should incorporate real-time recognition of road surface conditions to address hazards such as wet, icy, or uneven pavement, which significantly affect cyclist safety. One promising approach is to use lightweight deep learning models optimized for low-power embedded devices. For example, Gui et al. [22] proposed a solution that uses a custom neural network architecture (RefNet) running on an NVIDIA Jetson AGX Xavier. This architecture can classify road conditions with a precision greater than 88% accuracy at 669 frames per second (FPS). Integrating a simplified version of this technology could improve the risk assessment capabilities of the system in dynamic environments, particularly in adverse weather conditions. Furthermore, there is a significant opportunity to improve user feedback interfaces through mobile integration or by providing smart route planning suggestions. In order to enhance the relevance and applicability of the solution in real-world scenarios, it is necessary to scale it for multi-user scenarios and evaluate its performance within larger smart mobility ecosystems.

Bibliography

- [1] Timo Ali-Vehmas and Thomas R. Casey. “Examining Possible Value System Transitions: The Case of Smart Mobility Services”. In: *2015 48th Hawaii International Conference on System Sciences*. 2015, pp. 4366–4375. DOI: [10.1109/HICSS.2015.523](https://doi.org/10.1109/HICSS.2015.523).
- [2] Ruth Ande et al. “Internet of Things: Evolution and Technologies from a Security Perspective”. In: *Sustainable Cities and Society* 54 (July 2019). DOI: [10.1016/j.scs.2019.101728](https://doi.org/10.1016/j.scs.2019.101728).
- [3] Ahmad Ashraff Bin Ariffin, Noor Hafizah Abdul Aziz, and Kama Azura Othman. “Implementation of GPS for location tracking”. In: *2011 IEEE Control and System Graduate Research Colloquium*. 2011, pp. 77–81. DOI: [10.1109/ICSGRC.2011.5991833](https://doi.org/10.1109/ICSGRC.2011.5991833).
- [4] Luigi Atzori, Antonio Iera, and Giacomo Morabito. “The Internet of Things: A survey”. In: *Computer Networks* 54.15 (2010), pp. 2787–2805. ISSN: 1389-1286. DOI: [10.1016/j.comnet.2010.05.010](https://doi.org/10.1016/j.comnet.2010.05.010).
- [5] Saleh Hussein Al-Awami, Mohammed Mahfud Al-Aty, and Mousa Faraj Al-Najar. “Comparison of IoT Architectures Based on the Seven Essential Characteristics”. In: *2023 IEEE 3rd International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA)*. 2023, pp. 305–310. DOI: [10.1109/MI-STA57575.2023.10169289](https://doi.org/10.1109/MI-STA57575.2023.10169289).
- [6] Carlos Carvalho and Rui José. “The gap between cycling practices and mapping services for smart cycling”. In: *2022 IEEE International Smart Cities Conference (ISC2)*. 2022, pp. 1–7. DOI: [10.1109/ISC255366.2022.9922089](https://doi.org/10.1109/ISC255366.2022.9922089).
- [7] Vítor de Castro Paes et al. “Analyzing the Challenges for Future Smart and Sustainable Cities”. In: *Sustainability* 15.10 (2023), p. 7996. DOI: [10.3390/su15107996](https://doi.org/10.3390/su15107996).
- [8] Greig Charnock, Hug March, and Ramon Ribera-Fumaz. “From smart to rebel city? Worlding, provincialising and the Barcelona Model”. In: *Urban Studies* 58.3 (2021), pp. 581–600. DOI: [10.1177/0042098019872119](https://doi.org/10.1177/0042098019872119).
- [9] Abiy Biru Chebudie, Roberto Minerva, and Domenico Rotondi. “Towards a definition of the Internet of Things (IoT)”. PhD thesis. Politecnico di Torino, Aug. 2014.

- [10] Konstantinos Christantonis and Christos Tjortjis. “Data Mining for Smart Cities: Predicting Electricity Consumption by Classification”. In: *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*. 2019, pp. 1–7. DOI: [10.1109/IISA.2019.8900731](https://doi.org/10.1109/IISA.2019.8900731).
- [11] Qingyu Ci. “Research on the Application of Computer Big Data Technology in Smart Travel”. In: *Journal of Combinatorial Mathematics and Combinatorial Computing* 119 (2021), pp. 291–303. DOI: [10.61091/jcmcc119-28](https://doi.org/10.61091/jcmcc119-28).
- [12] Daniel G. Costa and Cristian Duran-Faundez. “Open-Source Electronics Platforms as Enabling Technologies for Smart Cities: Recent Developments and Perspectives”. In: *Electronics* 7.12 (2018). ISSN: 2079-9292. DOI: [10.3390/electronics7120404](https://doi.org/10.3390/electronics7120404).
- [13] Lei Cui et al. “Security and Privacy in Smart Cities: Challenges and Opportunities”. In: *IEEE Access* 6 (2018), pp. 46134–46145. DOI: [10.1109/ACCESS.2018.2853985](https://doi.org/10.1109/ACCESS.2018.2853985).
- [14] Martin David and Florian Koch. ““Smart Is Not Smart Enough!” Anticipating Critical Raw Material Use in Smart City Concepts: The Example of Smart Grids”. In: *Sustainability* 11.16 (2019). ISSN: 2071-1050. DOI: [10.3390/su11164422](https://doi.org/10.3390/su11164422).
- [15] Haipeng Deng et al. “A Geospatial Data and Model Hub for Online Geospatial Analysis”. In: *2023 11th International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*. 2023, pp. 1–6. DOI: [10.1109/Agro-Geoinformatics59224.2023.10233547](https://doi.org/10.1109/Agro-Geoinformatics59224.2023.10233547).
- [16] Sunita M Dol and Pradeep M Jawandhiya. “Use of Data mining Tools in Educational Data Mining”. In: *2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT)*. 2022, pp. 380–387. DOI: [10.1109/CCICT56684.2022.00075](https://doi.org/10.1109/CCICT56684.2022.00075).
- [17] Sérgio Pedro Duarte, Jorge Pinho de Sousa, and Jorge Freire de Sousa. “Co-designing urban mobility solutions in a socio-technical transition context: Guidelines for participative service design”. In: *Journal of Urban Mobility* 6 (2024), p. 100086. ISSN: 2667-0917. DOI: [10.1016/j.urbmob.2024.100086](https://doi.org/10.1016/j.urbmob.2024.100086).
- [18] Ander Eguiluz et al. “Design and implementation of an open-source urban mobility web service based on environmental quality and bicycle mobility data”. In: *2022 7th International Conference on Smart and Sustainable Technologies (SpliTech)*. 2022, pp. 1–5. DOI: [10.23919/SpliTech55088.2022.9854330](https://doi.org/10.23919/SpliTech55088.2022.9854330).
- [19] European Commission. *Cyclists - Facts and Figures*. https://road-safety.transport.ec.europa.eu/system/files/2023-02/ff_cyclists_20230213.pdf. Accessed: 2025-07-12. 2023.
- [20] Matteo Fresta et al. “Supporting a .csv-based Workflow in MongoDB for Data Analysts”. In: *2023 IEEE 32nd International Symposium on Industrial Electronics (ISIE)*. 2023, pp. 1–4. DOI: [10.1109/ISIE51358.2023.10228044](https://doi.org/10.1109/ISIE51358.2023.10228044).

- [21] Dimitrios Glaroudis, Alexandra Sampsonidou, and Eugenia Papaioannou. “Smart Cities, IoT, and e-Government: Applications in Greek Municipalities”. In: *Proceedings* 111.1 (2024). ISSN: 2504-3900. DOI: [10.3390/proceedings2024111026](https://doi.org/10.3390/proceedings2024111026).
- [22] QiXin Gui et al. “Road surface state recognition using deep convolution network on the low-power-consumption embedded device”. In: *Microprocessors and Microsystems* 96 (2023), p. 104740. ISSN: 0141-9331. DOI: [10.1016/j.micpro.2022.104740](https://doi.org/10.1016/j.micpro.2022.104740).
- [23] Vinayak Hegde, T S Aswathi, and R Sidharth. “Student residential distance calculation using Haversine formulation and visualization through GoogleMap for admission analysis”. In: *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*. 2016, pp. 1–5. DOI: [10.1109/ICCIC.2016.7919699](https://doi.org/10.1109/ICCIC.2016.7919699).
- [24] Mennatullah Hendawy et al. “Envisioning of smart cycling in seven cities: Discourse analysis of official city documents”. In: *Ain Shams Engineering Journal* 15.8 (2024), p. 102820. ISSN: 2090-4479. DOI: [10.1016/j.asej.2024.102820](https://doi.org/10.1016/j.asej.2024.102820).
- [25] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *GPS: Theory and Practice*. 5th. Springer, 2001. ISBN: 9783211835340.
- [26] International Energy Agency. *Transport – Tracking Clean Energy Progress*. <https://www.iea.org/reports/transport>. Accessed: 2025-07-12. 2023.
- [27] J. Jayachitra, A. Aslam Khan, and V. Yuvaraj. “IoT-Driven Robotic System for Enhanced Nighttime Perimeter Security and Real-Time Surveillance in Smart Environments”. In: *2025 International Conference on Visual Analytics and Data Visualization (ICVADV)*. 2025, pp. 440–445. DOI: [10.1109/ICVADV63329.2025.10961734](https://doi.org/10.1109/ICVADV63329.2025.10961734).
- [28] Izabela Jonek-Kowalska and Radosław Wolniak. *Smart Cities in Poland: Towards sustainability and a better quality of life?* Routledge, Aug. 2023. ISBN: 9781003358190. DOI: [10.4324/9781003358190](https://doi.org/10.4324/9781003358190).
- [29] Christina Kaß et al. “A Methodological Approach to Determine the Benefits of External HMI During Interactions Between Cyclists and Automated Vehicles: A Bicycle Simulator Study”. In: *HCI in Mobility, Transport, and Automotive Systems. Driving Behavior, Urban and Smart Mobility*. Ed. by Heidi Krömker. Cham: Springer International Publishing, 2020, pp. 211–227. ISBN: 978-3-030-50537-0. DOI: [10.1007/978-3-030-50537-0_16](https://doi.org/10.1007/978-3-030-50537-0_16).
- [30] Wojciech Kozłowski and Kacper Suwar. “Smart City: Definitions, Dimensions, and Initiatives”. In: *EUROPEAN RESEARCH STUDIES JOURNAL XXIV* (Sept. 2021), pp. 509–520. DOI: [10.35808/ersj/2442](https://doi.org/10.35808/ersj/2442).
- [31] Wanwan Li. “Synthesizing Virtual Night Scene on OpenStreetMap”. In: *2023 International Conference on Communications, Computing and Artificial Intelligence (CCCAI)*. 2023, pp. 177–181. DOI: [10.1109/CCCAI59026.2023.00039](https://doi.org/10.1109/CCCAI59026.2023.00039).
- [32] Jing Liang. “Research on Encryption Algorithm and Embedded System Optimization Strategy Based on IoT Security”. In: *Journal of Cyber Security and Mobility* 14.1 (2025), pp. 229–257. DOI: [10.13052/jcsm2245-1439.14110](https://doi.org/10.13052/jcsm2245-1439.14110).

- [33] Yirang Lim, Jurian Edelenbos, and Alberto Gianoli. “What is the impact of smart city development? Empirical evidence from a Smart City Impact Index”. In: *Urban Governance* 4.1 (2024), pp. 47–55. ISSN: 2664-3286. DOI: <https://doi.org/10.1016/j.ugj.2023.11.003>.
- [34] Marco Lombardi, Francesco Pascale, and Domenico Santaniello. “Internet of Things: A General Overview between Architectures, Protocols and Applications”. In: *Information* 12.2 (2021). ISSN: 2078-2489. DOI: [10.3390/info12020087](https://doi.org/10.3390/info12020087).
- [35] Siddharth Mahobia and Rahul Pawar. “Comprehensive Review of Smart Cities Using IoT in Cloud”. In: *International Journal of Innovative Research in Computer and Communication Engineering* (2024). DOI: [10.15680/IJIRCCE.2024.1205194](https://doi.org/10.15680/IJIRCCE.2024.1205194).
- [36] Sudha Ellison Mathe et al. “A Comprehensive Review on Applications of Raspberry Pi”. In: *Computer Science Review* (2024). DOI: [10.1016/j.cosrev.2024.100636](https://doi.org/10.1016/j.cosrev.2024.100636).
- [37] David E. Mills et al. “Smart City Capacities: Extant Knowledge and Future Research for Sustainable Practical Applications”. In: *Sustainability* 16.9 (2024). ISSN: 2071-1050. DOI: [10.3390/su16093719](https://doi.org/10.3390/su16093719).
- [38] Douglas Mitieka et al. “Smart Mobility in Urban Areas: A Bibliometric Review and Research Agenda”. In: *Sustainability* 15.8 (2023). ISSN: 2071-1050. DOI: [10.3390/su15086754](https://doi.org/10.3390/su15086754).
- [39] Sri Banu Munisamy et al. “Enhancing Cyclist Safety: Real-Time Fallen Cyclist Detection and Emergency Notification System”. In: *2024 International Conference on Information and Communication Technology Convergence (IMCOM)*. 2024. DOI: [10.1109/IMCOM60618.2024.10418329](https://doi.org/10.1109/IMCOM60618.2024.10418329).
- [40] Anne H. Ngu et al. “IoT Middleware: A Survey on Issues and Enabling Technologies”. In: *IEEE Internet of Things Journal* 4.1 (2017), pp. 1–20. DOI: [10.1109/JIOT.2016.2615180](https://doi.org/10.1109/JIOT.2016.2615180).
- [41] Anna Nikolaeva. “Smart Cities and (Smart) Cycling: Exploring the Synergies in Copenhagen and Amsterdam”. In: *Journal of Urban Technology* 31.1 (2024), pp. 29–49. DOI: [10.1080/10630732.2024.2322007](https://doi.org/10.1080/10630732.2024.2322007).
- [42] Franklin Oliveira et al. “A Survey of Technologies and Recent Developments for Sustainable Smart Cycling”. In: *Sustainability* 13.6 (2021). ISSN: 2071-1050. DOI: [10.3390/su13063422](https://doi.org/10.3390/su13063422).
- [43] João Paulo Just Peixoto et al. “CityZones: A Geospatial Multi-Tier Software Tool to Compute Urban Risk Zones”. In: *SoftwareX* 23 (2023), p. 101409. DOI: [10.1016/j.softx.2023.101409](https://doi.org/10.1016/j.softx.2023.101409).
- [44] Mousi Perumal et al. “Spatial Data Mining Approaches for GIS – A Brief Review”. In: *Emerging ICT for Bridging the Future - Proceedings of the 49th Annual Convention of the Computer Society of India CSI Volume 2*. Ed. by Suresh Chandra Satapathy et al. Cham: Springer International Publishing, 2015, pp. 579–592. ISBN: 978-3-319-13731-5. DOI: [10.1007/978-3-319-13731-5_63](https://doi.org/10.1007/978-3-319-13731-5_63).

- [45] Maryam Qonita, Rini Rachmawati, and R. Rijanta. “Assessing Amsterdam’s urban mobility through the lens of smart mobility goals”. In: *Case Studies on Transport Policy* 20 (2025), p. 101443. ISSN: 2213-624X. DOI: [10.1016/j.cstp.2025.101443](https://doi.org/10.1016/j.cstp.2025.101443).
- [46] Shammya Shananda Saha et al. “A Framework for Generating Synthetic Distribution Feeders using OpenStreetMap”. In: *2019 North American Power Symposium (NAPS)*. 2019, pp. 1–6. DOI: [10.1109/NAPS46351.2019.9000187](https://doi.org/10.1109/NAPS46351.2019.9000187).
- [47] Fadi Al-Turjman and Ayman Radwan. “Data Delivery in Wireless Multimedia Sensor Networks: Challenging and Defying in the IoT Era”. In: *IEEE Wireless Communications* 24.5 (2017), pp. 126–131. DOI: [10.1109/WCM.2017.1700054](https://doi.org/10.1109/WCM.2017.1700054).
- [48] Annemiek van Marsbergen et al. “Exploring the role of bicycle sharing programs in relation to urban transit”. In: *Case Studies on Transport Policy* 10.1 (2022), pp. 529–538. ISSN: 2213-624X. DOI: [10.1016/j.cstp.2022.01.013](https://doi.org/10.1016/j.cstp.2022.01.013).
- [49] Md Whaiduzzaman et al. “A Review of Emerging Technologies for IoT-Based Smart Cities”. In: *Sensors* 22.23 (2022). ISSN: 1424-8220. DOI: [10.3390/s22239271](https://doi.org/10.3390/s22239271).
- [50] Wikipedia contributors. *Geographic coordinate system — Wikipedia, The Free Encyclopedia*. Accessed: 2025-07-12. 2024. URL: https://en.wikipedia.org/wiki/Geographic_coordinate_system.
- [51] Radosław Wolniak. “Analysis of the Bicycle Roads System as an Element of a Smart Mobility on the Example of Poland Provinces”. In: *Smart Cities* 6.1 (2023), pp. 368–391. ISSN: 2624-6511. DOI: [10.3390/smartcities6010018](https://doi.org/10.3390/smartcities6010018).
- [52] Radosław Wolniak. “Smart mobility in smart city – Copenhagen and Barcelona comparison”. In: *Scientific Papers of Silesian University of Technology Organization and Management Series* 2023 (June 2023). DOI: [10.29119/1641-3466.2023.172.41](https://doi.org/10.29119/1641-3466.2023.172.41).
- [53] Ying Ye, Chen Zhong, and Esra Suel. “Unpacking the perceived cycling safety of road environment using street view imagery and cycle accident data”. In: *Accident Analysis & Prevention* 205 (2024), p. 107677. ISSN: 0001-4575. DOI: [10.1016/j.aap.2024.107677](https://doi.org/10.1016/j.aap.2024.107677).