*Author:*
**Strain, Tom J**

*Title:*
**On Sensors in Transportation Asset Management**

*Applications of Data Science and Mathematical Modelling*

# On Sensors in Transportation Asset Management

*Applications of Data Science and Mathematical Modelling*

By

TOM STRAIN


Department of Engineering Mathematics
UNIVERSITY OF BRISTOL

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of DOCTOR OF PHILOSOPHY in the Faculty of Engineering.

DECEMBER 2021

Word count: thirty five thousand four hundred and thirty three

# Abstract

Fixed, remote survey, and ad-hoc sensor deployments are investigated to examine how intelligent transportation technologies can enrich the transportation asset management process. Initially, we consider whether data captured from a structural health monitoring system installed on a bridge might also provide an estimate of the number of vehicles that travel across it. A case study of the Clifton Suspension Bridge in Bristol, UK is developed. Overall, on a withheld test set, our spectrogram thresholding-based method achieves a 1.36 vehicle accuracy, outperforming an existing state-of-the-art system ROCKET.

Second, a computer vision-based decision support system, designed to automatically verify an inventory of roadside assets, is developed and tested with assets along an eight-kilometre section of the A27 highway in England, UK. To develop the system, remote survey data from a vehicle equipped with a forward-facing camera and a GPS-enabled inertial measurement unit (IMU), aerial highway imagery, and the asset inventory, are fused. Overall, 91% of assets in a withheld test set are automatically verified, thus greatly reducing an analyst's manual workload. We also explore how visual simultaneous localisation and mapping (vSLAM) systems might make our method more robust, by refining or replacing missing IMU data in GPS-denied environments.

Finally, we consider a case study of how the Highways England traffic officer (TO) fleet might capture asset data across the strategic road network in England, UK, alongside their primary function of incident response. The TOs patrol under one of two distinct regimes: one that aims to minimise the fleet's incident response time, and one that aims to maximise the fleet's coverage (for asset data capture). Comparison of the network coverage and incident response times achieved under each regime shows that fleets (of various sizes) can successfully be deployed for asset data capture while only increasing incident response times by a few minutes.

## Dedication and acknowledgements

Firstly, I would like to thank my PhD supervisor R. Eddie Wilson who has provided an unquantifiable amount of guidance and support — both technical and personal — during the four years we have worked together. If I had the choice again, knowing what I know now, I would chose the same PhD project and supervisor all over again. I would also like to extend my thanks to my industrial supervisor (for the majority of the project) Roger Littleworth, whose inputs and perspectives embedded my research in reality.

Thank you to the cohorts of Engineering Mathematics PhD students (or BUNCAERites) I have had the pleasure of working alongside. Your conversations, seminars, coffee breaks, runs, and pub trips have no doubt shaped this thesis.

To my family, for always being there and for your emotional and financial support throughout my education. Thank you Mum, the mathematics teacher, from whom I've inherited the scientific way-of-thinking required to take on this PhD, and thank you Dad for your wisdom.

Lastly, thank you Gemma, who I met a month into the PhD but I now get to call my wife. You have bore the brunt of my research frustrations and difficulties over the last four years, as well as sharing and celebrating my (occasional) successes. You have always taken a genuine interest in my research, and I truly do not know how I would have gotten through this without you.

## AUTHOR'S DECLARATION

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: ..... DATE: .....

# TABLE OF CONTENTS

# List of Tables

# List of Figures

CHAPTER 1

# INTRODUCTION

A successful society relies on convenient and readily available infrastuctures and services such as water, electricity, and roads. These critical infrastructures, defined as, the “basic systems and services that a country or organization uses in order to work effectively” [37] require an extensive set of people (e.g., operators and maintenance staff) and processes (e.g., planning, installation, and maintenance), that we seldom consider — we often take for granted that our kettle boils when we flick the switch, that the trains run on time, and that clean water runs from the tap on demand.

To ensure such critical infrastructures meet the desired level-of-service (i.e., they provide a suitable quality of service to the user), governments set requirements that are met through collaboration with public agencies or industry partners. For example, the Infrastructure and Projects Authority (IPA) is an executive agency of the UK government that works closely with the National Infrastructure Committee (NIC, an impartial advisory board). Together, the IPA and NIC have produced a roadmap detailing their vision of how critical UK infrastructure is to be improved and maintained by 2030 — largely moving towards a digitalised future of ‘smart infrastructures’ that enable data-driven decision making [134]. The roadmap was developed alongside a set of future priorities recommended by the NIC, including: introducing more inter-city public transport systems; improving the flood and drought resilience of our existing infrastructure (in response to the ongoing climate emergency); and installing fibre-optic broadband across the entire country [180]. Similarly, Infrastructure Australia (IA) advises the Australian government on infrastructure investment, and annually publishes the Infrastructure Priority List — IA’s recommended "investment opportunities that can deliver nationally significant benefits to Australians" [135].

In this thesis, we focus on highway and road infrastructure; that is, the transport networks

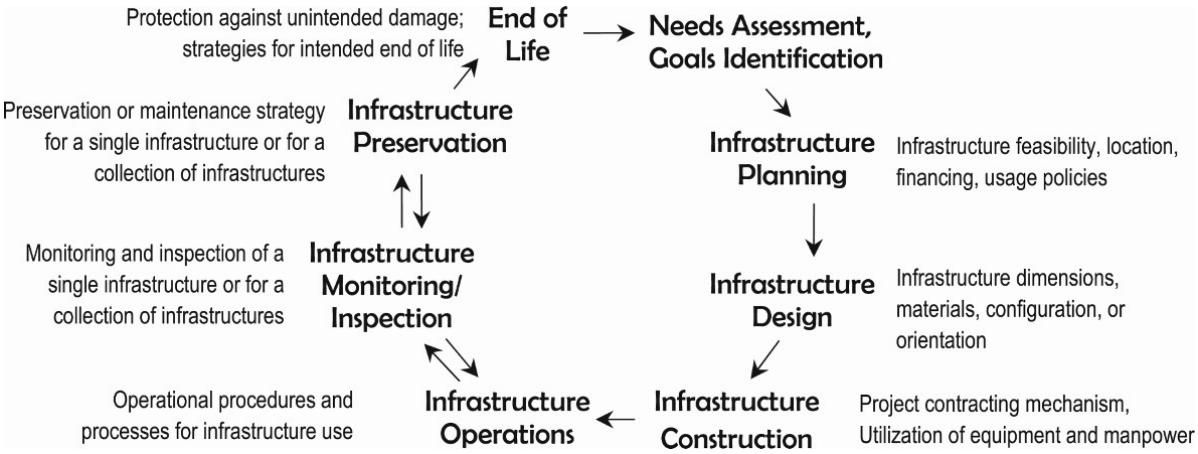



Figure 1.1: Infrastructure lifecycle, reproduced from Sinha et al. [232].

and systems that allow us to travel across the country, usually in our privately owned vehicles. A safe and operational road network is a critical component of society, without which, we could not (easily) visit our friends and families, and businesses could not trade; in 2019, 11.84 billion tonnes of freight was transported via road in the U.S. [236] and 79% of all freight in the UK was moved by road in 2018 [61]. As such, countries invest heavily to continuously improve and repair highway and road infrastructure. The Organisation for Economic Co-operation and Development (OECD) reports that the U.S., Japan, Germany, and Australia respectively invested 93, 30, 16, and 14 billion Euros into their national road infrastructure in 2019 [193].

## 1.1 Transportation Asset Management

At the heart of any physical infrastructure is its assets; that is, its equipment and furniture. Road infrastructure is comprised of a multitude of assets, ranging from the pavement surface to maintenance vehicles. For example, over 121,000 lighting units, 70,000 sensors (e.g., vehicle detectors and cameras, see page 16), 20,000 large structures (e.g., bridges and gantries), and 13,000 km of crash barriers are installed across the strategic road network (SRN, see page 101) in England [115].

To ensure that highway and road assets are properly monitored and maintained, agencies (i.e., those that manage infrastructure) follow a set of business practices known as transportation asset management (TAM). The U.S. Federal Highway Administration (FHWA) define TAM as the "strategic and systematic process of operating, maintaining, upgrading, and expanding physical assets effectively throughout their lifecycle" [253], which can be understood in terms of the three following processes:

1. maintaining or improving the current condition of assets;
2. developing processes and projects to improve the future condition of assets; and

3. managing assets in a cost-effective way.

The final process is key. Essentially, TAM helps an agency to maximise the performance of their system from their (often stretched) budgets. This can be realised by, for example, preemptively strengthening a bridge (as a result of a timely inspection) before it fails and incurs large costs in damages and repairs. Fundamentally, TAM aims to derive the maximum value from infrastructure throughout its lifecycle; that is, from construction (or installation) through to disposal, see Figure 1.1, while always meeting the desired level-of-service.

To regulate and perform effective TAM, agencies follow procedures defined in a TAM plan (TAMP) that are developed by government or national-level departments. For example, highway agencies from the Netherlands, Ireland, Italy, and the UK are currently collaborating with research centres across Europe on the Asset Monitoring for Infrastructure (AM4INFRA) project [69]. The initiative aims to provide an asset management framework to "enable consistent and coherent cross-asset, cross-modal and cross-border decision-making" to ensure standardized, safe, and value-for-money TAM across Europe. Similarly, the U.S. Department of Transportation (DoT) provides extensive guidance on how to develop an effective TAMP [9]. Furthermore, TAMPs became a statutory requirement for every U.S. state in 2005 (signed into law under the 'Moving Ahead for Progress in the 21st Century' act [254]) — the TAMP developed by each U.S. state highway agency can be found in [255].

There is unfortunately no one-size-fits-all TAMP, as they are often tailored to an asset's application. However, Sinha et al. [232] recommend the following high-level TAMP components:

- **Comprehensive inventory and monitoring of assets and physical operating systems**

  Deploying systems (automated or manual) to determine the asset's condition (e.g., corrosion or failure) and recording it in an inventory.

- **Database development**

  Designing systems to securely store the asset inventory, making it easily interpretable and retrievable. The inventory may contain the asset condition, position, installation, and maintenance history, for example.

- **Performance assessment and modelling**

  Using historical asset data to model and predict the expected future performance of an asset (e.g., computing expected failure rates).

- **Evaluate standard and alternative treatments based on cost benefit analysis**

  Given the current asset state (via the inventory) and predicted future state, evaluate the effectiveness of various asset treatment (i.e., improvement) strategies and weigh up their

costs against the available budget. For example, each strategy may be scored by each asset's future life-span, value, or usage.

- **Project selection and implementation**

  Determine and implement a set of long and short-term asset improvement projects.

Feedback and self-evaluation are important aspects of a TAMP. Ideally, all asset condition assessment, improvement, and maintenance projects are evaluated after their completion, alongside a budget review to improve future TAMPs.

As well as identifying cost-savings and deriving maximum value from an asset, effective TAM provides a framework (via pre-determined performance measures) to accommodate the concerns and priorities of all stakeholders. For large infrastructures, stakeholders might include a large group of people and entities — for example, a road improvement scheme may involve residential, environmental, and road-user stakeholders before even considering the dedicated project team. Furthermore, TAM practices promote integration of activities, potentially performed by disparate teams and resources across an agency, and provide a consistent management framework, irrespective of asset type. A comprehensive review of TAM practices and plans, and their future research directions is provided by Sinha et al. [232].

## 1.2 Intelligent Transport Technology

Intelligent transport technologies (ITTs), such as cameras, accelerometers, and LIDAR have revolutionised the way in which TAM is performed. Historically, all asset inventory and condition assessment (i.e., the first component in Sinha's high-level TAMP) was performed manually; i.e., on-site physical inspection. However, today, industry practitioners and analysts are supported by additional data sources (e.g., digital imagery) that provide rich views of assets and their condition. Thus, an asset may be inspected in fine detail, post-inspection, and potentially with some degree of autonomy (i.e., the asset's condition is automatically assessed). Note, for large structures where failure may pose a severe risk (e.g., a bridge failure), on-site human inspection is required [123].

The sensors used within TAM can be broadly categorised by their mode of deployment; fixed, remote survey, or ad-hoc, see Figure 1.2. Each mode is now described.

**Fixed**

The first group of sensors are those fixed onto an asset (either temporarily or permanently). These *static* sensor deployments are often used to monitor large civil assets (such as a bridge or building [194, 233]); for example, accelerometers and displacement transducers are often installed as part of a structural health monitoring (SHM) system, see Section 2.1. Similarly, vehicle detection systems often use static inductance loop detectors (ILDs) [46, 129] installed into the road surface, see Section 2.1.1.

(a)  (b) 

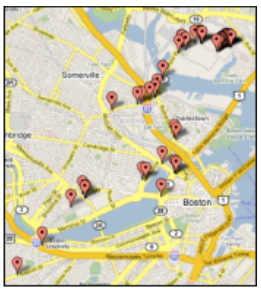
(c) 


FIGURE 1.2. Sensing modes in transportation asset management (TAM). (a) Fixed accelerometer installed onto a bridge. (b) Specialised vehicle used to collect pavement condition data on a drive-by remote survey. (c) Potholes detected ad-hoc by a fleet of taxis in Boston. Panels (a), (b), and (c) are respectively reproduced from Carrión-Viramontes et al. [39], Highways England [115], and Eriksson et al. [67]

Static sensors are simple to operate (compared with remote survey and ad-hoc deployments) as they are usually installed and then left unattended, and are designed to collect high-quality targeted data; that is, they are deployed with a single monitoring application in mind (e.g., vehicle detection).

**Remote Survey**

Alternatively, a sensor that is deployed via a remote survey is a *dynamic* capability that 'visits' an asset. Thus, multiple assets may be monitored with a single sensor. The general idea, is that a sensor (e.g., camera or LIDAR) is transported (e.g., via vehicle [266] or aircraft [250]) to a group of assets.

Remote surveys are commonplace in TAM. For example, agencies often collect imagery of roadside highway assets such as traffic signs via drive-by surveys; that is, a specialised vehicle equipped with a camera drives along the length of a highway while capturing imagery data of assets [138] (some sophisticated survey vehicles are also equipped with LIDAR [266]), see page 55.

**Ad-hoc**

Ad-hoc sensing deployments are a relatively new concept, compared with traditional sensing methods, that aim to address an important shortcoming of remote surveys — assets can be left unmonitored for long periods of time if the surveys are infrequent. For example, the UK Department for Transport (DfT) remotely surveys the pavement condition (i.e., checking for potholes, cracks, and rutting) of the entire road network in the UK once every two years [60]. Thus, pavement potholes and cracks can be unattended, potentially, for years at a time.

Instead, ad-hoc sensing methods collect asset data 'on-the-fly', often using services or vehicle fleets not designed for TAM purposes. For example (and continuing with the theme of pavement condition monitoring), a number of systems have been proposed that capture city-wide pavement

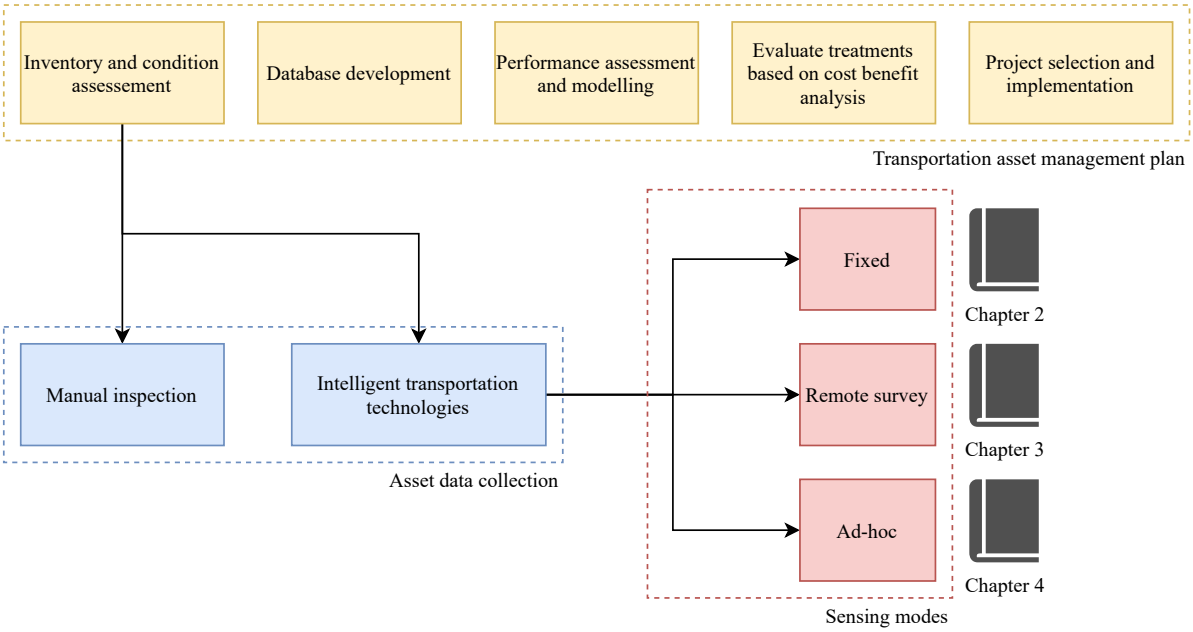



FIGURE 1.3. The thesis's three technical chapters in the context of the wider transportation asset management problem.

data from accelerometers installed in taxis (i.e., potholes can be detected from sudden 'jerks' of the vehicle), all while the taxis continue to pick up and drop off passengers as normal [67, 190].

Similarly, today, SHM systems are being reimagined as an ad-hoc sensing capability. Rather than relying on traditional static sensors, the innovative methods reviewed in [168] instead use in-vehicle sensors to determine the structural health of a bridge as a vehicle drives across it.

## 1.3 Research Objectives and Project Context

The over-arching aim of this thesis is to explore how ITTs can enrich the TAM process. Specifically, we investigate how data captured by ITTs (either fixed or remote survey sensors) might be used to provide decision support or added value in TAM, and consider novel ad-hoc asset data collection methods.

The work that we present here was sponsored by Jacobs through an NPIF (National Productivity Investment Fund, award number 2107418) grant. Jacobs are contracted by Highways England (HE)[1], the largest highway agency in the UK to manage, coordinate, and perform a number of TAM practices [137]. Jacobs therefore have access to a number of TAM-based data sources (e.g., accelerometer data and imagery), and hoped that the PhD would produce new

[1]Highways England was known as the Highways Agency prior to 2015, see page 99. Highways England again rebranded in August 2021, and changed their name to National Highways. In this thesis, we refer to the agency as HE — the agency's name throughout the majority of the PhD.

research and/or working systems, that use their data, to improve their current TAM operations. Indeed, when we refer to the 'analyst' during the thesis, we usually mean a Jacobs employee, tasked with a particular TAM operation.

As a consequence, each case study in each chapter was, to some degree, selected according to Jacobs' interests in this topic, and those of their principle clients and stakeholders. To be clear, Jacobs had little input into the initial research direction of each case study (provided by the author), rather, they provided the data and context around the operational landscape in which our interventions sit. Similarly, we liaised with HE on a number of occasions during the project, who acted as a soundboard for our ideas, and ensured the practicality of our work in the real world

To realise our aims, this thesis contains three technical chapters that each focus on one of the three sensing modes (i.e., fixed, remote survey, or ad-hoc), see Figure 1.3. In doing so, we consider ITTs used right across the breadth of TAM. Accordingly, we define the following high-level research questions (RQs) considered in this thesis.

RQ1. *Can extra value be derived from pre-existing fixed sensors; that is, can they be used for a secondary purpose in addition to their primary function?*

RQ2. *To what degree can TAM be automated; to what degree is there scope for decision support systems to assist agency analysts?*

RQ3. *Are there ad-hoc sensing capabilities available to an agency, so that they may regularly capture asset data, rather than relying on scheduled remote survey data?*

## 1.4 Thesis Structure

Following this introduction, the remainder of the thesis is split into four chapters. Note that unlike some thesis styles, we do not include a single comprehensive literature review chapter and glossary. Rather, each technical chapter contains its own review of relevant background material and literature and glossary.

Chapter 2 aims to address research question RQ1 by considering the data collected by an SHM system installed on the Clifton Suspension Bridge (CSB) in Bristol. We identify two shortcomings of existing systems: (1) structural health data (from fixed sensors) is rarely used for a secondary purpose; and (2) vehicle detection systems usually rely on specialised systems (such as ILDs) that often require a road closure to install (as the surface must be cut into). Therefore, we develop a system to count the number of vehicles that travel across the CSB, purely from data collected by an SHM setup. Thus, our method adds value to existing fixed sensor installations. Our methods are tested with ground-truth vehicle counts collected from toll-barrier data collected in a previous study [94], and by human observers during a second trial.

In Chapter 3, a computer vision-based decision support system is developed to automatically verify an inventory of roadside assets (e.g., traffic signs and reference marker posts), and thus, we address research question RQ2. Currently, asset inventory updating is performed via manual inspection of drive-by survey data. Our proposed system automatically identifies those assets that are recorded incorrectly in the inventory and require updating. This may be case if, for example, the asset has been moved between surveys, or its physical properties, such as width, height, or position are recorded incorrectly in the inventory.

Furthermore, the number of assets that are incorrectly recorded is usually small compared with those that are correct. Thus, our system significantly reduces the manual work of an analyst. The system was developed and tested with a set of roadside assets installed along the A27 highway on the south coast of England. Further, image-based localisation methods are explored to improve the robustness of the system.

Chapter 4 aims to address research question RQ3 by considering the suitability of highway traffic officers (TOs, i.e., a fleet that patrols across a highway network and responds to traffic incidents) as an ad-hoc asset data capture capability. We consider, as a proof-of-concept, a case study on the HE TO fleet that patrols along the SRN. A simulator is built in which the fleet patrols and responds to dynamically generated incidents. The fleet patrols under one of two distinct regimes; one that prioritises fast incident response, and one that prioritises the fleet's network coverage (for asset data capture). The incident response times and coverage achieved under each regime are compared to determine the feasibility of our proposed schemes.

Finally, the thesis is concluded in Chapter 5 where a summary of the research highlights are presented, and directions for future work are discussed.

In this thesis, we present TAM systems and processes developed from data science, computer vision, and mathematical modelling perspectives — which might not be considered in typical TAM studies (e.g., designing structures and processes to maximise asset value). As such, the technical contributions in each chapter demonstrate the offerings these methodologies — where my skill sets lie — bring to the area of TAM. Furthermore, with the envisioned 'smart infrastructures' (see page 1) of the near future, the intersection of data science and TAM, considered here, is only going to become increasingly relevant in the coming years.

### 1.4.1 Original Contributions

The original contributions to knowledge produced during this PhD are listed below in order of publication (or submission) date.

- Developed a system to estimate the number of vehicles that travel across a bridge from data captured by an SHM system (installed onto the bridge). Research was published in the Proceedings of the International Conference on Smart Infrastructure and Construction (ICSIC) 2019 [238].

- Developed a computer vision-based decision support system designed to automatically verify an inventory of roadside assets. Research published in the Transportation Research Record [241].
- Augmented an existing image-based localisation system to estimate the position of a survey vehicle on the highway. Research published in the Proceedings of 23$^{rd}$ IEEE Intelligent Transportation Systems Conference (ITSC) 2020 [239].
- Built a simulator to consider whether TOs (equipped with a dashboard camera) may be used as an ad-hoc asset data capture capability. Research accepted for publication in the Proceedings of the 2022 Transportation Research Board Conference and selected for review for publication in the Transportation Research Record [240].

CHAPTER 2

# ESTIMATION OF VEHICLE COUNTS FROM THE STRUCTURAL RESPONSE OF A BRIDGE

*Sections 2.2–2.6 are based on work summarised in the paper 'Estimation of Vehicle Counts from the Structural Response of a Bridge' published in the Proceedings of the International Conference on Smart Infrastructure and Construction 2019 (ICSIC) [238].*

**Problem Statement**

An accurate vehicle count estimation along a road segment enables suitable road improvement schemes (e.g., road widening) and effective traffic management (e.g., incident detection). Traditional vehicle counting technologies can be expensive and difficult (as a road closure is required) to install and maintain, and can lead to road surface failure. Therefore, we investigate whether data collected by an accelerometer, installed as part of a structural health monitoring system (SHM), might provide a vehicle count estimation instead, thus adding value to pre-existing sensor setups.

**Chapter Glossary**

| Term | Definition |
|---|---|
| AIC | Akaike information criterion |
| API | Application programming interface |
| AR | Auto-regressive |
| ARMA | Auto-regressive moving average |
| CNN | Convolutional neural network |
| CSB | Clifton Suspension Bridge |
| IAE | Instantaneous amplitude envelope |
| IID | Identically and independently distributed |
| ILD | Inductance loop detector |
| KLT | Kanade-Lucas-Tomasi (tracker) |
| LW | Leigh Woods |
| MAE | Mean average error |
| MAPE | Mean average percentage error |
| MIDAS | Motorway incident detection and automatic signalling |
| MRMS | Moving root mean square |
| PACF | Partial auto-correlation function |
| ROCKET | Random convolutional kernel transformation |
| SGD | Stochastic gradient descent |
| SHM | Structural health monitoring |
| SRN | Strategic road network |
| ST | Spectrogram thresholding (method) |
| TSR | Time series regression |
| WSN | Wireless sensor network |
| YOLO | You only look once |

Structural health monitoring (SHM) systems enable those responsible for the safe and proper maintenance of civil infrastructure to make data-driven decisions when managing structural damage and deterioration. Structural health monitoring systems usually rely on sensors (e.g., accelerometers and displacement transducers) that are fitted to an asset (e.g., a bridge or large building) so that any movement of its structural components may be recorded.

Sensing capabilities such as these, that perform remote and intelligent SHM, provide rich data sources, however, this data is rarely used for a secondary purpose. Further, once an SHM system has been installed onto an asset, it usually only requires a small amount of maintenance (battery replacement, for example), and provides structural health data over long periods of time. Thus, there might be great potential to add value by using the SHM data for secondary purposes, that were not foreseen in the original deployment.

An accurate estimation of the number of vehicles that travel along a road segment is an important statistic for an agency to collect, that might subsequently inform decisions on road maintenance and improvement schemes (e.g., widening), and over short time scales, ensure the safety of road users (e.g., by incident detection) [109]. Traditionally, vehicle counts are collected via inductance loop detectors (ILDs) installed in the road surface, however, they are expensive to install, often requiring a road closure to do so.

Therefore, in this chapter, we consider whether data collected from an SHM system might also be used to provide an estimation of vehicle counts, in addition to its primary function of collecting structural health data. While SHM systems are deployed across a range of civil structures [33, 156], in this work, we specifically consider SHM deployments on bridges, that form key links in many transportation networks.

## 2.1 Background

Structural health monitoring has applications across the civil, mechanical, and aeronautical engineering domains [70], and is defined in [34] as the process of "designing and implementing a strategy to identify structural damage". The term structural health monitoring covers a range of activities and practices, such as sensor installation and manual condition assessment, for example, to ensure the 'fitness of purpose' [34] of large structural assets.

Structural health monitoring systems provide operators and maintenance teams with information on gradual or sudden changes to a structure's condition. An SHM system can be broadly understood as three separate components; namely, a sensing capability, a data processing system, and a condition assessment system [2], and is usually deployed as in one of the following high-level use cases identified by Moss and Matthews [177]:

- fatigue assessment;
- structures subject to long-term degradation of materials (e.g., corrosion);

- monitoring of structures affected by external works (tunnelling under buildings and road works on bridges, for example);
- monitoring during demolition (to ensure safe demolition); or,
- novel systems of construction (that use new materials with unknown physical properties).

Rather than provide a definitive condition assessment of a structure, SHM systems are usually designed to determine when its condition has changed, compared against an established 'normal' state, thus prompting an operator to perform an on-site inspection. Note that, in this sense, SHM systems are analogous to the computer vision-based system developed in Chapter 3, in that they help prioritise the work of human analysts.

While a number of large structures and SHM practices pre-date modern computing technology, today, SHM systems tend to refer to the computer-based hardware (e.g., accelerometers and displacement transducers) and software to collect, visualise, and interpret structural condition data. One notable SHM system was deployed on the Humber bridge in the UK [33], see Figure 2.1(a). The aim of the deployment was to extensively record the bridge's acceleration and displacement in various weather conditions in order to validate mathematical models of the bridge's structural response to wind (known as wind flutter, see the infamous Tacoma Narrows incident [154]). Similarly, stress and strain gauges were installed on the structural components of the Republic Plaza building in Singapore to monitor its condition during and after construction [35], see Figure 2.1(b). The gauges revealed how the building's dead (i.e., static) load distributed through its structural components, and more importantly, how the distribution changed over the building's operational life-span.

More recently, SHM systems have improved by using wireless sensor networks (WSNs) to reduce installation and maintenance time and costs, and improve scalability [2]. The key idea behind a WSN-based SHM system is that each sensor (that represents a node in the network) is wirelessly connected to a central base station computer (either directly or via an access point), and thus, its data may be remotely collected and analysed.

Although WSNs offer compelling advantages to traditional wired systems, they pose their own set of challenges to overcome. For example, the rate at which a sensor may be sampled is limited by the WSN's bandwidth (usually lower than its wired counterpart), and the sensors rely on battery-based power sources that may need replacing. Furthermore, a WSN's topology is chosen to suit each structure and application. WSN-based SHM systems usually employ a star or cluster tree topology [2], however peer-to-peer and mesh-grid topologies have also been used [263]. A number of other factors, such as sensor position and orientation, and the number of access points require detailed consideration in the design of the system. Previous WSN-based SHM systems include those deployed in a medieval tower by Zonta et al. [276], on a highway bridge in New York, USA by Whelan and Janoyan [263], and on a railway bridge by Ghosh et al. [80].

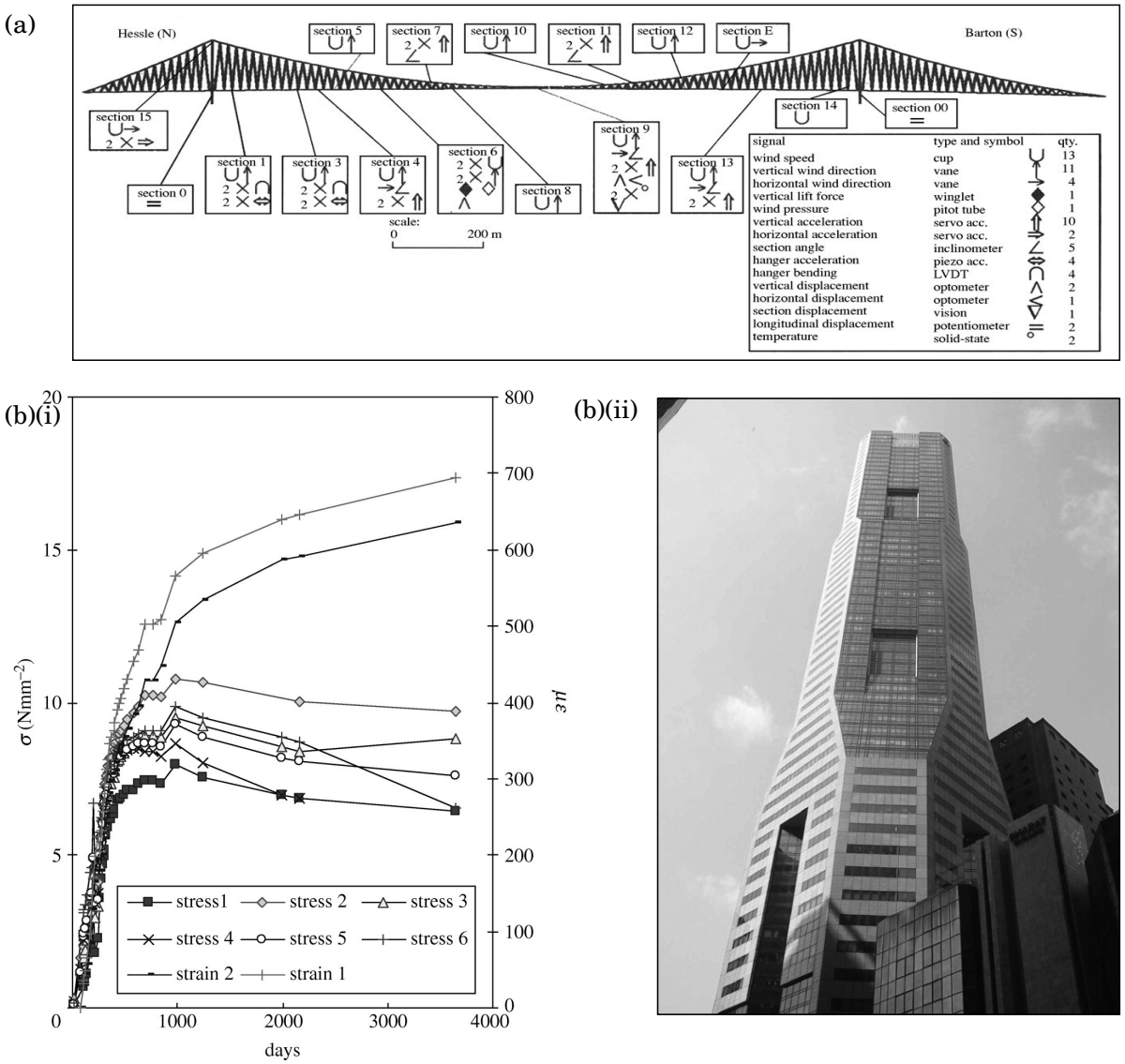



FIGURE 2.1. Previous SHM deployments on the (a) Humber bridge, UK and (b) Republic Plaza building, Singapore. Panel (b)(i) shows the data collected by stress and strain gauges installed in the building, shown in panel (b)(ii). Panels (a) and (b) are respectively reproduced from Brownjohn et al., 1994 [33] and Brownjohn et al., 1998 [34].

### 2.1.1 Vehicle Counting Technologies

Automatic vehicle counting systems can be broadly split into two groups; on-road and roadside. The first group of technologies are installed on (or in) the road itself, see Figure 2.2, for example, pneumatic tubes that are laid across the road surface (often temporarily) [108] and piezoelectric sensors [271]. In contrast, the second group are installed next to, or in close proximity to the road

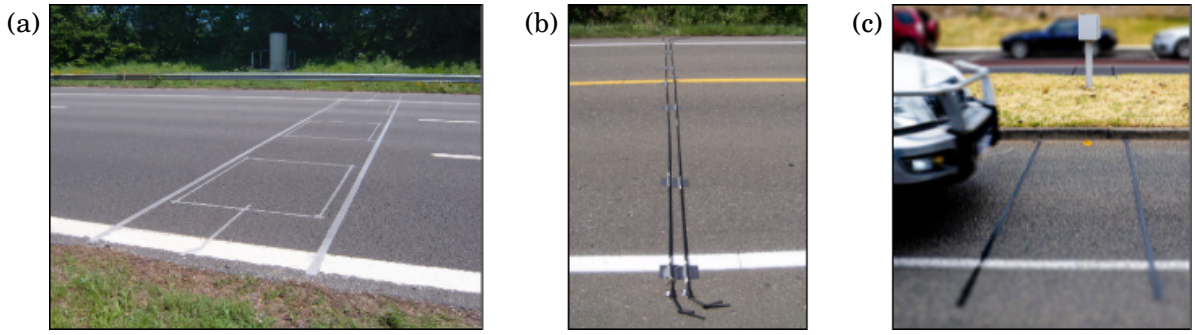



FIGURE 2.2. On-road vehicle counting technologies. (a) Inductance loop detector, (b) pneumatic tubes, and (c) piezoelectric sensors. Panels (a), (b), and (c) respectively reproduced from Electronique Contrôle Mesure Website [66], QUORA website [211], and Retail Sensing website [219].

(on a gantry or bridge, for example), such as radar and camera-based systems [108, 234].

Due to their low cost, and the relative immaturity of other emerging technologies, ILDs are the most commonly used vehicle counting system (on-road or otherwise). For example, in the UK, Highways England (the largest highway agency in the UK, see page 98) employ the MIDAS (Motorway Incident Detection and Automatic Signalling) system — a network of ILDs installed across the strategic road network (SRN, see page 101) in England to determine traffic levels and detect congested traffic conditions [129]. Highways England are also currently trialling a radar-based roadside system to complement the ILD data [129].

An ILD consists of an electrical circuit in a loop installed either singly or as a pair. As a vehicle travels over the loop, the vehicle's metallic components alter the ILD's inductance and produce a signature in the inductance time series, thus, alerting the system to the presence of the vehicle [72]. Despite their prevalence, ILDs have their limitations. To install an ILD, the road (or lane) must be closed and the road surface cut into, resulting in high installation costs and traffic disruption. In addition, ILDs can be subject to the stresses of vehicle loading and temperature [72] and cause failure of the road surface [29]. Further, ILDs perform best in single-lane and steady-flow conditions, and thus, vehicle detection can be poor in multi-lane real-world conditions, where, for example, a vehicle may change lane while travelling across a pair of parallel ILDs [7].

Inductance loop detectors are often used in pairs, where one ILD is positioned downstream of the other to determine a vehicle's speed (known as a speed trap). The vehicle speed is simply computed by dividing the distance between the two detectors by the time between the two detections [146]. Alternatively, a number of systems that compute vehicle speed, and perform vehicle classification from the inductance signatures provided by a single ILD (in addition to vehicle detection) have been proposed. For example, in the work presented by Oh et al. [189], features (such as slew, maximum amplitude, and duration) are extracted from inductance signatures which are then used to determine vehicle speed and type (defined by the vehicle length). Overall,

on a withheld test set, an average speed error of $1.52\,\mathrm{m\,s^{-1}}$ and vehicle classification accuracy of 85% was achieved.

Thanks to the development and success of computer vision for image classification and detection tasks (see Section 3.1.1), a large number of systems that provide vehicle counts from imagery have also been developed. These systems are an attractive proposition, as cameras are a relatively cheap sensing capability, whose data is easily interpreted by a human (i.e., their data does not require any post-processing, unlike radar or LIDAR data, for example [132]), and furthermore, CCTV cameras are already installed on many junctions, highways, and urban areas.

Early computer vision approaches aimed to segment vehicles from the image using background subtraction; that is, the methods first determine (and potentially update) what an empty road scene looks like, which is then compared against new imagery to determine when a vehicle is present [153, 228]. However, thanks to the advent of deep learning, today, researchers have been able to employ convolutional neural networks (CNNs, see Section 3.3.5) to develop highly effective image-based vehicle counting systems instead.

In [234], a novel system is proposed which works in three stages. Firstly, the highway area is segmented from the image, before the YOLOv3 network (you only look once, see page 89) is used to detect vehicles on the road. Lastly, a tracking algorithm, that matches ORB features extracted from vehicles (see Figure 3.17 on page 85) is used to determine which detections in different images correspond to the same vehicle. The authors report an average 7% vehicle count error over three real-world highway video scenes captured by elevated CCTV cameras. A similar system is developed by Abdelwahab [1] who instead uses the R-CNN architecture (that first identifies candidate regions where an object may appear, before classifying the region) to perform vehicle detection, and a Kanade Lucas Tomasi-based (KLT) vehicle tracker, that tracks the coordinates of the detected vehicle's bounding box. For a comprehensive description of image-based vehicle tracking and counting methods, we refer the interested reader to the review provided by Al-Smadi et al. [3].

Previous works on vehicle counting systems that employ sensors typically found in SHM setups (i.e., accelerometers) are relatively limited, compared against deployed industry practices (i.e., ILDs) and active research areas (i.e., computer vision-based methods).

One successful deployment is proposed by Ma et al. [165] who installed multiple accelerometers and magnetometers respectively across of the width and length of a multi-lane highway section in California, USA. The accelerometers are used to determine the number of axles of a vehicle, and the time at which each axle passes the accelerometers. The vehicle's magnetic signature, captured by the magnetometers, is then used to infer the vehicle speed, and spacing between each of the axles (thus classifying the vehicle). The system achieves an impressive 99% vehicle classification accuracy, validated by cameras and an upstream weigh-in-motion system. A similar deployment is presented by Rivas et al. [222] who installed four accelerometers along one side of a single-lane road. The authors found that, perhaps unsurprisingly, their algorithms

worked best for those vehicles travelling on the side of the road adjacent to the sensors.

### 2.1.2 Time Series Analysis

As an SHM system might collect structural health data over a structure's life-span, methods that are designed to analyse or integrate with SHM systems are usually built upon the extensive field of time series analysis. One of the earliest time series problems is forecasting; that is, predicting the next data point in a time series, given a set of previous data points. Forecasting problems have a number of applications including, stock prices [13], water quality [71], and household energy consumption prediction [41].

Typically, forecasting problems are solved via an auto-regressive moving-average (ARMA) process that assumes each data point in a time series is the linear combination of the previous data points and errors (i.e., the difference between previous predictions and true values). Specifically, an ARMA process models a time series with value $y_t$ and error $\epsilon_t$ (assumed to be Guassian white noise) at time $t$ according to,

$$y_t = \phi_0 + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_q y_{t-q} + \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \cdots - \theta_r \epsilon_{t-r}. \tag{2.1}$$

The model orders $q$ and $r$ and parameters $\phi_i$ for $i = 0,1,2,\ldots,q$ and $\theta_j$ for $j = 0,1,2,\ldots,r$ are usually tuned using a training data set. As an extension, some recent forecasting approaches have proposed hybrid methods that combine ARMA models with nonlinear classifiers, such as neural networks and support vector machines, so that non-linearities in the time series might be captured [71, 270].

A second popular class of time series problems is classification; that is, mapping an entire time series to a single classification label. Typically, these problems follow the standard machine learning paradigm: features are extracted from the time series, which are then used to train a classification algorithm. One notable application is automatic activity classification which has become an ever-growing research area, largely due to the introduction and development of wearable sensor technology (e.g., smart watches) that capture health (e.g., heart-rate) and motion time series data. Automatic activity classification systems often have important health applications, such as ensuring the user (wearer) reaches a recommended level of exercise [198] (e.g., the 150-300 minutes a week recommended by the World Health Organisation for an adult [265]), and identifying when a vulnerable (e.g., elderly) person has had an incident (e.g., a fall) [52], for example.

In the pioneering and relatively early work presented by Parkka et al. [198] (currently cited in over 100 patents), a suite of wearable sensors (35 channels in total) were considered for the task of classifying seven activities including running, walking, rowing, and sitting down from their data. The authors found that two accelerometers, one attached to a chest strap and one worn on the wrist, provided sufficiently descriptive features. A number of classifiers were trained and tested, however an automatic decision tree classifier achieved the highest classification accuracy

on a withheld test set. Other notable time series classification problems include automatic audio classification [101], speech recognition [91], and mechanical fault detection [199].

### 2.1.3 Summary of Background and Research Objectives

To summarise, SHM systems are a standard industry practice and provide continuous structural health data to those responsible for the proper maintenance of large structures. However, data collected from sensors in SHM systems is rarely used for a secondary purpose. Furthermore, typical vehicle counting systems employ specialised, bespoke sensing solutions (such as ILDs, for example), and rarely deploy sensors one would find in an SHM system, such as accelerometers. As such, we define the following research questions (RQs) to be considered in this chapter:

RQ1.1. *Can a system be developed to count the number of vehicles that travel across a bridge from data collected by an SHM system?*

RQ1.2. *Can such a system work for both limited and complex (e.g., multi-sensor) SHM setups?*

The research question numbering convention here reflects that the research questions considered in this chapter aim to address high-level research question RQ1 defined on page 6. This convention is used to number the subsequent research questions defined in the following chapters of the thesis.

To address our research questions we will develop a case study based on the Clifton Suspension Bridge (CSB) in Bristol UK. The University of Bristol and the CSB trust, who maintain the bridge, have a history of collaboration [94, 166] and thus we may easily organise and install SHM systems across this iconic piece of infrastructure. We will develop vehicle counting methods designed specifically to work with the SHM data, and their performance will be determined by comparing the estimated vehicle counts against collected ground-truth counts.

We begin by considering SHM data collected from a previous installation (trial 1) on the CSB, which motivates our work and enables the initial development of our vehicle counting system. However, due to inaccuracies in the ground-truth vehicle counts (provided by toll barriers), structural health data from a second, smaller installation (trial 2) is collected with accurate ground-truth vehicle counts. Our system is then further developed and tested with the trial 2 data.

The rest of this chapter is organised as follows. In Section 2.2 the SHM-system deployed in trial 1 is described. In Section 2.3 the data collected from the trial is described and pre-processed. Our first vehicle-counting method, developed with the trial 1 data is described, tested, and discussed respectively in Sections 2.4, 2.5, and 2.6. Similarly, trial 2 and the steps taken to process the data collected during the trial are respectively described in Sections 2.7 and 2.8. In Section 2.9 we consider the performance of the method developed with the trial 1 data, on the trial 2 data. Then the method is improved, tested, and discussed in Sections 2.10, 2.11, and 2.12

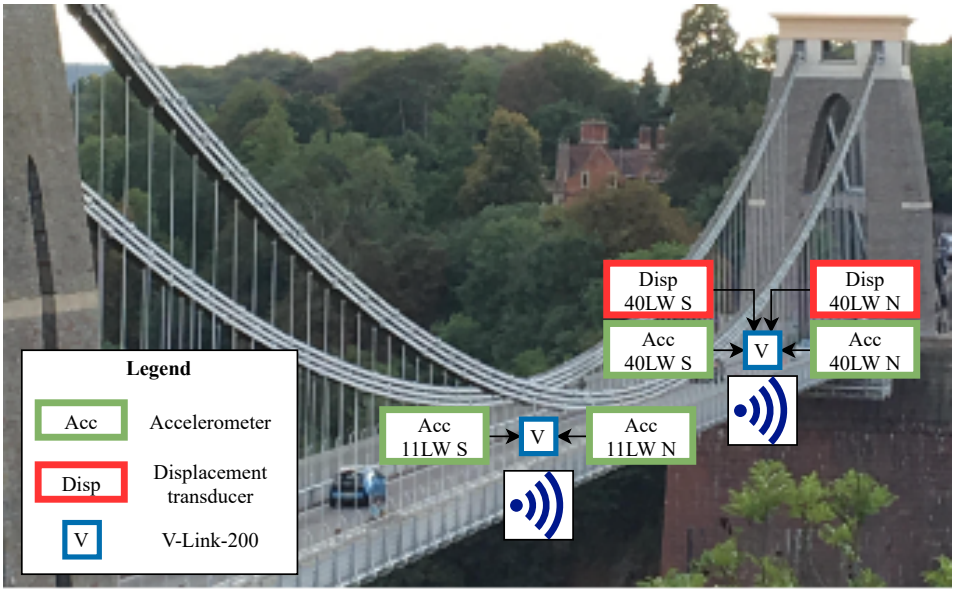



FIGURE 2.3. Sensor deployment in the CSB trial. The Clifton end is in the foreground and the Leigh Woods end is furthest away from the viewer. The camera is pointing in a broadly westerly direction.

respectively. A justification of our methods is provided in Section 2.13 before finally the chapter is concluded in Section 2.14.

## 2.2 Trial 1 Environment

We begin our work by describing the initial trial (trial 1) undertaken by Gunner et al. [94]. In 2017, a WSN-based SHM system was deployed on the CSB to investigate the rapid deployment of SHM systems. During the two-month study, the bridge's structural response was captured and displayed via an interactive dashboard. The study aimed to investigate the following:

1. Can a WSN-based SHM system be deployed quickly for short-term and ad-hoc bridge monitoring? and

2. Can the SHM system integrate with existing open-source software tools and data management systems?

The CSB is an iconic structure that stands 75 m above the River Avon and spans 214 m across the Avon Gorge joining Clifton to Leigh Woods. Since its opening in 1894, the bridge has proven to be an important piece of infrastructure, these days carrying up to 20,000 vehicles every day [94]. In addition to providing a key route in and out of the city of Bristol, the history and location (providing views of the city) of the bridge attracts tourists who significantly increase pedestrian flow. The bridge has one lane in each direction that is accessible to cyclists. A 20 mph speed limit

is imposed on the bridge and so it should take at least 24 s to drive across the entire span. There is a protected footpath on each side of the bridge for pedestrians.

A toll barrier at each end of the CSB records when a vehicle enters the bridge, truncated to the nearest minute. Vehicles travelling towards Leigh Woods pass through the toll barrier at the Clifton end of the bridge and vice versa. There is no information available on when cyclists and pedestrians enter or leave the bridge.

In total, the deployed WSN consisted of two Lord Micro Strain V-Link-200 communication nodes [163] that collect and process data from connected sensors (i.e., accelerometers and displacement transducers), before wirelessly sending their data to a personal onsite computer. Each V-Link-200 node samples the sensors at a rate of 64 Hz, or 4,000 readings per minute, a rate chosen to preserve battery life.

Due to time and funding constraints, only six sensors, in three pairs, were deployed during the study. Each pair of sensors straddles the width of the bridge deck so that both vertical and torsional displacement can be measured. Thus, each sensor is identified by the vertical iron suspension rod it is closest to, counting from the middle of the span, and whether it is on the north or south (N or S) side of the bridge.

A pair of Lord Microstrain G-Link-200 accelerometers [161], connected to a V-Link-200 node, were installed 26.8 m from the bridge's midpoint (rod 11LW) as Macdonald identified that all deck vibrations modes below 3 Hz can be detected there [166]. In addition, a pair of G-Link-200 accelerometers and a pair of displacement transducers, connected to a V-Link-200 node, were installed at the Leigh Woods End of the Bridge (rod 40LW). This location was chosen as it was believed that it would show the greatest structural response to vehicles entering and leaving the bridge. The accelerometers have a range of $10\,\mathrm{m\,s^{-2}}$ and a resolution of $5.2 \times 10^{-5}\,\mathrm{m\,s^{-2}}$, and the transducers have a range of 0.5 m and a resolution of $1 \times 10^{-6}$ m.

The dashboard produced in the trial, see Figure 2.4, is built upon the four main components that relay sensor data from the bridge to the end user at a remote computer. The bespoke system integrated the following open-source software tools, application programming interfaces (APIs), and data management systems:

- the Lord Microstrain Communication Libarary [162] used to configure the V-Link-200 nodes;
- a message broker, Kafka [12], used to integrate data sources with different formats, sizes and message protocols;
- a time series database, InfluxDB [133], to provide long-term data storage; and,
- a web-based data visualisation interface, Grafana [90], to display the data.

A comprehensive description of the trial, and each component of the deployed WSN-based SHM system is provided by Gunner et al. [94].

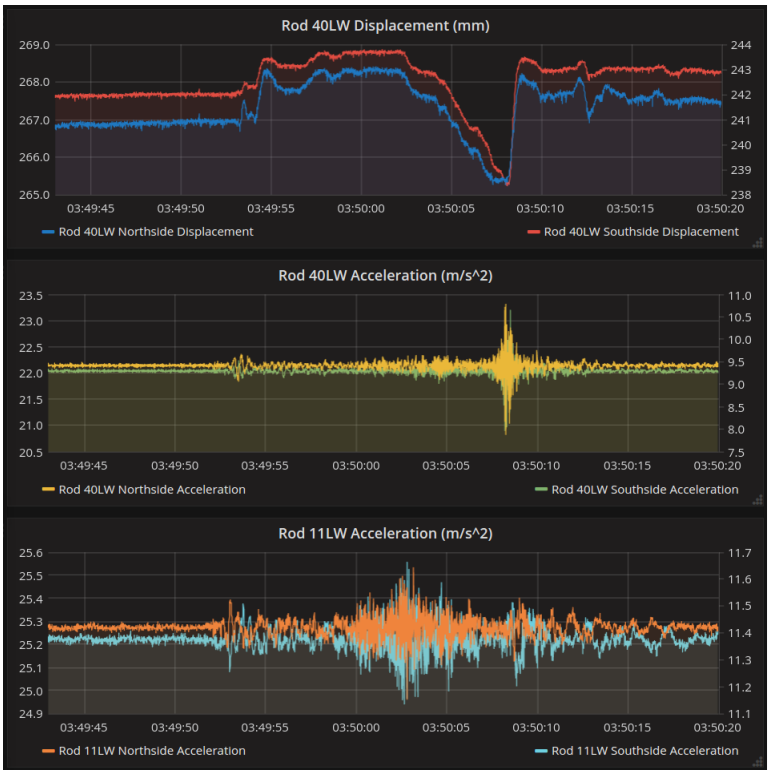



FIGURE 2.4. The SHM dashboard produced from trial 1. Acceleration and displacement data captured by the six sensors in the WSN: the CSB's structural response to a single vehicle entering the bridge at the Clifton is displayed. Figure reproduced from Gunner et al. [94].

## 2.3 Data Sources and Preparation

We now proceed to describe, in greater detail, the data sources obtained from trial 1 and the preparation steps undertaken in order to develop our vehicle counting method. In order to address research question RQ1.2 (see page 20), we only consider the accelerometers (rather than the displacement transducers) since they are cheap to procure, simple to install, and used in a number of previous SHM (both single and multi-sensor) setups [78, 143, 273, 276].

### Toll Barriers

The toll barriers at each end of the CSB record when a vehicle enters the bridge, truncated to the nearest minute. That is, a vehicle arriving at $t :=$ YYYYDDMM hh:mm:ss is attributed to

$$T(t) := \text{YYYYDDMM hh:mm:00}. \tag{2.2}$$

Minute by minute, we thus have counts for the number of vehicles that enter each end of the bridge.

**Accelerometers**

Figure 2.5 shows two-minute windows of the bridge's structural health response to vehicles travelling across the bridge in each direction, where the directions are determined by the toll barrier data. The figure shows that accelerometers 11LW N and 11LW S are featureless under loading for both directions of travel, whereas, a clear response is observed in accelerometers 40LW N and 40LW S. There is no obvious difference between the responses recorded by the 40LW N and 40LW S accelerometers, therefore, in the following work, we will only consider the acceleration recorded by 40LW N, which we denote by $a_t$ at time $t$.

### 2.3.1 Segmentation and Matching of Accelerometer Data

In order to develop a method to count vehicles, the accelerometer data is segmented into windows with corresponding ground-truth vehicle counts.

Initially, the accelerometer windows are segmented into one-minute windows beginning at time $T(t)$ with a corresponding vehicle count provided by the toll barrier data. However, consider the following situation. A vehicle enters the bridge at time $t_1$, and is still travelling across the bridge at time $t_2$, such that $T(t_1) < T(t_2)$. In this case, the vehicle may cause a structural response in the one-minute window after that in which it enters the bridge.

To ensure that all structural responses of a vehicle entering on a given minute are captured, we group that minute's toll barrier data with the accelerometer data for that and the following minute, resulting in a two-minute window.

Further, to prevent confusion with the structural response caused by vehicles that enter the bridge either before or after the selected minute, we take forward for analysis only those minutes for which the immediately preceding and succeeding toll barrier counts are zero. Of course, this unfortunately means that our analysis can consider only relatively quiet times of day — a limitation that is addressed in trial 2. The accelerometer has a sampling interval of $\Delta t = 0.015\,\text{s}$, and so each two-minute window of accelerometer data can be expressed as $a_i$ for $i = 1,2,\ldots,m$, where $m = 8{,}000$.

### 2.3.2 Instantaneous Amplitude Envelope Computation

To ensure that each window of accelerometer data has a similar minimum and maximum amplitude, a normalistion is performed on each window in the form

$$\tilde{a}_i = 2\left[\frac{a_i - \min(a_i)}{\max(a_i) - \min(a_i)}\right] - 1, \tag{2.3}$$

to give normalised readings $\tilde{a}_i$ between -1 and 1.

We now summarise the accelerometer in a smoothed form known as the instantaneous amplitude envelope (IAE) [200]. We consider the three following methods for computing the IAE, see Figure 2.6.

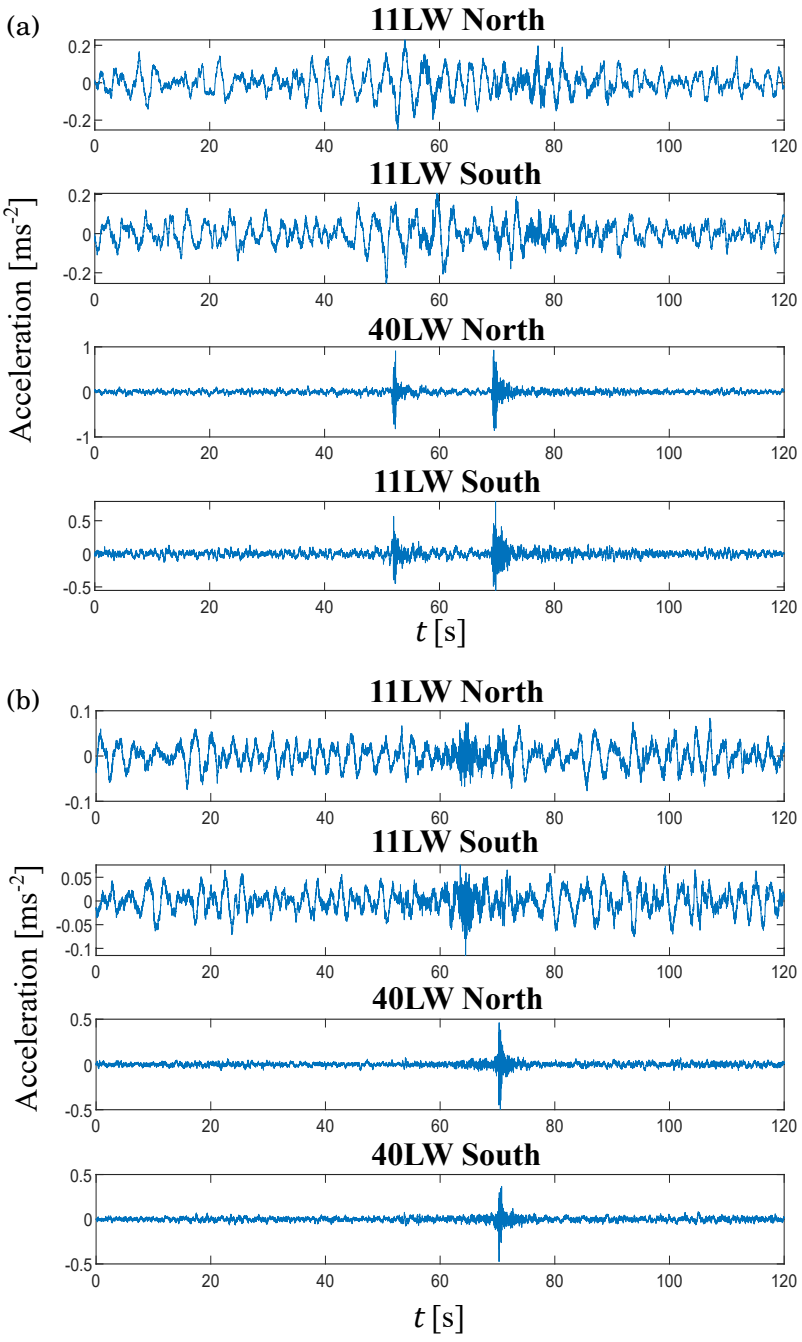



FIGURE 2.5. Accelerometer readings for two-minute windows where (a) two vehicles enter from the Leigh Woods end and (b) a single vehicle enters from the Clifton end (b).

1. **Peak to peak.** The accelerometer data is replaced by a cubic spline that interpolates local maxima, provided that they are separated by at least $d$ time steps. Here, $d = 5$ was observed

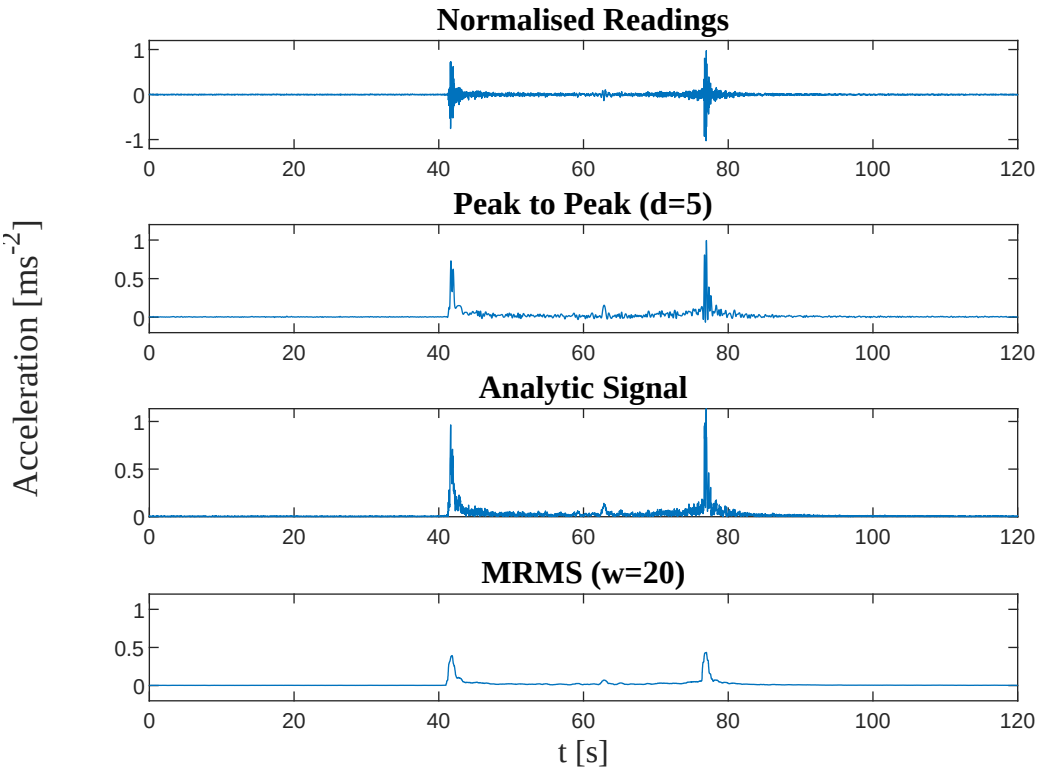



FIGURE 2.6. Normalised accelerometer data for a two-minute window and the corresponding instantaneous amplitude envelope (IAE) computed by the three methods.

to achieve the required level of smoothing, but the computed envelope is not necessarily positive, so the method was not taken forward.

2. **Analytic signal.** The IAE is found from the magnitude of the discrete-time analytic signal [216]. The computed envelopes are less smooth than we require, so the method is also discarded.

3. **Moving root mean square (MRMS).** MRMS computes the root mean square from a window of width $2w$ that slides across the accelerometer data. Thus, smoothed readings $\hat{a}_i$ are found by

$$\hat{a}_i = \sqrt{\frac{1}{2w-1}\sum_{i-w_i}^{i+w_i} \tilde{a}_i}\,, \tag{2.4}$$

where $w_i = \min(i, w, m-i)$. The computed envelope is thus strictly positive, and a choice of $w = 20$ (window with of 0.6 s) seems to provide an appropriate level of smoothing. The MRMS method is thus used to smooth the trial 1 accelerometer windows.

TABLE 2.1. Labelled data set of 30 training windows and 120 validation windows.

| $n_{\mathrm{LW}}$ | $n_{\mathrm{C}}$ | Training Windows | Validation Windows |
|---|---|---|---|
| 1 | 0 | 15 | 20 |
| 2 | 0 | 0 | 20 |
| 3 | 0 | 0 | 20 |
| 0 | 1 | 15 | 20 |
| 0 | 2 | 0 | 20 |
| 0 | 3 | 0 | 20 |

### 2.3.3 Labelled Data Set

We begin by considering light traffic conditions, where only a single vehicle passes the accelerometer at any given moment (i.e., two vehicles do not simultaneously pass the accelerometer in different directions). Thus, the toll barrier data is inspected to find one-minute windows when one, two, or three vehicles enter the bridge at one end, and no vehicles enter at the other end. Further, only windows for which no vehicles enter at either end in the precceeding and the following minute are considered. A two-minute window of IAE accelerometer data, corresponding to the selected minute, and that which follows, is then extracted for analysis. Such quiet traffic conditions feature mostly at night-time, so the search was restricted to between 10 pm and 7 am each night. In addition, this selection reduces the likelihood of structural responses to pedestrians or cyclists, whose ground-truth counts are not known.

In total, we built a training set of 30 windows and a validation set of 120 windows, as shown in Table 2.1. Here the counts from the Leigh Woods and Clifton toll barriers are denoted by $n_{\mathrm{LW}}$ and $n_{\mathrm{C}}$ respectively. The training windows are used to tune the method in Section 2.4.4 and the validation windows are used to test its accuracy in Section 2.5.

The method identifies the structural response to each vehicle, and thus automatically counts vehicles travelling across the bridge in each window. The method is developed to work both when the response is clear for each vehicle, as shown in Figures 2.5 and 2.6, and when the bridge is busier.

In summary, each selected window is equipped with the count for each toll barrier $n_{\mathrm{LW}}$ and $n_{\mathrm{C}}$, and a smoothed IAE time series $\hat{a}_i$ for $i = 1, 2, \ldots, m$. For reference purposes, the time $T(t)$ (YYYMMDD hh:mm:00) at which the window commences is also recorded.

## 2.4 Method

We now proceed to develop our vehicle counting method. The bridge will exhibit a damped oscillatory response to loading that is multi-modal and complex. The general idea is to seek the optimal fit of the IAE with a *simplified* closed-form function that captures the number of vehicles crossing the bridge.

### 2.4.1 Basis Function for a Single Vehicle

We aim to fit a function to the smoothed IAE time series, $\hat{a}_i$ for $i = 1,2,\ldots,m$. The IAE begins at time $t_0$ and the $i^{\text{th}}$ reading is recorded at $t = t_0 + i\Delta t$. This provides a mapping from the discrete time index $i$ of the IAE to the continuous time $t$.

We use the basis function $f(t;\boldsymbol{\theta})$ to fit the IAE of the bridge's structural response under the loading of a single vehicle. We assume that the function is smooth and is parameterised by $p$ parameters $\boldsymbol{\theta} = [\theta_1,\theta_2,\ldots,\theta_p]$. In addition, we include a scaling parameter $\gamma$ to account for different envelope amplitudes, and a shift parameter $\tau$ that locates the vehicle within the two-minute window. Thus, we fit

$$g(t;\boldsymbol{\theta},\gamma,\tau) = \gamma f(t-\tau;\boldsymbol{\theta}). \tag{2.5}$$

### 2.4.2 Multiple Vehicles

Under a linear superposition assumption, we model the structural response to $n$ vehicles by the sum of $n$ scaled and shifted single-vehicle basis functions in the form

$$g_n(t) = \sum_{j=1}^{n} g(t;\boldsymbol{\theta}_j,\gamma_j,\tau_j) = \sum_{j=1}^{n} \gamma_j f(t-\tau_j;\boldsymbol{\theta}_j). \tag{2.6}$$

The function $g_n(t)$ is parameterised by $n(p+2)$ parameters $\boldsymbol{\theta}_j = [\theta_{j_1},\theta_{j_2},\ldots,\ldots\theta_{j_p}]$, $\gamma_j$, and $\tau_j$ for $j = 1,2,\ldots,n$.

The chosen estimate $n^*$ for the number of vehicles, is then found by a putative 'trial' number of vehicles $n$ that minimises the mean square error (MSE),

$$\text{MSE} = \frac{1}{m}\sum_{i=1}^{m}\left(\hat{a}_i - g_n(t_0 + i\Delta t)\right)^2, \tag{2.7}$$

between the fitted function and the IAE.

The parameters $\boldsymbol{\theta}_j$, $\gamma_j$, and $\tau_j$ are found in sequence. That is, the parameters that describe a single vehicle ($\boldsymbol{\theta}_1$, $\gamma_1$, and $\tau_1$) are solved for and fixed. Then, the parameters that describe a second vehicle ($\boldsymbol{\theta}_2$, $\gamma_2$, and $\tau_2$) are solved for and fixed, and so on. The procedure continues to add vehicles until the information criterion described in the following section tells it to stop. At this point, the procedure has arrived at its final estimate $n^*$.

### 2.4.3 Akaike Information Criterion and Auto-correlation

To avoid over-fitting, the Akaike information criterion (AIC) [6] is used to penalise the goodness of fit of $g_n(t)$ by the number of parameters $n(p+2)$. The AIC is defined by

$$2n(p+2) - 2\log\hat{\ell}, \tag{2.8}$$

where $\log\hat{\ell}$ is the maximum value of the log-likelihood function of the fit $g_n(t)$ given the IAE. The chosen estimate $n^*$ is selected by the function that minimises the AIC.

Standard likelihood techniques make the assumption that the error terms between the fitted function and the IAE, $\epsilon_i = \hat{a}_i - g_n(t_0 + i\Delta t)$ for $i = 1,2,\dots m$, are identically and independently distributed (IID). However, this is not an appropriate assumption when dealing with time series data, as consecutive errors are likely to be correlated.

Instead, we assume that the errors follow an auto-regressive (AR) process (i.e., an ARMA process with order $r = 0$, see Section 2.1.2). Each error is modelled as the weighted sum of the previous $q$ errors plus a noise term, which is assumed to be normally IID with zero mean and variance $\sigma_\epsilon^2$ in the form

$$\epsilon_i = \sum_{k=0}^{q} \phi_k \epsilon_{i-k} + \mathcal{N}(0, \sigma_\epsilon), \tag{2.9}$$

where $|\phi_k| < 1$ for $k = 0,1,\dots,q$. The parameter $q$ is known as the order of the AR model, which is denoted as AR($q$). The AR coefficients (weights) define how correlated each error $\epsilon_i$ is with the $k^{\text{th}}$ error before ($\epsilon_{i-k}$), known as the $k^{\text{th}}$ lag. The choice of model order $q$ is described in Section 2.4.6.

The weights are determined by the Partial Auto-correlation Function (PACF) [23]. In practice, software routines such as the `parcorr` function in Matlab can be employed to compute the PACF. The $i^{\text{th}}$ error from the AR process is then defined by

$$\hat{\epsilon}_i = \epsilon_i - \sum_{k=0}^{q} \phi_k \epsilon_{i-k}, \tag{2.10}$$

where $\hat{\epsilon}_i \sim \mathcal{N}(0, \sigma_e)$.

Note the two-minute windows contain periods when there are no vehicles on the bridge, and therefore, there is no structural response, only background noise. So that we only consider data above the noise floor, a threshold $\delta$ is set. We then consider only those $\bar{m} < m$ errors $\bar{\epsilon}$ for which $\hat{a}_i > \delta$ for $i = 1,2,\dots,\bar{m}$. In this work, the threshold $\delta$ is set to $0.05\,\mathrm{m\,s^{-2}}$, by visual inspection.

The errors $\bar{\epsilon}_i$ are also normally distributed with zero mean but with variance $\sigma_{\bar{\epsilon}}^2$, and thus

$$p(\bar{\epsilon}_i; \sigma_{\bar{\epsilon}}^2) = \frac{1}{\sqrt{2\pi\sigma_{\bar{\epsilon}}^2}} \exp\left(-\frac{\bar{\epsilon}_i^2}{2\sigma_{\bar{\epsilon}}^2}\right). \tag{2.11}$$

As the errors $\bar{\epsilon}_i$ for $i = 1,2,\dots,\bar{m}$ are now IID, we may proceed with the standard likelihood calculation, which is now provided for the benefit of the reader.

The likelihood of the fitted function $g_n(t)$ given the IAE is given by

$$\ell(\bar{\epsilon}_1, \bar{\epsilon}_2, \dots, \bar{\epsilon}_{\bar{m}}; \sigma_{\bar{\epsilon}}^2) = \prod_{i=1}^{\bar{m}} p(\bar{\epsilon}_i; \sigma_{\bar{\epsilon}}^2) = \left(2\pi\sigma_{\bar{\epsilon}}^2\right)^{-\frac{\bar{m}}{2}} \exp\left(-\frac{\sum_{i=1}^{\bar{m}} \bar{\epsilon}_i^2}{2\sigma_{\bar{\epsilon}}^2}\right), \tag{2.12}$$

and thus the log-likehood is computed by

$$\log \ell(\bar{\epsilon}_1, \bar{\epsilon}_2, \dots, \bar{\epsilon}_{\bar{m}}; \sigma_{\bar{\epsilon}}^2) = -\frac{\bar{m}}{2}\log 2\pi - \frac{\bar{m}}{2}\log \sigma_{\bar{\epsilon}}^2 - \frac{1}{2\sigma_{\bar{\epsilon}}^2} \sum_{i=1}^{\bar{m}} \bar{\epsilon}_i^2. \tag{2.13}$$

TABLE 2.2. MMSE for the log-normal, gamma and Weibull fits to single-vehicle training windows.

| Basis function | MMSE Leigh Woods [ms$^{-2}$] | MMSE Clifton [ms$^{-2}$] | MMSE Total [ms$^{-2}$] |
|---|---|---|---|
| log-normal | 0.0005 | 0.0012 | **0.0008** |
| gamma | 0.0006 | 0.0012 | 0.0009 |
| Weibull | 0.0012 | 0.0022 | 0.0019 |

The maximum value of the log-likelihood $\log\hat{\ell}$ is given by $\log\ell(\bar{\epsilon}_1,\bar{\epsilon}_2,\ldots,\bar{\epsilon}_{\bar{m}};\hat{\sigma}^2_{\bar{\epsilon}})$, where the maximum likelihood estimator of the variance

$$\hat{\sigma}^2_{\bar{\epsilon}} = \frac{\left(\sum_i^{\bar{m}} \bar{\epsilon}_i^2\right)}{\bar{m}} \tag{2.14}$$

is the solution to

$$\frac{\partial \ell}{\partial \sigma_{\bar{\epsilon}}} = 0. \tag{2.15}$$

By substituting Equation 2.14 into Equation 2.13 we derive

$$\begin{aligned} \log\hat{\ell} &= -\frac{\bar{m}}{2}\log 2\pi - \frac{\bar{m}}{2}\log\left(\sum_i^{\bar{m}} \bar{\epsilon}_i^2\right) + \frac{\bar{m}}{2}\log(\bar{m}) - \frac{\bar{m}}{2} \\ &= -\frac{\bar{m}}{2}\log\left(\sum_i^{\bar{m}} \bar{\epsilon}_i^2\right) + c, \end{aligned} \tag{2.16}$$

for constant $c$.

Substituting Equation 2.16 into Equation 2.8 gives the quantity to be minimised (ignoring the constant term). To summarise, in doing so we find the estimated number of vehicles that cross the bridge, from

$$n^* = \arg\min_n \left[2n(p+2) + \bar{m}\log\left(\sum_i^{\bar{m}} \bar{\epsilon}_i^2\right)\right]. \tag{2.17}$$

### 2.4.4 Training the Method

We now proceed to train the method described in the previous sections. Specifically, we consider the training set of single-vehicle two-minute windows (see Table 2.1) to determine the basis function $f(t;\boldsymbol{\theta})$ and its statistics, and the AR coefficient $q$. The restriction to single-vehicle windows is in order to factor out additional problems if the linear superposition assumption turned out to be poor.

### 2.4.5 Choice of Basis Function

We consider the basis function $f(t;\boldsymbol{\theta})$ that models the IAE time series of the bridge's structural response to a single vehicle entering the bridge at either end. A good basis function will give a small mean square error, defined in Equation 2.7 with $n = 1$.

Each selected IAE summarises the damped oscillatory response of the bridge to dynamic loading causes by a single passing vehicle. Thus, the overall shape should be one with a rapid 'attack' and a gradual (probably exponential) decay. Inspired by probability density functions, we considered the following three forms for the basis function, each of which has two parameters ($p = 2$) that control the 'location' and 'dispersion' and thus the overall shape.

1. log-normal:

$$f(t,\boldsymbol{\theta}) = \frac{1}{t\sigma\sqrt{2\pi}} \exp\left(-\frac{\big(\log(t)-\mu\big)^2}{2\sigma^2}\right), \quad \boldsymbol{\theta} = [\mu,\sigma]. \tag{2.18}$$

2. gamma:

$$f(t,\boldsymbol{\theta}) = \frac{\beta^\alpha}{\Gamma(\alpha)} t^{\alpha-1} \exp(\beta t), \quad \boldsymbol{\theta} = [\alpha,\beta], \tag{2.19}$$

   where $\Gamma$ denotes the Gamma function [15].

3. Weibull:

$$f(t,\boldsymbol{\theta}) = \frac{k}{\lambda}\left(\frac{t}{\lambda}\right)^{k-1} \exp\left(-\frac{t^k}{\lambda^k}\right), \quad \boldsymbol{\theta} = [k,\lambda]. \tag{2.20}$$

The quality of each function is computed by calculating the optimal MSE for each of the 30 single-vehicle training windows. Table 2.2 shows the mean MSE (MMSE) for each considered basis function, i.e., the average across the 30 windows. The log-normal fit is best and so log-normal basis functions are used going forward.

We now consider whether the log-normal parameters $\mu$ and $\sigma$ are similar for each vehicle travelling across the bridge. If they were, they might be fixed in the fitting process so that only $\gamma$ and $\tau$ (see Equation 2.6) need to be determined for each vehicle.

For each of the training windows, we fit a log-normal by least squares to find the optimal parameters $\boldsymbol{\theta} = [\mu,\sigma]$, $\gamma$, and $\tau$. A scatter plot of the fitted $\mu$ and $\sigma$ parameters is shown in Figure 2.7. There is unfortunately no consistent pattern and therefore the log-normal parameters must be found independently for each vehicle going forward.

### 2.4.6 Auto-regressive Model Order

We now use the training windows to determine the AR model order $q$ described in Section 2.4.3. For each window, we fit a log-normal function and compute the errors $\epsilon_i$ and the first 30 weights $\phi_k$ for $k = 1,2,\ldots,30$. We then calculate the mean weights $\bar{\phi}_k$ averaged over the training windows.

The model order is determined by the lag for which the mean weight falls within the 95% confidence bounds, which assuming the weights are approximately normally IID with zero mean, are given by $\pm 0.96/\sqrt{m}$ [32]. Figure 2.8 shows the mean weights for each of the test lags. All of

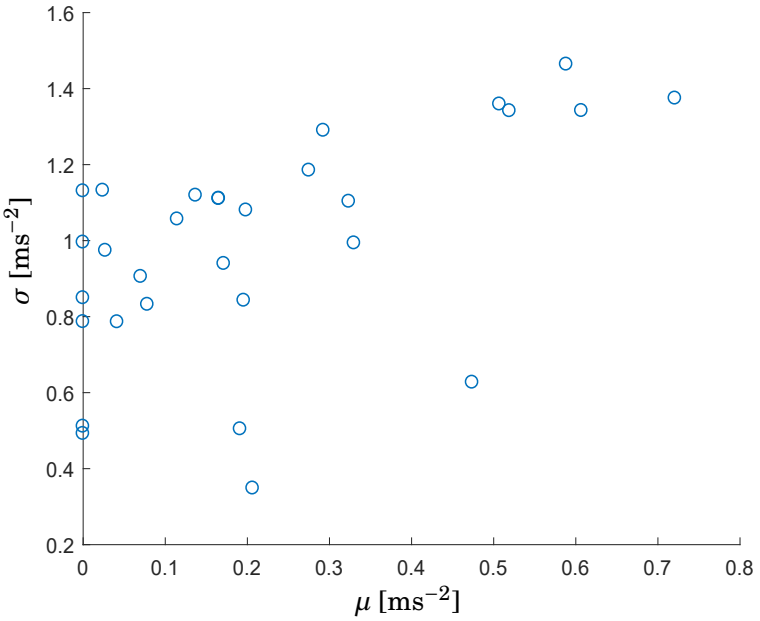



FIGURE 2.7. Scatter plot of log-normal parameters fitted to 30 training windows.

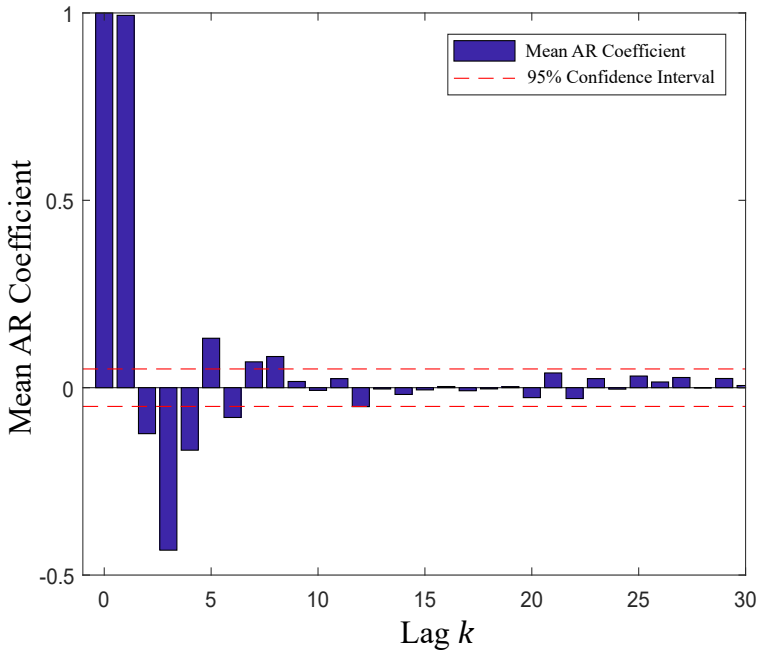



Figure 2.8: Mean AR coefficients for each lag and their 95% confidence bounds.

the significant auto-correlation is contained within the first eight lags. Therefore, the error terms $\bar{\epsilon}_i$ for $i = 1,2,\ldots,m$ are computed by an AR(8) process (see Equation 2.10).

TABLE 2.3. Toll barrier counts and the corresponding number of vehicles $n^*$, for 120 validation windows.

| $n_{\mathrm{LW}}$ | $n_{\mathrm{C}}$ | $n^*=1$ | $n^*=2$ | $n^*=3$ | $n^*=4$ |
|---|---|---|---|---|---|
| 1 | 0 | 19 | 1 | 1 | 0 |
| 2 | 0 | 0 | 18 | 2 | 0 |
| 3 | 0 | 0 | 3 | 15 | 2 |
| 0 | 1 | 13 | 4 | 3 | 0 |
| 0 | 2 | 0 | 12 | 6 | 2 |
| 0 | 3 | 0 | 2 | 12 | 6 |

The AR model order $q$ is the only system parameter learnt and fixed prior to the vehicle counting procedure. In summary, for each putative number of vehicles, new log-normal parameters ($\mu$ and $\sigma$), scales $\gamma$, and shifts $\tau$ are found by least squares. In the likelihood calculation, new sets of AR coefficients $\phi_k$ for $k=0,1,\ldots,8$ are also computed.

## 2.5 Results

We test the method by computing the estimated number $n^*$ of vehicles for 120 validation windows and compare with their ground-truth counts $n_{\mathrm{LW}}$ and $n_{\mathrm{C}}$, see Table 2.1. The results are shown in Table 2.3. Since the method does not identify the direction of travel, $n_{\mathrm{LW}}+n_{\mathrm{C}}=n^*$ corresponds to a correct prediction (although for any one validation window, only one of $n_{\mathrm{LW}}$ or $n_{\mathrm{C}}$ is non-zero). The method estimates the correct number of vehicles in 89 (74%) of the validation windows.

Figure 2.9 illustrates the method at work for two exemplar windows. In each example, the MSE decreases as the putative number of 'trial' vehicles increases and the function $g_n(t)$ becomes more complex. However, the AIC penalises the complexity of $g_n(t)$, and the number of vehicles is found correctly, by its minimisation.

## 2.6 Discussion

The overall prediction accuracy of 74% is somewhat disappointing. However, when $n_{\mathrm{C}}=0$ (vehicles only enter at the Leigh Woods end), we achieve a prediction accuracy of 87% and we thus have the beginnings of a plausible operational method. In addition, for the windows considered here, there is no evidence of bias towards over or under-counting.

In contrast, the fit to vehicles entering at the Clifton end is poor, and there is apparently a bias towards over counting. This may be explained by the exemplar window shown in Figure 2.10. A single log-normal basis function is a poor fit to the structural response in this case, because the vehicle causes two peaks in the structural response: one when it enters the bridge at the Clifton end, and a second when it reaches the accelerometer at the Leigh Woods end. The estimation attempts to fit the two peaks with two basis functions, and so counts two vehicles when only one

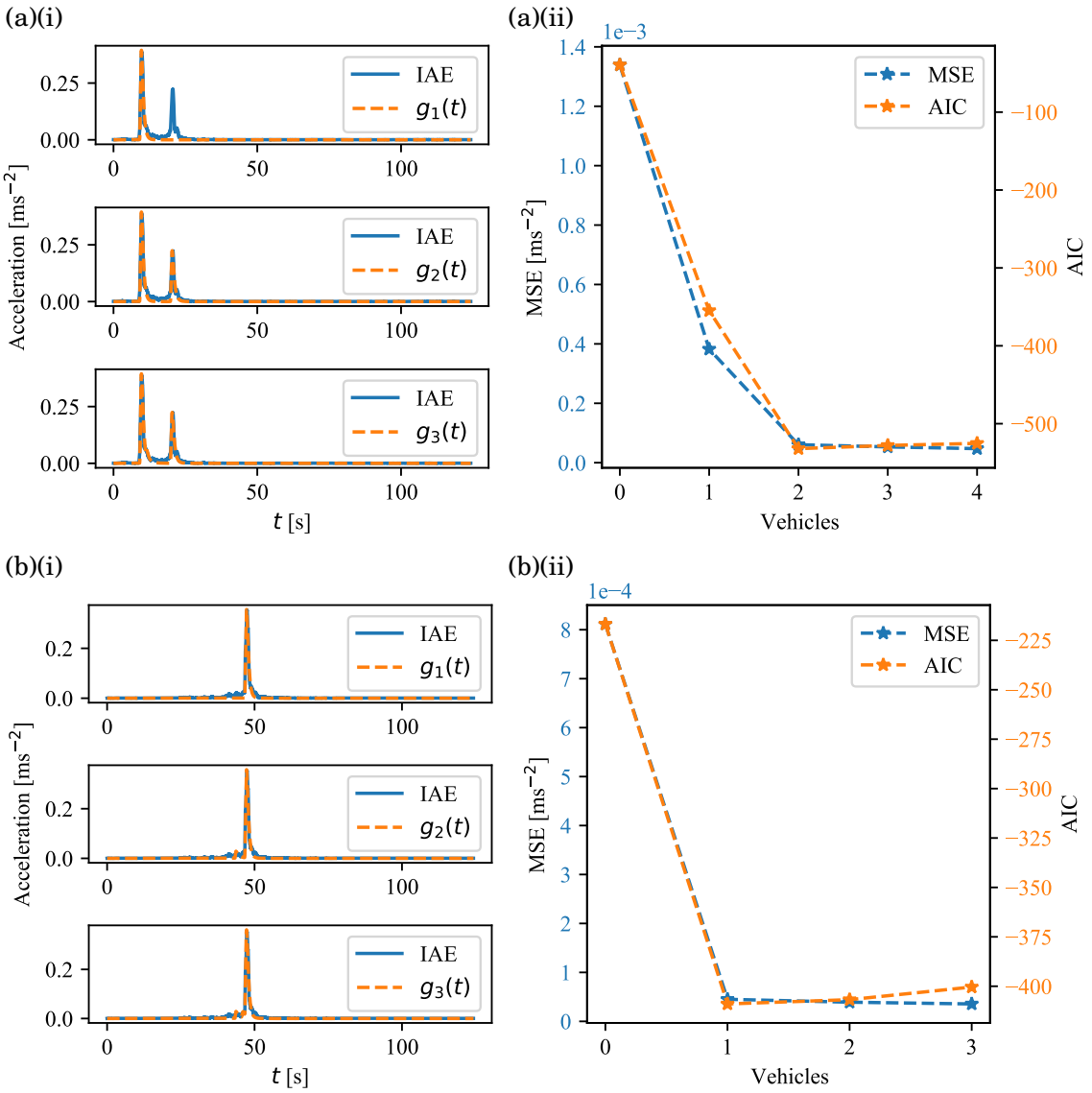



FIGURE 2.9. The vehicle estimation method in operation for two exemplar windows. (a) $n_{\mathrm{LW}} = 2$, $n_{\mathrm{C}} = 0$, (b) $n_{\mathrm{LW}} = 0$, $n_{\mathrm{C}} = 1$. Left-hand panels (i) show the superposition of one, two, and three basis functions fitted to the IAEs. Right-hand panels (ii) show the corresponding MSE and AIC for the fits.

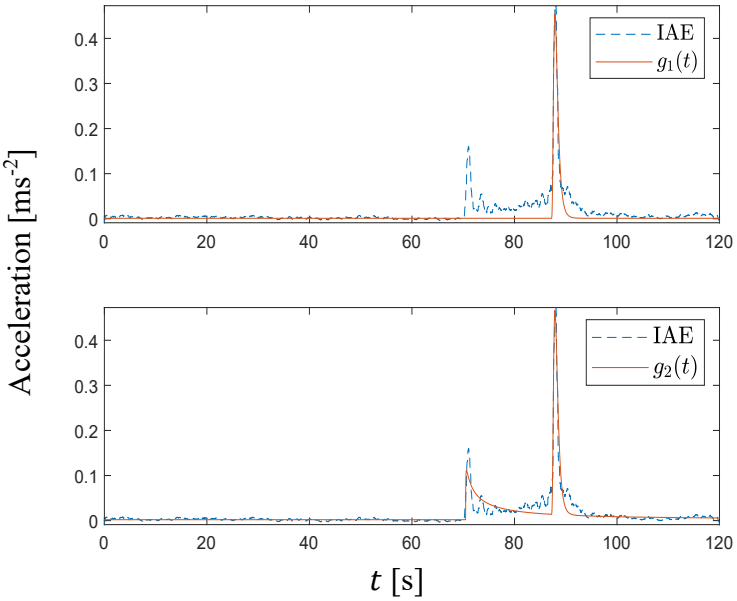



FIGURE 2.10. An example where $n_C = 1$, $n_{LW} = 0$ and $n^* = 2$. When the vehicle enters the Clifton end, there is an additional peak in advance of that when it reaches the Leigh Woods end detector.

is present. This problem does not arise for vehicles entering at the Leigh Woods end, because the entry peak is almost simultaneous with that when the accelerometer is reached. Note Table 2.2, which concerns the training set, has already indicated that the unimodal basis functions fit the Leigh Woods vehicles better than the Clifton ones.

Due to the toll barrier data truncation (see Equation 2.2), the method developed so far in this chapter has been only tested on a restricted validation set with low ground-truth vehicle counts and vehicles travelling in the same direction, that may be manually verified. However, to develop an operational vehicle counting capability we now aim to investigate to what degree our method can identify individual vehicles when there are many different overlapping structural responses, perhaps including those from cyclists and pedestrians too, and indeed whether the linear superposition holds in congested traffic flow conditions.

As such, we conducted a second trial (trial 2) on the CSB to capture its structural response under loading, with accurate, manually recorded ground-truth vehicle counts used instead of the toll barrier data.

## 2.7 Trial 2 Environment

Although a multi-sensor SHM system was deployed in trial 1, our vehicle counting method relies only on the data collected from a single accelerometer (40LW N). Therefore, for simplicity and

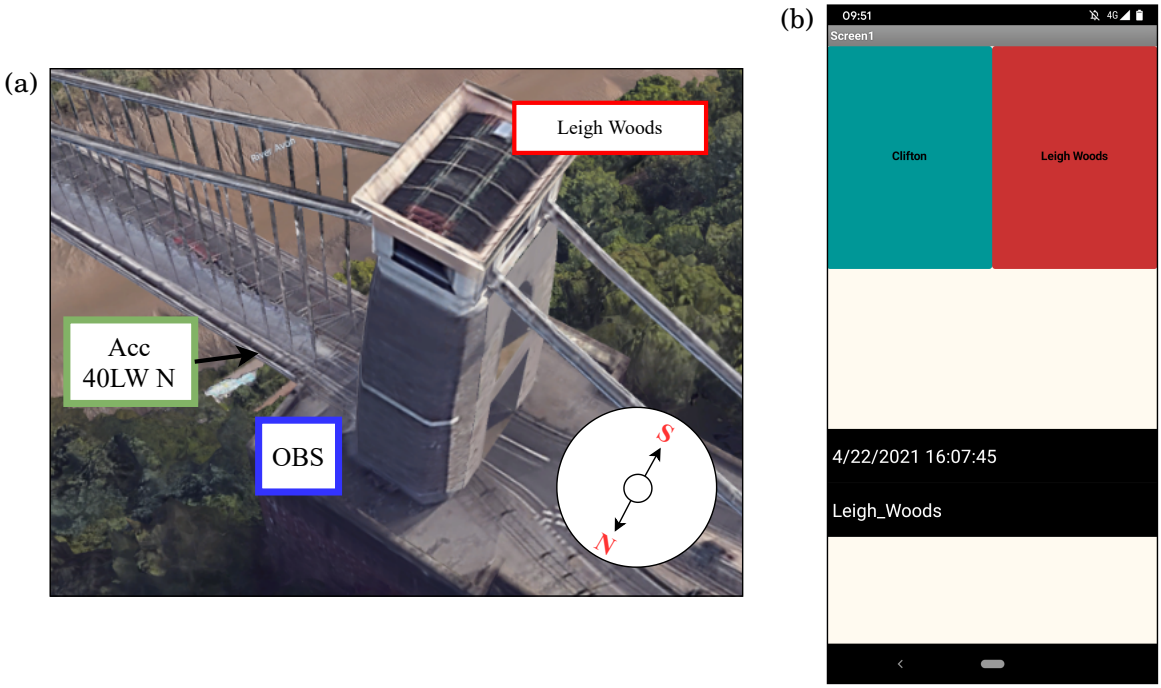



FIGURE 2.11. Trial 2 setup. (a) A single accelerometer was installed at rod 40LW N (green). An observer (OBS, blue) stood beside the accelerometer and used a mobile application (b) to accurately record the time and direction each vehicle passes the accelerometer while travelling across the bridge. Panel (a) is developed from Google Earth imagery [87].

continuity, in trial 2, we considered a simpler single-sensor setup; namely, a single accelerometer installed on rod 40LW N.

In contrast to the real-time data provided wirelessly by the WSN-based SHM system considered in trial 1, the data captured in trial 2 was stored locally on a storage device within the sensor and downloaded to a personal laptop after the trial. The accelerometer used in trial 2 is identical to those used in the trial 1 (a Lord Microstrain G-link-300 accelerometer).

In order to undertake the trial, a method statement and risk assessment was provided by our team (all from the University of Bristol) to the CSB trust who permitted us to perform the trial over a single day from 9am to 5pm.

To record accurate ground-truth vehicle counts, a single observer was positioned with direct line of site to the accelerometer. The observer position, see Figure 2.11(a), approved by the CSB trust, was protected by metal fencing and chosen to minimise pedestrian disruption. As a vehicle passed the accelerometer, the observer recorded the time and direction of the vehicle via a bespoke mobile application, developed with the MIT app inventor service [171], see Figure 2.11(b).

The application interfaces two buttons; one labelled ‘Clifton’ and one labelled ‘Leigh Woods’, corresponding to the two directions of travel. At the moment the observer presses either button,

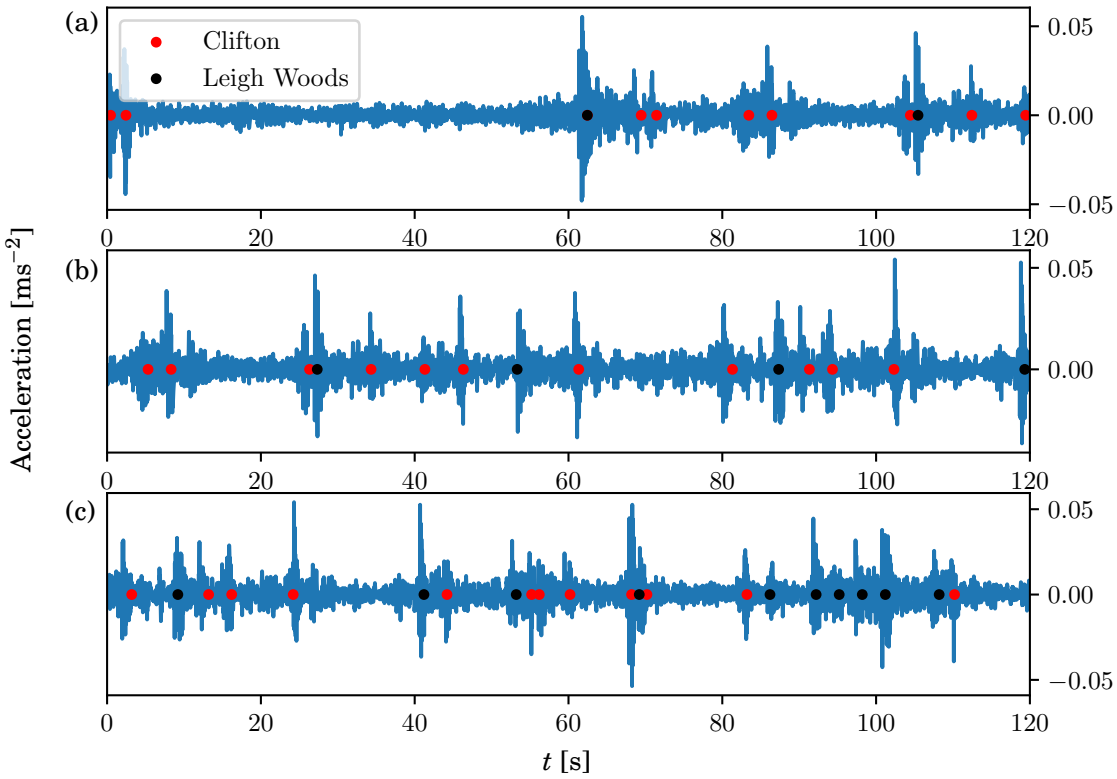



FIGURE 2.12. Three exemplar two-minute accelerometer windows captured in trial 2. (a) $n_{\mathrm{LW}} = 2$, $n_{\mathrm{C}} = 10$, (b) $n_{\mathrm{LW}} = 4$, $n_{\mathrm{C}} = 11$, and (c) $n_{\mathrm{LW}} = 10$, $n_{\mathrm{C}} = 12$. The dots depict the timestamp and direction of each vehicle that passes the accelerometer recorded by the observer.

its label and a timestamp (given by the mobile phone's internal clock) is recorded in an online Google spreadsheet [86]. The application also displays the last recorded vehicle, to reassure the observer that the application is operating correctly. The only requirement for the application is an android phone, however a mobile signal is needed to send the vehicle counts to the Google spreadsheet when in operation. In total, five different trial team members acted as the observer during the trial, and swapped every hour to maintain reasonable workload and thus good quality data.

## 2.8 Data Sources and Preparation

We now describe the data sources captured during trial 2 which we will take forward to further develop and test our vehicle counting method in realistic traffic conditions.

**Accelerometer Data**

The single accelerometer is configured as in trial 1. As such, the accelerometer has a sampling frequency of 64 Hz, a range of $10\,\mathrm{m\,s^{-2}}$, and a resolution of $5.2 \times 10^{-5}\,\mathrm{m\,s^{-2}}$. The accelerometer data is timestamped with a resolution of $1 \times 10^{-6}$ s.

**Vehicle Timestamps**

For each vehicle that travels across the CSB, the mobile application used by the observer provides the vehicle's direction of travel (entering from either the Clifton or Leigh Woods end of the bridge) and the time at which the vehicle passes the accelerometer to the nearest second. In total, 1,564 vehicles were recorded from the Leigh Woods end of the bridge and 1,536 vehicles were recorded from the Clifton end during the trial.

### 2.8.1 Pre-processing, Segmentation, and Matching of Trial 2 Accelerometer Data

In order to apply the method described in the previous sections (developed with data collected during the trial 1), we now segment the trial 2 accelerometer data into two-minute windows, as described in Section 2.3.1. For each window, the total number of recorded vehicles whose timestamp falls within the window is attributed as the ground-truth vehicle count, see Figure 2.12. In total, 173 two-minute windows are constructed from the trial 2 data.

Pedestrians, cyclists, and the congested traffic conditions present during trial 2 induce noise into the accelerometer readings, see Figure 2.13. Therefore, a low and high pass frequency filter is applied to the accelerometer data to in order remove the noise: those frequency components less than 11 Hz and greater than 50 Hz are removed.

Recall that to avoid confusion with the structural responses caused by uncounted vehicles, two-minute windows segmented from the original trial were only taken forward for analysis when the immediately preceding and succeeding toll barrier counts are zero, see page 23. Unfortunately, there are no such isolated windows captured in trial 2, where congested traffic conditions are (intentionally) considered. However, the larger vehicle counts (compared against the original trial) and accurate timestamps captured in trial 2 mean that the majority of the CSB's structural response in any given two-minute window is likely to be caused by a vehicle attributed to the window's ground-truth vehicle count. In Section 2.11.1, we consider the affect of using longer accelerometer windows that further reduce any confusion in the structural response.

## 2.9 Trial 1 Method Applied to Trial 2 Windows

We now apply the method developed and tested in Section 2.4 to the two-minute windows segmented from the trial 2 data. The method is not re-tuned, that is, we do not consider a new basis function or AR model order, and all 173 windows are used as a validation set. The IAEs are

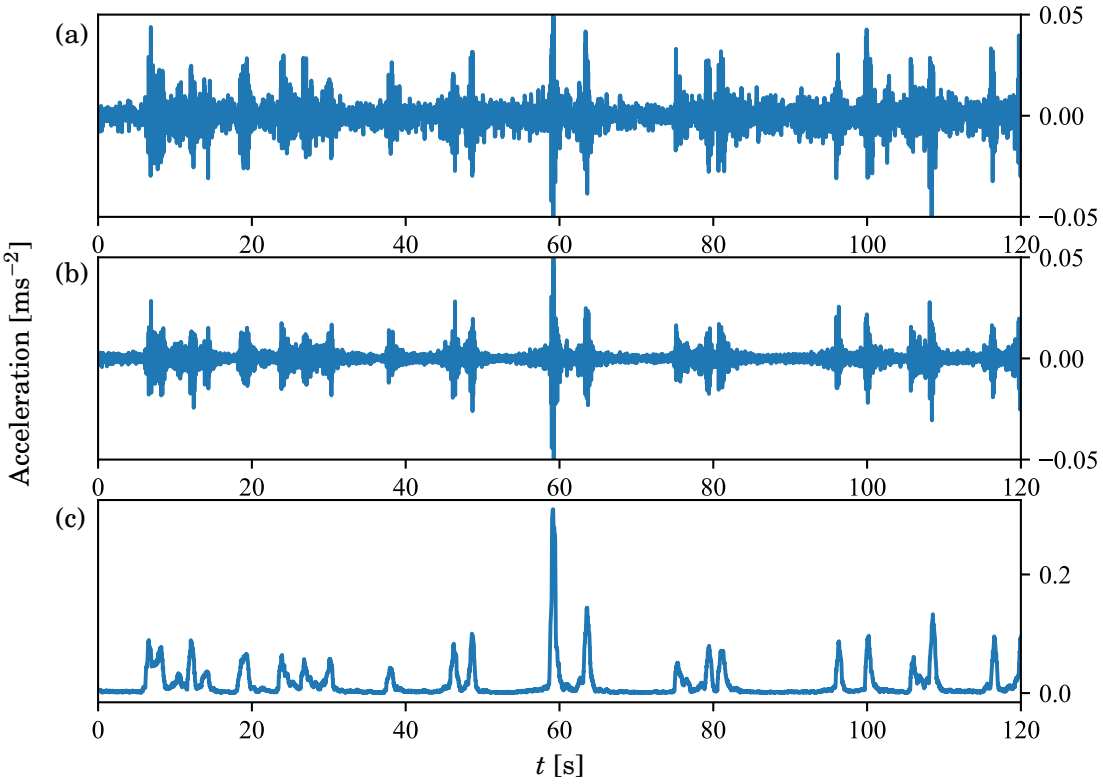



FIGURE 2.13. Two minute accelerometer windows from trial 2. (a) Pedestrians, cyclists, and congestion induce noise into the raw accelerometer readings. (b) A low and high pass frequency filter is applied to provide clearer structural responses from which the IAE is computed (c).

computed from the filtered data, see Figures 2.13(b) and 2.13(c). An illustration of our vehicle counting method in operation on a two-minute accelerometer window from trial 2, and thus realistic traffic conditions, is shown in Figure 2.14.

Figure 2.15 illustrates the number of vehicles predicted by our system $n^*$ for each trial 2 window, against the total ground-truth vehicle count $n_{\mathrm{LW}} + n_{\mathrm{C}}$, that is, the total number of vehicles that travel passed the accelerometer from either direction. Overall, the method achieves a somewhat disappointing prediction accuracy of 12%, however, the method does not systematically over or under estimate the number of vehicles.

As with the results provided in Section 2.5, vehicles entering the Clifton end of the CSB can cause the method to overestimate the number of vehicles in the trial 2 accelerometer windows (the log-normal basis function is a poor fit to the structural response caused by a single vehicle entering from that end, see Section 2.6), see Figure 2.16(a). Unfortunately, there are two

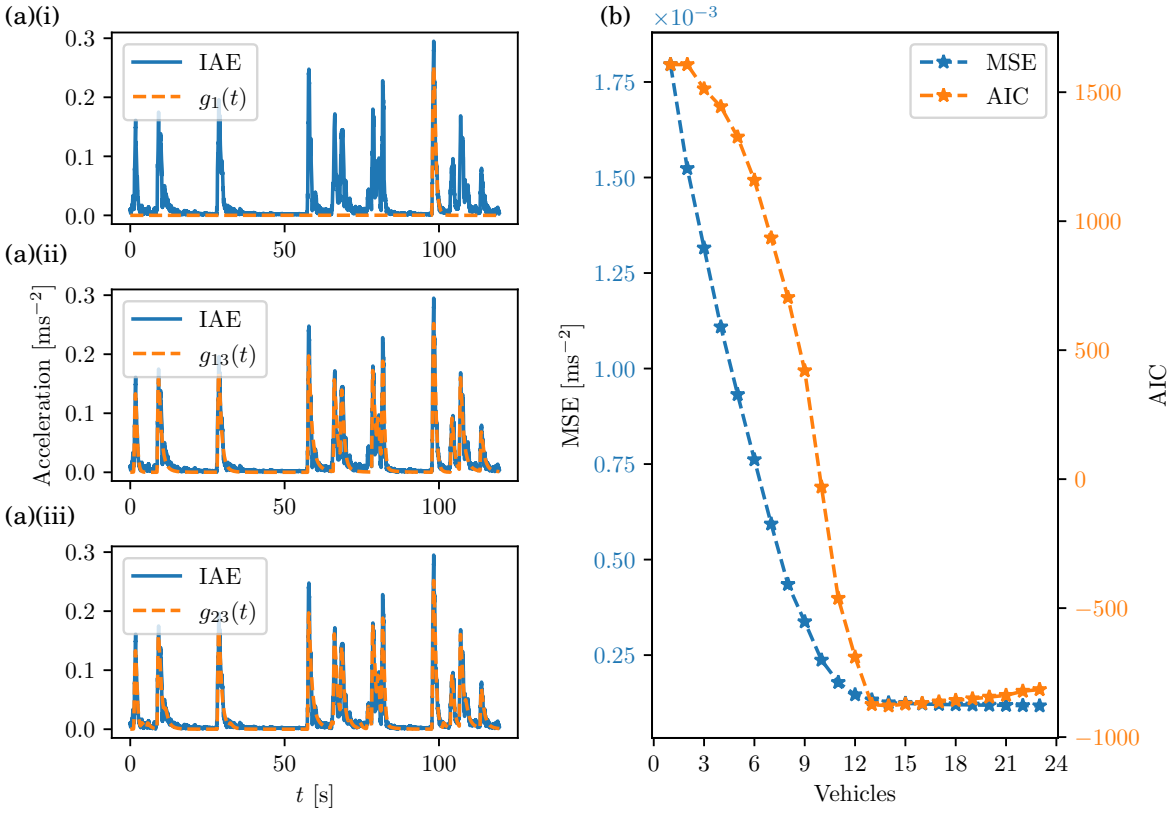



FIGURE 2.14. Vehicle estimation method in operation on a trial 2 accelerometer window with ground-truth vehicle count $n_{\text{LW}} + n_{\text{C}} = 13$. (a) Superposition of one, 13, and 23 basis functions. (b) MSE and AIC. Here, the method correctly estimates $n^* = 13$ vehicles.

additional problems that arise when using the method to estimate the number of vehicles in the trial 2 accelerometer windows:

1. busier traffic conditions on the bridge can cause the linear superposition assumption to fail, causing the method to underestimate the number vehicles, see Figure 2.16(b); and

2. the structural response caused by vehicles that enter the bridge at either the beginning or end of a window may be recorded in the previous or subsequent window. Note, that these these windows are removed from the trial 1 windows (leaving windows collected at night time).

The trial 2 windows are collected in realistic traffic conditions (in contrast to the windows segmented from the trial 1 data), and thus, have higher ground-truth vehicle counts. As a result, the method is not likely to provide an exact vehicle count estimation. However, bridge

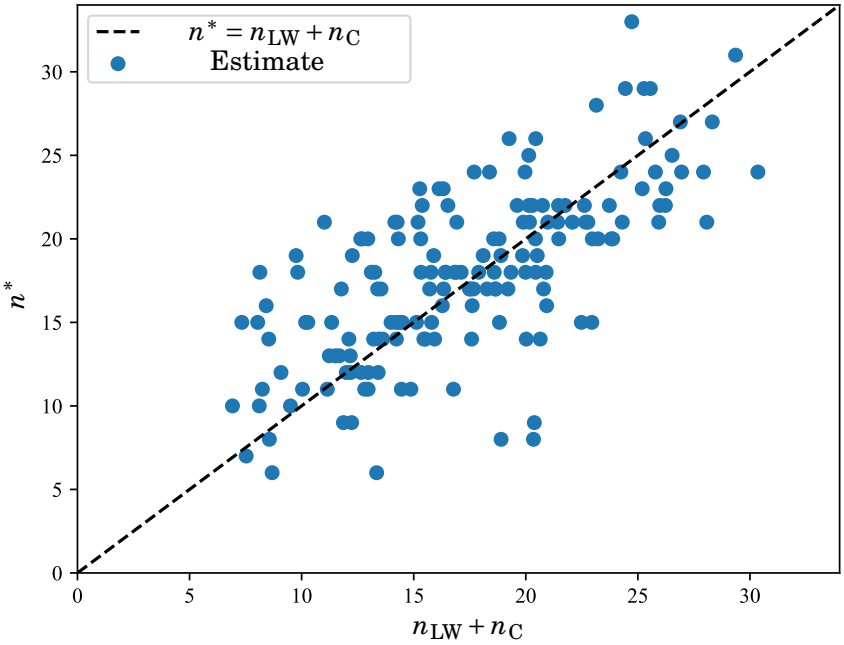



FIGURE 2.15. Vehicle estimation method applied to trial 2 windows.

maintenance teams and operators, who may use vehicle counts to either plan road improvement schemes or as an incident detection capability are likely to only need an unbiased estimate of vehicle counts, rather than an exact one, since they are only interested in aggregate traffic flows over longer time periods.

Therefore, we now shift from a prediction scheme (i.e., how often does the system exactly estimate the correct number of vehicles) to a regression framework where we instead consider the mean absolute error (MAE) and mean absolute percentage error (MAPE) achieved by our system, computed by

$$\text{MAE} = \frac{1}{|\mathscr{N}|} \sum_{n \in \mathscr{N}} |n^* - (n_{LW} + n_C)| \tag{2.21}$$

and

$$\text{MAPE} = 100 \times \frac{1}{|\mathscr{N}|} \sum_{n \in \mathscr{N}} \frac{|n^* - (n_{LW} + n_C)|}{n_{LW} + n_C}. \tag{2.22}$$

Here, $\mathscr{N}$ denotes the set of two-minute accelerometer windows that are used to test the system (i.e., usually a withheld validation set). Overall, across the $|\mathscr{N}| = 173$ trial 2 windows, the system achieves an MAE of 3.23 vehicles and an MAPE of 22%, which represent a more promising operational capability than the raw 12% prediction accuracy.

The shift from a prediction to a regression-based vehicle counting methodology prompts us to consider the following additional research question:

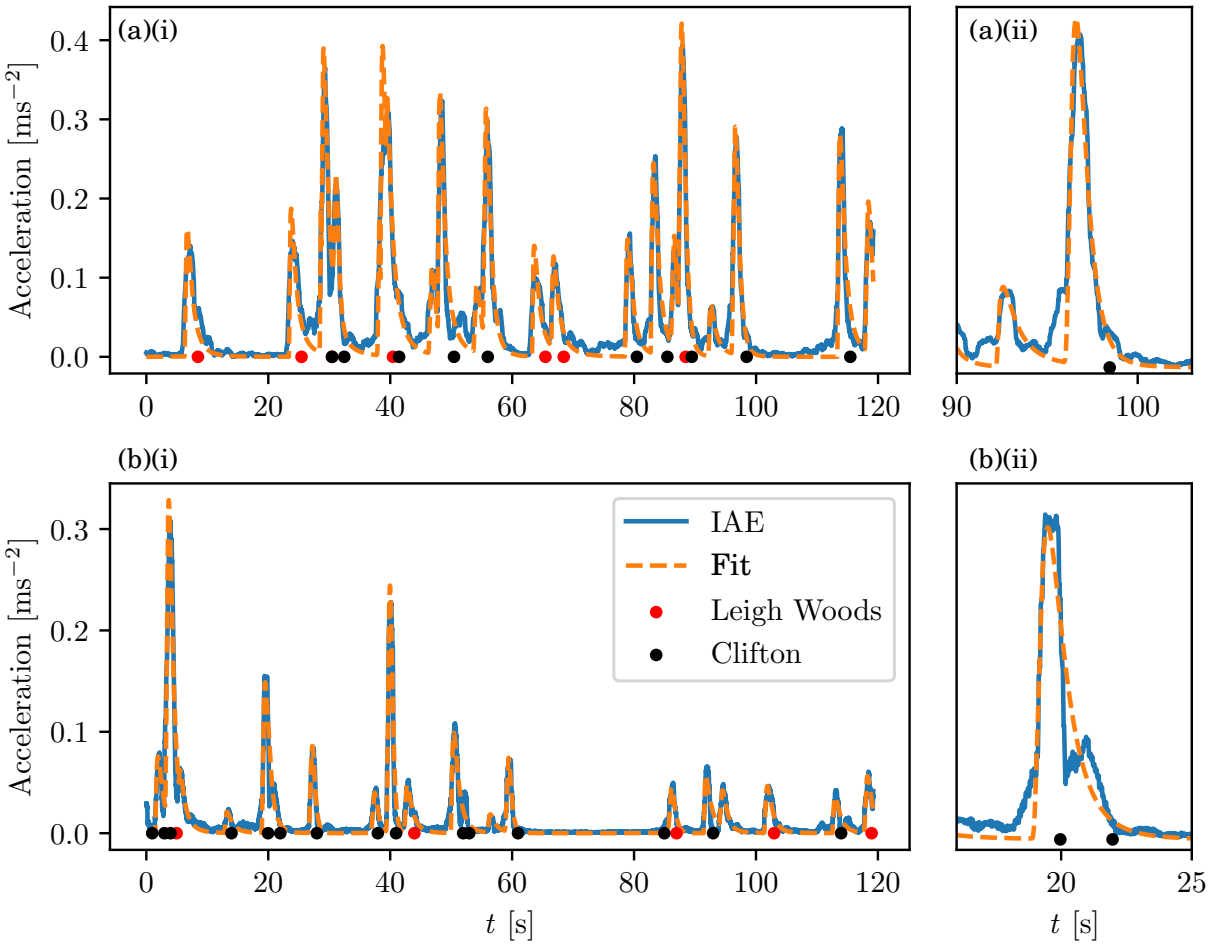



FIGURE 2.16. Incorrect vehicle count estimations of trial 2 accelerometer windows. (a) Structural response to a vehicle entering from the Clifton end of the bridge is modelled with two basis functions. (b) The linear superposition assumption fails and two vehicles recorded between 15 and 20 seconds are modelled with the same basis function.

RQ1.3. *What is the existing state-of-the-art for such time series regression (TSR) problems, and might they provide an improved vehicle count estimation system?*

To consider this additional research question, we now explore the state-of-the-art for TSR problems, and apply them to the accelerometer windows that we have already constructed.

## 2.10 Time Series Regression (TSR) Methods

While there is a large body of existing work on time series classification (TSC) (see Section 2.1.2), methods designed specifically for TSR (i.e., mapping a time series input to a continuous scalar

label) are less common. Time series regression can be viewed as a generalisation of the forecasting models described on page 18, with the relaxation that the label (i.e., what is predicted by the system) may not be inherent to the time series itself.

Previous works focusing on TSR have tended to develop specialised methods, designed for a particular task, such as those to predict heart and respiratory rates from photoplethysmogram (PPG) time series [202, 218]. In contrast, the first general TSR framework, published only in 2021, is provided by Wei Tan et al. [246]. The authors developed and tested a wide range of regression methods against 19 different benchmark data sets, each for a specific TSR task.

Generally speaking, the methods proposed in [246] build upon existing TSC schemes that are adapted for TSR, and follow the standard machine learning paradigm: features are extracted from the time series that are then subsequently fed into a standard regressor. Overall, a random convolutional kernel transform (ROCKET) feature extractor with a ridge regressor achieved state-of-the-art performance over the 19 benchmark data sets, and thus we take it forward to consider as our new regression-based vehicle counting capability.

We now describe the feature extraction process: the regressors are then described in Section 2.10.2.

### 2.10.1 Feature Extraction

#### Random Convolutional Kernel Transform

The ROCKET method was first proposed for TSC by Dempster et al. [54] and obtained state-of-the-art classification performance with hugely reduced computational expense compared with other existing methods. To classify a time series, ROCKET generates a set of one-dimensional convolutional filters (see page 71) each with random length and weights. The kernels are then applied to the inputted time series to produce a set of features at different scales (i.e., both global and local) that are inputted into a machine learning algorithm to classify the time series. In contrast to standard CNNs (see Section 3.3.5), the weights attributed to each filter are not optimised during training, rather, they are randomly assigned (hence the reduced computational expense). The number of filters used by the ROCKET method for feature extraction is considered in Section 2.10.3.

#### Spectrogram Thresholding

In addition to the state-of-the-art TSR ROCKET method provided in [246], we propose our own relatively simple regression-based vehicle counting method. In contrast to our first method, that models the accelerometer amplitudes (via the IAEs, see Section 2.3.2), we now instead consider the frequency components of the bridge's structural response under loading.

Each two-minute accelerometer window is transformed into a spectrogram, $s_a(f,t)$, that represents the accelerometer's frequency power spectrum density ($f$) at discrete time intervals ($t$) across the window, see Figure 2.17. Each column of the spectrogram represents the short-time

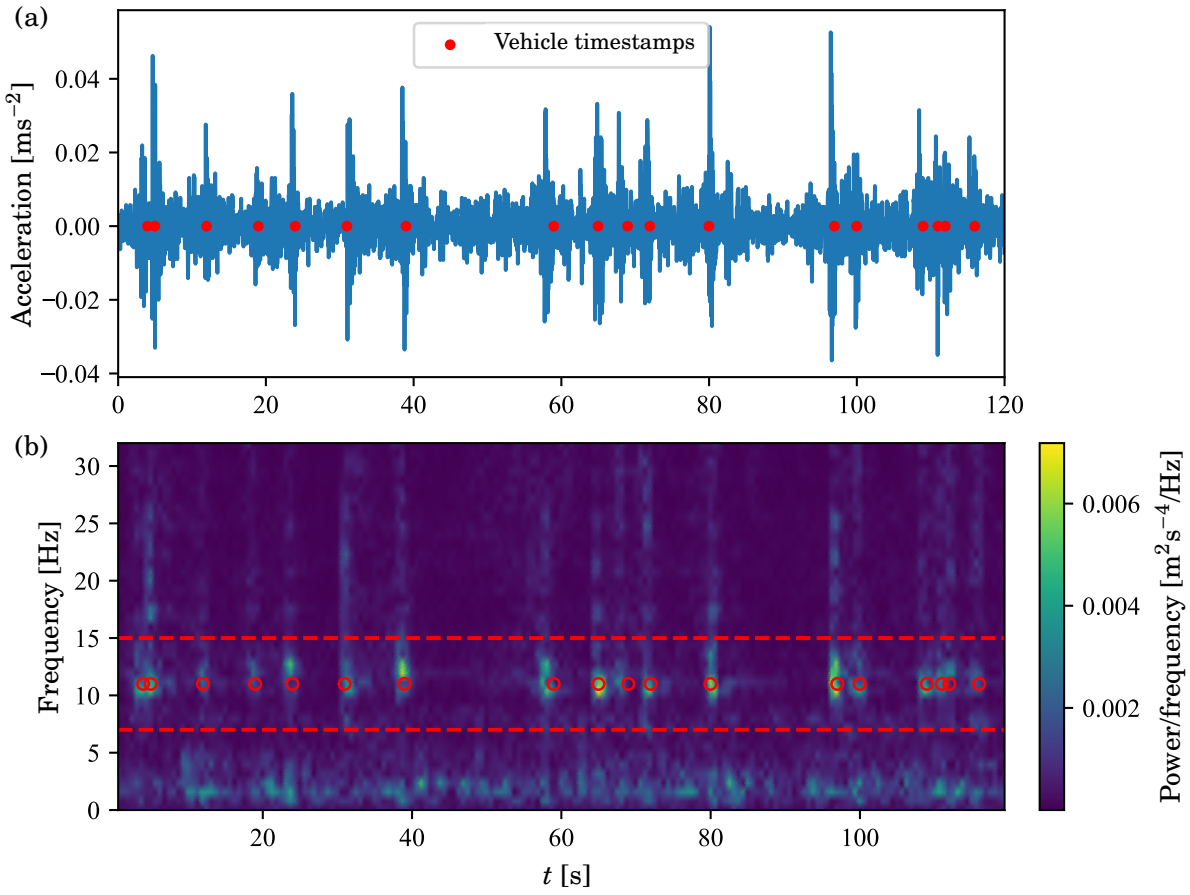



FIGURE 2.17. (a) Two-minute window of accelerometer readings and vehicle timestamps (for both directions) depicted by the red circles. (b) Corresponding spectrogram, note, for illustration purposes the timestamps are not filled. The red lines are plotted at 7 Hz and 15 Hz.

Fourier transform (STFT) of a localised windowed section of the accelerometer readings [93]. As such, we need to chose the window length and the number of overlapping readings in consecutive windows, respectively set to 80 readings and 40 readings. The spectrogram for the inputted accelerometer readings (with units ms$^{-2}$) sampled at 64 Hz has units m$^2$s$^{-4}$/Hz.

Spectrograms are a commonly used feature space for a number of TSC problems, for example, human gait [248], music [47], and acoustic-based scene classification [257]. For a comprehensive description of spectrograms and other related signal processing techniques, we direct the reader to [192].

On inspection of the accelerometer spectrograms, we discover that the moment at which a vehicle passes the accelerometer appears, at least visually, to increase the power of those frequencies between 7 Hz and 15 Hz, see Figure 2.17(b). Therefore, we crudely propose that the number of vehicles may be estimated by the number of 'hotspots' in the spectrogram power

density plots, between between 7 Hz and 15 Hz, that are greater than a threshold $\delta$, denoted by $S_a^\delta$. For each accelerometer window, we therefore extract the following single feature

$$S_a^\delta = \sum_{f=7\,\text{Hz}}^{15\,\text{Hz}} \sum_{t=0\,\text{s}}^{120\,\text{s}} S_a(f,t), \tag{2.23}$$

where

$$S_a(f,t) = \begin{cases} 1 & \text{if } s_a(f,t) > \delta, \\ 0 & \text{otherwise.} \end{cases} \tag{2.24}$$

Henceforth, for the sake of brevity, we will refer to our spectrogram thresholding-based method by its initials ST.

### 2.10.2 Regressors and Training

Given a set of $l$ features extracted from an accelerometer window $\boldsymbol{x} = [x_1, x_2, \ldots, x_l]^\intercal$ (i.e., ROCKET or spectrogram features), we consider the following standard regressors to map the features to a scalar output, and thus, estimate the number of vehicles.

**Linear**

The linear regressor assumes a linear relationship between the features and the number of vehicles, such that,

$$n^* = f_{\text{lin}}(\boldsymbol{x};\boldsymbol{\theta}) = \boldsymbol{\theta} \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}. \tag{2.25}$$

Here, $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_{l+1}]$ is a vector of weights that represents the relative importance of each extracted feature.

During training, $\boldsymbol{\theta}$ is tuned to minimise the sum squared error (SSE) between the predicted and ground-truth vehicle counts. Specifically, given $p$ feature sets extracted from $p$ training windows, $\boldsymbol{x}_i$, with corresponding ground-truth vehicle counts $n_i$ for $i = 1, 2, \ldots, p$, we find $\boldsymbol{\theta}$ that minimises

$$\text{SSE} = \sum_{i=1}^{p} (n_i - f_{\text{lin}}(\boldsymbol{x}_i;\boldsymbol{\theta}))^2 \tag{2.26}$$

via stochastic gradient descent (SGD) [31].

**Ridge**

The ridge regressor also assumes a linear relationship between the extracted features and the number of vehicles (see Equation 2.25). However, during training, the regressor employs L2 regularisation and thus Equation 2.26 is modified to give

$$\text{SSE}_{\text{L2}} = \sum_{i=1}^{p} (n_i - f_{\text{lin}}(\boldsymbol{x}_i;\boldsymbol{\theta}))^2 + \lambda \sum_{j=1}^{l+1} \theta_j^2. \tag{2.27}$$

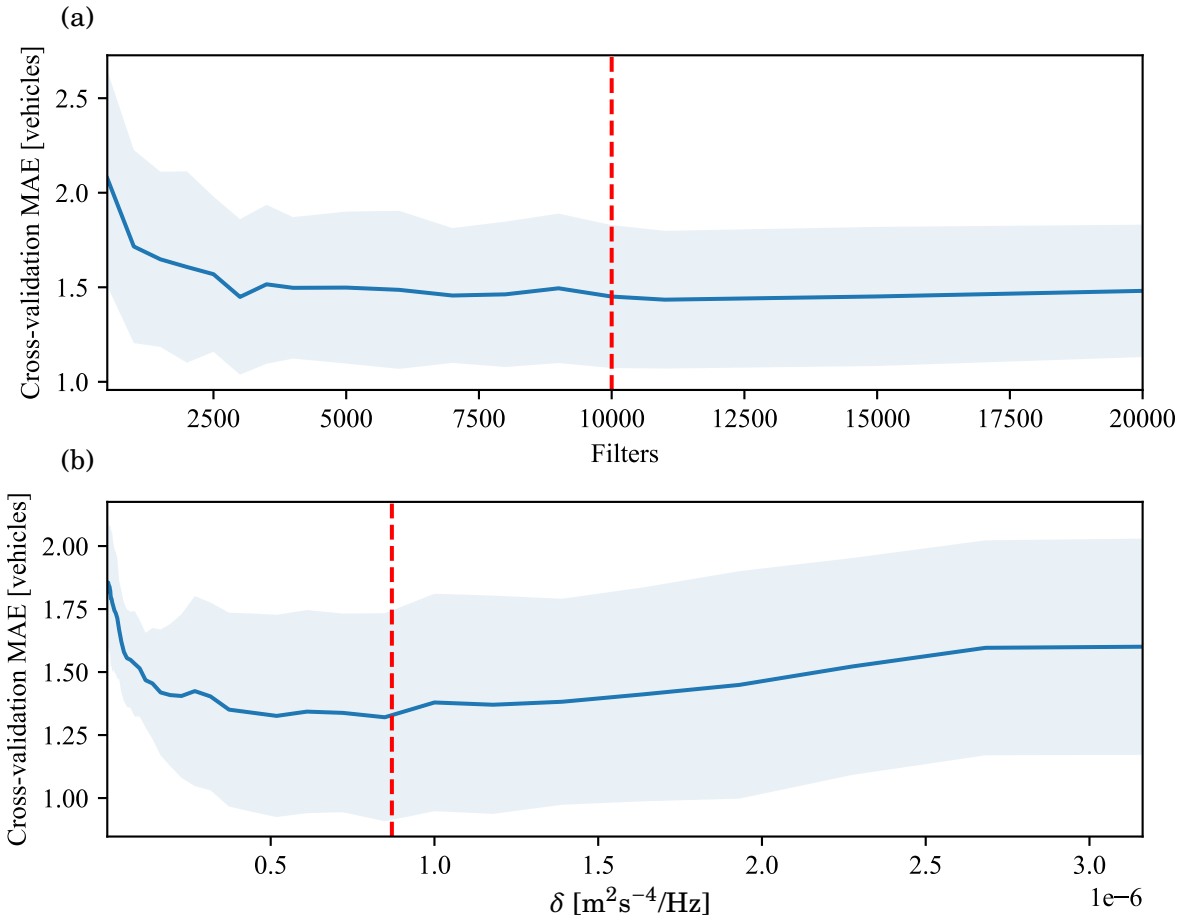



FIGURE 2.18. 10-fold cross-validation MAE achieved by the (a) ROCKET method and (b) ST method for varying numbers of convolutional filters and values of $\delta$ respectively. The average MAE across the ($k = 10$) partitions is shown by the solid blue line, whereas the shaded region illustrates one standard deviation either side of the mean. The minimum average MAEs are depicted by the red lines.

Here, the regularisation parameter $\lambda$ penalises parameters that become too large, and thus prevents over-fitting. A regularisation parameter of $\lambda = 0.5$ was found to work well when using a ridge regressor on the accelerometer windows.

In the following sections, we proceed to train and test the regression-based methods. As such, the trial 2 two-minute accelerometer windows are split into a training set of 115 windows (66%) and a withheld validation set of 58 windows (33%), respectively used to tune each method's (i.e., a feature extractor and a regressor) parameters, and determine their performance.

TABLE 2.4. Mean absolute error (MAE) achieved on the withheld validation two-minute windows by each combination of features and regressors for (a) raw and (b) filtered accelerometer data.

(a)

| MAE [vehicles] | **Regressor** | |
|---|---|---|
| **Feature** | Linear | Ridge |
| ROCKET | 1.95 | 2.00 |
| Spectrogram | **1.36** | **1.36** |

(b)

| MAE [vehicles] | **Regressor** | |
|---|---|---|
| **Feature** | Linear | Ridge |
| ROCKET | 1.67 | 1.61 |
| Spectrogram | **1.36** | **1.36** |

### 2.10.3 Determination of the Number of Convolutional Filters

We now determine the optimal number of convolutional filters used for feature extraction by the ROCKET method, see Section 2.10.1. To achieve this, we consider the method's performance on the 115 training windows for varying numbers of filters.

First, for any given number of filters, features are extracted from each of the training windows. Then, we employ $k$-fold cross-validation ($k$-fcv) to split the training windows into $k$ equally sized partitions [215] (i.e., partitions of $115/k$ windows, rounded to the nearest integer). We then iteratively consider each of the $k$ partitions as a withheld validation set. The remaining $k-1$ windows are used to train the ridge regressor which is tested by computing the MAE achieved over the validation set.

For each considered number of filters, $k$-fcv provides $k$ MAEs computed for each of the $k$ partitions, and thus, we consider the partition-average MAE. Overall, a ROCKET method with 10,000 filters minimised the average MAE computed via a 10-fcv ($k$ = 10) method, see Figure 2.18(a). Thus, we use this number of filters in what follows.

### 2.10.4 Determination of the Spectrogram Threshold Parameter

Similar to the analysis performed in the previous section, the training windows are now used to determine the value of the ST parameter $\delta$, again via 10-fcv. In this case, a ridge regressor is trained and tested on spectrogram features extracted for a range of thresholds via Equations 2.23 and 2.24. Overall, a threshold of $\delta = 8 \times 10^{-7}$ m$^2$s$^{-4}$/Hz minimised the average MAE computed via the 10-fcv method, see Figure 2.18(b).

## 2.11 Regression-based Vehicle Estimation

We now consider the performance of each combination of feature extractor and regressor. The 115 two-minute training windows are used to tune the regressor weights $\boldsymbol{\theta}$. The performance of each vehicle estimation method is then determined by the MAE, defined in Equation 2.21, achieved over the validation windows.

Table 2.4 shows the MAE achieved by each method when they are used on both the raw accelerometer data, and when the low and high-pass filter is first applied to the accelerometer

TABLE 2.5. Number of training and validation windows, and the MAE and MAPE achieved by each feature extraction method with a ridge regressor for various window lengths.

| | | | Spectrogram | | ROCKET | |
|---|---|---|---|---|---|---|
| Window length [minutes] | Train | Validation | MAE [vehicles] | MAPE [%] | MAE [vehicles] | MAPE [%] |
| 2 | 115 | 58 | 1.36 | 8.16 | 1.61 | 9.57 |
| 3 | 75 | 38 | 1.75 | 6.97 | 2.11 | 8.31 |
| 4 | 56 | 28 | 1.91 | 6.29 | 2.11 | 7.22 |
| 5 | 43 | 22 | 2.33 | 6.83 | 3.94 | 11.16 |

data (see Section 2.8.1). Overall, our ST method, with either a linear or ridge regressor, achieves the lowest MAE of 1.36 vehicles.

### 2.11.1 Varying Window Lengths

Recall (page 39), we noted that the structural response caused by a vehicle either at the beginning or end of the window can cause the method to perform poorly on the trial 2 windows (as the vehicle may be recorded by the observer in either the preceding or subsequent window). Therefore, to improve the accuracy of the ground-truth vehicle counts, we now consider extending the accelerometer windows, thus reducing the likelihood that a vehicle enters the bridge across a window boundary (i.e., near to the beginning or end of a window).

We consider four different accelerometer window lengths; two, three, four, and five minutes. For each length, the windows are segmented as described in Section 2.8.1. The corresponding number of training and validation windows, along with the MAE and MAPE achieved by each of the ROCKET and ST method (with filtered data and a ridge regressor) on the validation windows are given in Table 2.5.

## 2.12 Discussion

The results in Table 2.4 show that our regression framework may estimate the number of vehicles that travel across the CSB in a two-minute window to roughly one-vehicle accuracy. These results are incredibly encouraging, and suggest we have the beginnings of an operational system. The lowest MAE achieved on the two-minute validation windows of 1.36 vehicles (by our ST method) corresponds to an 8.16% MAPE. Not only does this beat the MAE (and MAPE) achieved by a state-of-the-art TSR method (i.e., ROCKET), it is comparable with the 7% error obtained by the specialised computer-vision based method presented in [234].

Our ST method performs identically well with both the raw and filtered data, see Table 2.4. This is a result of only considering those spectrogram powers between 7 Hz and 15 Hz, see Figure 2.17. Thus, a low and high pass filter is applied to the raw accelerometer data during

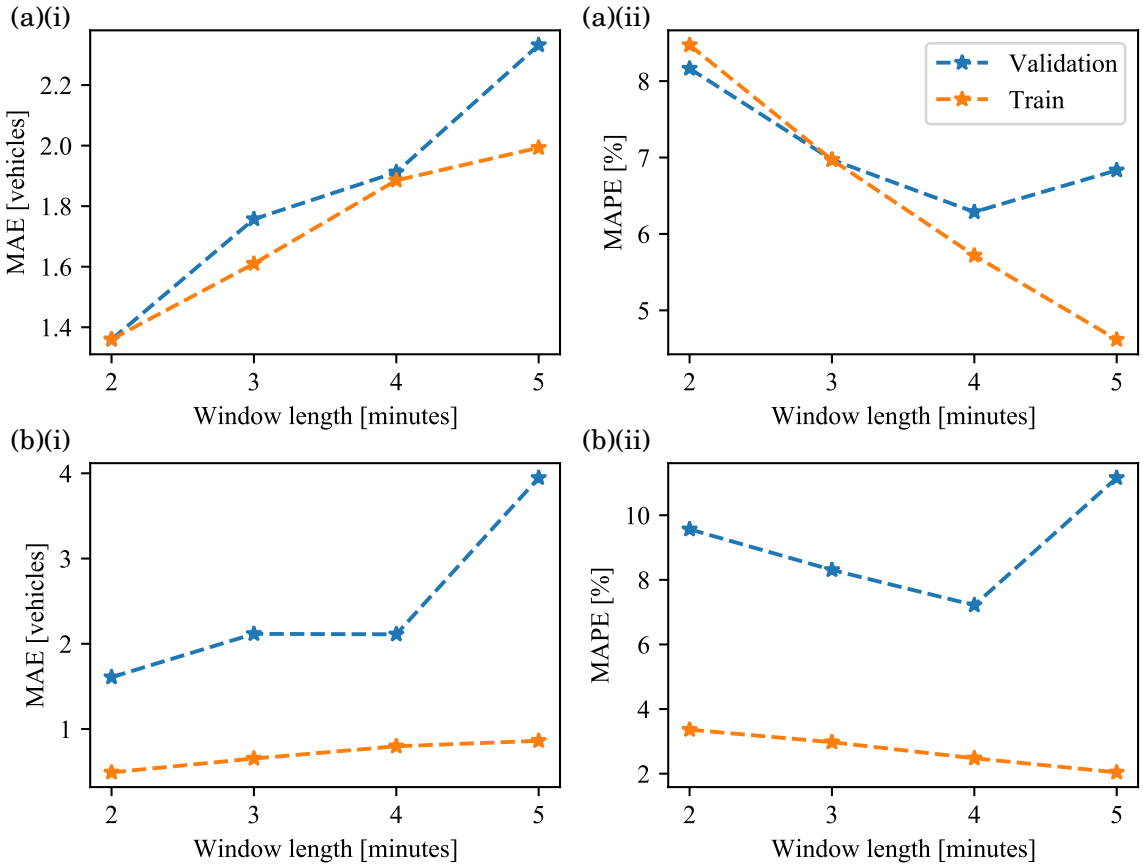



FIGURE 2.19. Train and validation MAE and MAPE for various window lengths, with filtered accelerometer data. (a) Spectrogram features with a ridge regressor and (b) ROCKET features with a ridge regressor.

feature extraction. By contrast, by filtering out the low frequency components of the bridge's structural response (which may be caused by cyclists and pedestrians), we improve the MAE and MAPE achieved by the ROCKET method.

An MAE of 1.36 vehicles is achieved by our ST method with both the linear and ridge regressors, which suggests there is no benefit in employing L2 regularisation. This is unsurprising, as only a single feature ($S_a^\delta$) is extracted for each window and thus the linear regressor is unlikely to overfit the training windows. On the other hand, with the filtered data, the ridge regressor improves the MAE achieved by the ROCKET method where a large number of features are extracted (by the 10,000 filters).

By extending the accelerometer window length, and thus improving the ground-truth vehicle count accuracy, we reduce the MAPE obtained by the regression methods. The MAE increases with window length, however this is expected as the windows contain more and more vehicles.

That said, over the four-minute accelerometer windows, our ST method achieves an average MAE of 1.91 vehicles corresponding to a 6.28% MAPE.

Unfortunately, the trade-off with increasing the window length is that we also reduce the number of training and validation windows, see Table 2.5. As a result, when we extend the windows to five minutes the regression methods overfit the (small number of) training windows and perform poorly on the withheld validation windows, see Figure 2.19. As such, to further improve and test our methods, additional trials should be conducted to collect more accelerometer data and ground-truth vehicle counts so that longer time windows may be considered.

### 2.12.1 Potential Improvements in a Future Vehicle Count Estimation System

The methods as presented in this chapter can be extended in a number of different ways, and thus present opportunities for further research. Firstly, note we cannot presently distinguish between different directions of travel. In Section 2.5 we identified that the log-normal basis function was a better fit to the structural response caused by vehicles travelling from the Leigh Woods end of the bridge. Thus, using a different basis function for each direction of travel might potentially offer a solution. However, when considering the trial 2 data, collected in real traffic conditions, our initial basis function fitting method is outperformed by our ST method (respectively achieving an MAE of 3.23 vehicles and 1.36 vehicles).

We have considered a case study on the CSB — a relatively small bridge (compared to the Humber bridge, for example) with a single lane in each direction. Future work should therefore investigate to what degree our methods work for bridges with multiple lanes in the same direction, where a larger number of vehicles (more than two) may pass the accelerometer at the same time. It seems likely that busier, multi-lane traffic conditions would make it difficult to distinguish a bridge's structural response to each vehicle. Furthermore, the CSB is a much more compliant structure than modern concrete bridges. Thus, future work should investigate to what degree our methods work for those bridges whose structural response under loading might be less pronounced than that considered here.

Our vehicle counting methods are based on the data collected from an SHM setup, and thus have their limitations when compared against systems designed and installed specifically for traffic management purposes (where sensors may be installed on the road surface, for example). For example, methods based on ILD data can compute vehicle speed, and type [189]. Therefore, future research could potentially investigate if these vehicle characteristics might also be determined directly from structural response data.

Nonetheless, the work presented in this chapter demonstrates the potential to exploit an existing data source for a secondary purpose — with our proposed method, bridge operators may simultaneously monitor the structural condition of the bridge and estimate the number of vehicles that travel across the bridge. Furthermore, most practical traffic management and incident detection practices typically rely on an unbiased vehicle count over time — achieved by

both the basis function fitting-based and TSR-based methods developed in this chapter.

## 2.13 Method Justification

The rationale behind the work presented in this chapter is based on a simple principle: large civil infrastructures are often already equipped with a suite of sensors, and so our methods, that add extra value to these existing data streams, may enrich SHM practices with no additional cost (assuming no extra sensor installation or maintenance costs).

In Section 2.4, our initial method was developed with accelerometer data in which the CSB's structural response to each vehicle was clear (due to the relatively low vehicle counts), and thus our method aimed to model this response in the time domain. The effectiveness of this first method was determined via a prediction accuracy metric, again suitable in this case given the low vehicle counts.

Our second method, developed with trial 2 data collected in real conditions, showed that the assumptions made by the initial method did not hold (e.g., linear superposition), and thus we were required to improve the method. Our frequency-based algorithm considered spectrogram features — a commonly used feature space in time series analysis, see Page 42. We determined the effectiveness of our second method through a regression-based framework (i.e, MAE and MAPE): appropriate metrics for road operators who typically require aggregate traffic flows estimates over long time periods.

The accelerometer windows captured in each trial were split into train and test sets; respectively used to tune and determine the effectiveness of each method (trial 1 windows for the first method, trial 2 windows for the second method). By segregating the accelerometer data in this way, we ensure the validity of our results (i.e., no model bias). Further, to compare our second method against existing systems, we applied a leading method (ROCKET) tested in a seminal TSR paper ([246]) to the trail 2 windows.

## 2.14 Conclusion

In this chapter we have developed and tested novel vehicle counting methods based on the data collected from a structural health monitoring (SHM) system installed on the Clifton Suspension Bridge (CSB). The key idea and motivation of the work is that SHM systems are widely deployed, but seldom used for a secondary purpose that extends beyond that for which they were designed at the point of installation. Furthermore, SHM deployments are usually cheap and simple to maintain and install (in comparison to ILDs for example, where maintenance and installation may require a road closure), and thus represent a rich potential data source for future traffic management practices.

First we considered the data captured from a single accelerometer, installed as part of a wider wireless sensor network on the CSB. A prediction-based vehicle counting system was developed,

built upon a linear superposition assumption, that achieved a 74% prediction accuracy overall. However, due to inaccuracies in the ground-truth vehicle counts (i.e., truncation of the toll barrier timestamps), testing of the system was initially limited to a small number of vehicles travelling in one direction across the bridge.

To test the system in real conditions, a second trial was performed with a simpler single-sensor setup, in which human observers collected accurate ground-truth vehicle counts. On this new data, our initial method achieved an underwhelming 12% prediction accuracy (i.e., the proportion of windows where the estimated count is exact), explained by the relatively large number of vehicles present in each window. This result caused us to consider the method's effectiveness through a regression-based analysis: after all, estimating 19 vehicles when there are actually 20 is far better than estimating one! This motivated us to explore the state-of-the art in time series regression, and how it may be applied to our problem. In addition, we considered our own relatively simple method, based on thresholding the frequency components of the bridge's structural response.

On a withheld validation set of two-minute accelerometer windows, our method achieved a mean average error (MAE) of 1.36 vehicles with a corresponding mean average percentage error (MAPE) of 8.16%. Furthermore, as we increased the window length, and thus improved the accuracy of the ground-truth counts, the MAPE was reduced to 6.29%. All methods presented in this chapter are designed to consider the data captured from a single accelerometer, and thus may be applied to data captured from either complex multi-sensor SHM installations or more limited setups.

CHAPTER 3

# APPLIED COMPUTER VISION FOR RAPID UPDATING OF THE HIGHWAY ASSET INVENTORY

*Sections 3.2–3.5 are based on work summarised in the article 'Computer Vision for Rapid Updating of the Highway Asset Inventory' published in the Transportation Research Record (TRR) [241].*

*Section 3.6 is based on work summarised in the paper 'Augmented Visual SLAM for the Localisation of a Transportation Asset Management Survey Vehicle' published in the Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC) [239].*

**Problem Statement**

Highway agencies maintain an accurate and up-to-date roadside asset inventory via manual inspection of remote survey imagery, which given the large number of assets (e.g., traffic signs and lighting units), can become a difficult and time-consuming task. Therefore, we develop a computer vision-based decision support system designed to automatically verify the inventory record of each asset (i.e., confirm that the asset is installed in the correct location), and present those relatively small number of incorrectly recorded assets to the analyst for further manual inspection.

**Chapter Glossary**

| Term | Definition |
|---|---|
| AVIS | Asset Visual and Information System |
| BA | Bundle adjustment |
| CNN | Convolutional neural network |
| DoT | US Department of Transportation |
| FCNN | Fully connected neural network |
| GTSDB | German traffic sign database |
| HA | Highways Agency |
| HAPMS | Highways Agency pavement management system |
| HCI | Human computer interaction |
| HE | Highways England (UK highway agency, formally HA) |
| HOG | Histogram of oriented gradients |
| IAM-IS | Integrated asset management information system |
| IB | Intentional binding |
| ILSVRC | ImageNet large scale visual recognition challenge |
| IMU | Inertial Measurement Unit |
| LISA | Laboratory for safe and intelligent automobiles |
| MS | Matrix sign |
| ORB | Oriented FAST and rotated BRIEF (features) |
| OS | Ordnance Survey |
| PTAM | Parallel tracking and mapping |
| RANSAC | Random sample consensus |
| RM | Reference marker |
| SfM | Structure-from-motion |
| SIFT | Scale-invariant feature transform |
| SHA | (US) State Highway Association |
| SLAM | Simultaneous localisation and mapping |
| VSLAM | Visual SLAM |
| SGD | Stochastic gradient descent |
| SMIS | Structures management information system |
| SOA | Sense of agency |
| SRN | Strategic road network |
| SURF | Speeded-up robust features |
| TAM | Transportation asset management |
| TO | Traffic officer |
| TS | Traffic sign |
| YOLO | You only look once |

Roadside assets such as traffic signs that provide guidance to drivers, street lights that illuminate the highway at night, and CCTV cameras that enable agencies to monitor the highway are all crucial to a functioning highway system.

To effectively manage roadside assets, agencies rely on an inventory that contains each asset's geographical (e.g., position), physical (e.g., width and height), and condition data [17, 110, 251]. Thus, the inventory allows agencies to design appropriate maintenance schedules and improvement schemes [18, 115].

Agency analysts periodically check the inventory to ensure it is accurate and up to date. This updating is usually performed in two stages; first, asset data is collected in a field survey of the highway (often with a survey vehicle equipped with sensors). Then, secondly, the survey data is inspected by an analyst to determine which assets are correctly recorded in the inventory, and which need updating [111]. Survey data collection and asset inventory are important TAM practices, and thus highway agencies routinely survey roadside assets; over 70% of U.S. state highway agencies periodically survey the highway [179].

There are a number of reasons why an inventory record may need updating: the condition of the asset may have changed, or an asset may have been installed between surveys, for example. Consequently, inventory updating via manual inspection of survey data can be a difficult and time-consuming task to comprehensively and accurately undertake. Furthermore, the number of assets that are recorded incorrectly is usually small compared with those that are correct (Jacobs, personal communication), and thus an analyst might spend only a fraction of their time inspecting assets that require updating. Unfortunately, due to funding constraints, the development of asset monitoring systems for high-cost, low-quantity assets has previously been prioritised [22]. Thus, there is a requirement for tools and systems to assist an analyst in their decision making when updating the inventory of (low-cost, high-quantity) roadside assets.

Cameras provide a relatively low-cost sensing capability, and therefore analysts often inspect imagery captured during a drive-by survey [195, 266]. Moreover, an analyst may immediately interpret and inspect imagery data, whereas other types of asset data require processing post-survey (such as LIDAR, for example [132]).

In this chapter, a computer vision-based decision support system is developed and tested. The system is designed to automatically verify the asset inventory and identify those assets that require further manual inspection, thus reducing the analyst's manual workload. An illustration of how our proposed system improves upon the current asset inventory updating process is provided in Figure 3.1.

## 3.1 Background

Survey imagery of roadside assets is usually captured by a vehicle equipped with sensors during a drive-by survey of the highway. Jalayer et al. [138] interviewed 50 U.S. state and 7 Canadian

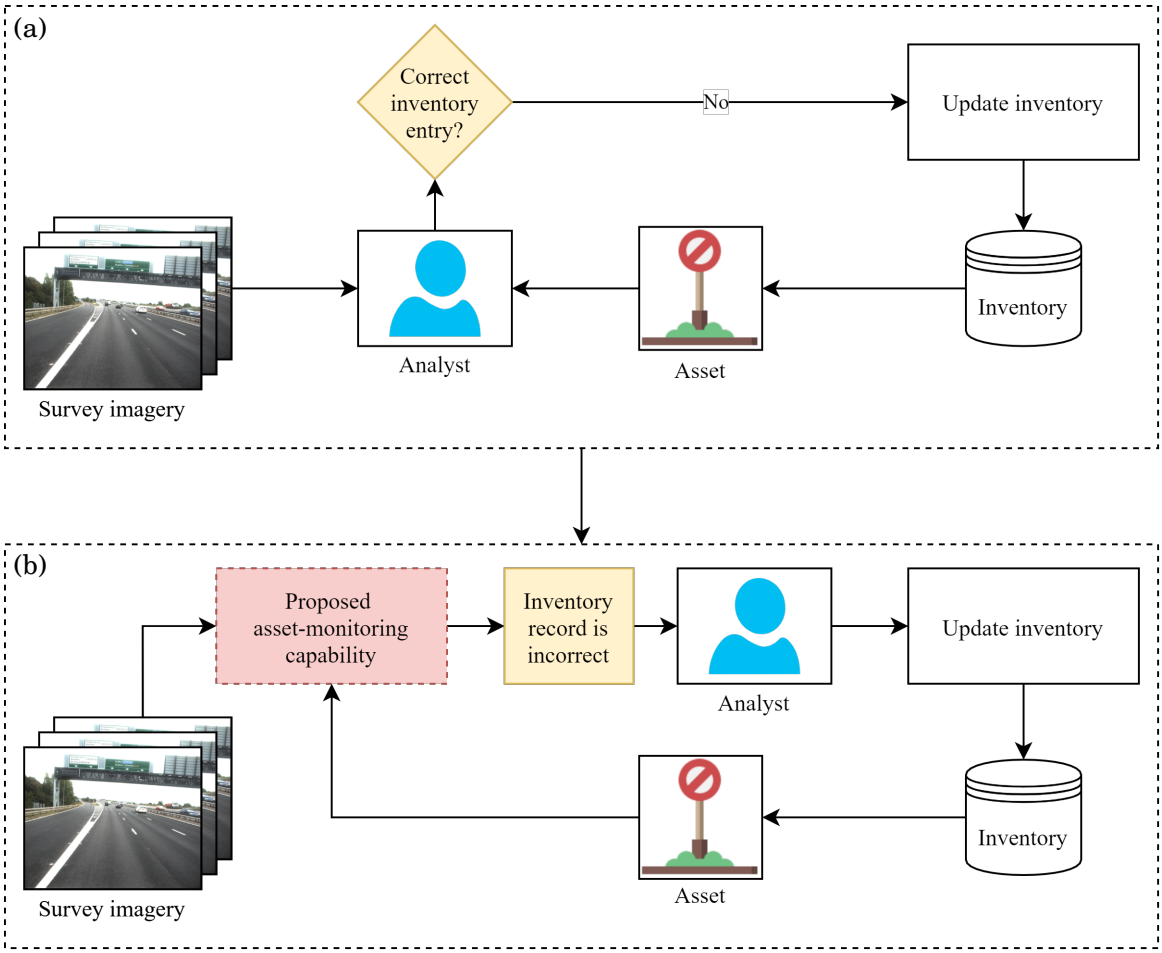



FIGURE 3.1. (a) The current roadside asset inventory updating process. An analyst identifies assets that are recorded incorrectly in the inventory via manual inspection of the survey imagery. (b) The computer vision-based decision support system proposed in this chapter automatically identifies the relatively small number of assets that require further manual inspection. The analyst then decides whether they are recorded incorrectly or not, and updates the inventory accordingly.

province departments of transport (DoTs) on their highway inventory data collection methods. Overall, over 50% of the respondent entities update the asset inventory via remotely captured imagery. In the UK, Highways England (HE, the largest agency in the UK) contract IBI group to perform the surveys; the IBI survey vehicle is shown in Figure 3.2(b). The company surveys the entire strategic road network (SRN, over 4,000 km of highway managed by HE, see page 101) once per year [266].

The survey data is often inspected manually via a bespoke piece of software that integrates the imagery with the inventory. Thus, the analyst may update the inventory from the safety of their office. For example, HE's current inventory updating process for roadside assets is

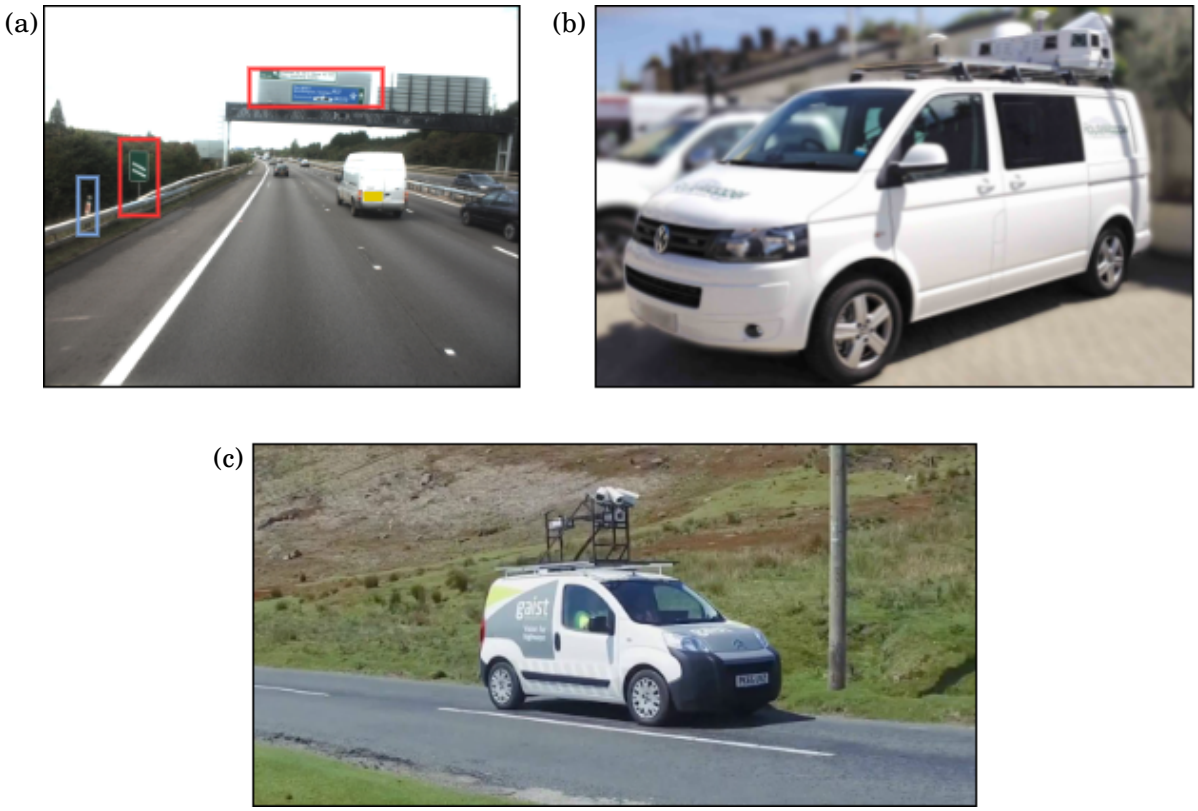



FIGURE 3.2. (a) High-resolution survey imagery of roadside assets captured on an annual drive-by survey. Two manually annotated traffic signs and a reference marker post are shown respectively by the red and blue boxes. (b) The IBI survey vehicle employed by HE. (c) Gaist survey vehicle. Panels (b) and (c) respectively reproduced from World Highways Magazine [266] and Gaist website [77].

performed through the Asset Visual and Information System (AVIS) [266] that displays high-resolution survey imagery, a top-down outline (a shapefile) of the highway, and the inventory, to the analyst. Note AVIS is only used to monitor roadside assets; for other asset types, HE employ alternative systems specific to their operational and maintenance requirements. For example, structural assets, such as bridges are managed via the Structures Management Information System (SMIS) [8], and the highway pavement surface is managed via the Highways England Pavement Management System (HAPMS, note that HE was formerly Highways Agency (HA), see page 99) [125]. The Maryland State Highway Administration (SHA) uses a bespoke imaging tool developed by L3Harris to view aerial imagery of the state highway [149]. Although the system claims to provide a level of automation, Maryland SHA have so far used the tool to automatically identify yellow and white lines — a relatively tame computer vision problem.

A number of emerging companies are competing to provide additional asset data, beyond the periodic surveys undertaken by highway agencies. One such UK-based company is Gaist,

who since their incorporation in 2007, have developed a suite of products to assist the TAM process. For example, a number of local authorities in the UK are currently employing Gaist to collect pavement condition data in urban areas where SCANNER, the vehicle historically used for highway pavement condition surveys cannot operate (it must be driving at at least 20 kph to survey the pavement surface) [131], see Figure 3.2(c). Gaist also collaborate with the University of York in a project in which refuse trucks are equipped with cameras to monitor the pavement condition [207]. The trucks typically traverse the same routes on a regular basis and therefore it is hoped that pavement deterioration might be modelled from their data. As of yet, no results or methods have been published by the collaboration. However, an image differencing method (see page 17), that compares a recent image of the pavement against an older image (where the pavement is in perfect condition) to determine deterioration (e.g., cracks) is a possible way forward.

From dialogue with the company (in 2018), it is understood that their products do not currently incorporate automation. Rather, they manually label polygons in images with condition data so that future products might be trained to evaluate the highway surface automatically.

Similarly, HE are in discussion with coach providers in the UK that regularly convey passengers along the SRN (HE, personal communication). Highways England hope to equip each coach with a camera to provide an additional asset data capture capability. The international company Mobileye have recently partnered with Ordnance Survey (OS) to develop products that map assets captured by cameras installed in a fleet of survey vehicles [174]. So far, Mobileye and OS have undertaken trials in the Manchester, the North-Eastern, and the South-Eastern areas of England. The collaboration claims to have automatically identified over 1 million unique assets, however no information on the system's asset detection, classification or mapping performance in the trials has been published. In Chapter 4, we also explore the concept of using vehicle fleets for asset monitoring. Specifically, we consider whether a fleet of traffic officer (TO) vehicles (each with a dashboard camera), who already patrol across highway networks, might provide an alternative, ad-hoc (see page 5) asset data capture capability, in addition to their primary function of incident response.

### 3.1.1 Computer Vision

Computer vision can be broadly understood as two distinct problem areas. Firstly, there are techniques that perform the reconstruction of the three-dimensional scene in a two-dimensional image (or collection of two-dimensional images). The structure-from-motion (SfM) methods and algorithms described rigorously in Hartley and Zisserman [97] such as feature matching methods (see Figure 3.17) and the pinhole projection equation (see Section 3.3.2) allow such rich views of the world to be created. Secondly, there are pattern recognition-based techniques that extract features from a two-dimensional image to perform classification via either supervised or unsupervised learning. A classical overview of the latter problem area is provided by Gonzalez and

Woods [85]. Further, a variety of computer vision techniques employed by intelligent transport systems are described by Loce et al. [159].

A great deal of work has been undertaken in applying computer vision for the automatic detection of assets in images. In particular, there is a large body of literature on traffic sign detection and classification, partly as a result of freely available labelled datasets such as the German Traffic Sign Detection Benchmark (GTSDB) [235] and the Laboratory for Intelligent and Safe Automobiles (LISA) Traffic Sign [175] datasets. Arnoul et al. [14] developed and deployed a Kalman filter-based system to detect roadside assets from their motion relative to a survey vehicle, and Greenhalgh and Mirmehdi [92] presented a support vector machine to classify histogram of oriented gradients (HOG) features extracted from images of traffic signs. However, state-of-the-art performance is achieved by deep neural networks; the classifiers proposed by Sermanet and LeCun [229] and Cireşan et al. [43] achieve respectively 99.17% and 99.46% accuracy on the GTSDB dataset. However, these systems consider images in which the sign occupies the majority of the image. Improved methods that can detect and classify traffic signs in whole scene imagery (in the wild) are presented by Zhu et al. [274] and Kryvinska et al. [148].

In addition to detecting and classifying assets in imagery, a number of systems that estimate the asset position have been developed. Wang et al. [259] presented a stereo camera system to estimate the position of assets identified in imagery and tested different configurations (one vehicle with two cameras, and two vehicles each with one camera). Alternatively, a method that leverages Google Street View imagery to map the position of the assets was proposed by Balali et al. [22].

Two methods that perform a dense three-dimensional reconstruction of the highway and its assets are presented by Golparvar-Fard et al. [84] and Uslu et al. [256]. These approaches employ SfM techniques to generate point clouds purely from the imagery. Semantic segmentation is then performed on the two-dimensional image providing asset classification and three-dimensional mapping at the pixel level. Unfortunately, point cloud generation from imagery is computationally expensive, however, a cheaper LIDAR-based method is presented in Sairam et al. [226]. Examples of three roadside asset management systems in operation are shown in Figure 3.3.

Further, TAM systems that assess the condition of highway assets have been proposed. Systems that automatically evaluate the retroflectivity of traffic signs, and detect defective road studs (reflective cats eyes) are presented respectively in Chengbo and Yichang [5] and Mcloughlin et al. [172].

In addition to roadside assets, computer vision-based methods have been used successfully to monitor the pavement. Zhang et al. [269] developed a deep learning architecture to automatically identify cracks in images of asphalt pavement at the pixel level. The system is developed and tested on illuminated overhead images of the pavement and thus requires a specialised vehicle to be employed. More readily deployable pavement monitoring capabilities include the system presented in Radopoulou and Brilakis [212], which considers imagery data from a parking camera

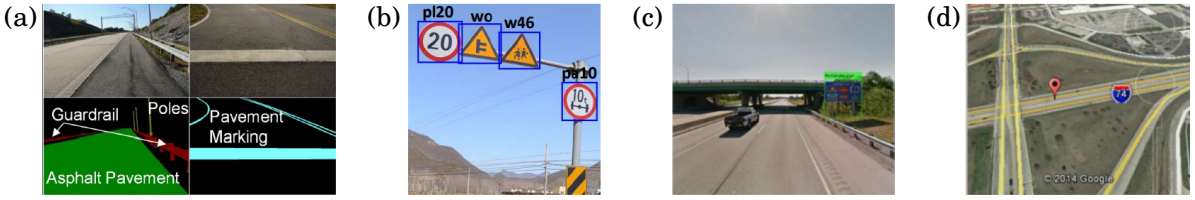



FIGURE 3.3. Three existing computer vision-based asset management systems in the literature. (a) The three-dimensional image reconstruction and asset segmentation described in Golparvar-Fard et al. [84]. (b) The labelling exercise undertaken by Zhu et al. [274]. The bounding box and class for each traffic sign are manually collected and used to train a system capable of detecting traffic signs in a whole scene image of the highway. (c) A traffic sign detected from a Google Street View image and (d) mapped by the system proposed in Balali et al. [22].

mounted on the bumper of a vehicle.

One notable computer vision crowdsource-based service is Mapillary that aims to apply automation to street-level data collected by their application's users [170]. From an asset data collection perspective, Mapillary is an exciting prospect; their application applies SfM techniques to match images taken by various users to create long, uninterrupted image sequences. As of 2019, the project had mapped over 8 million km of road around the globe [247]. Further, the project has created and published the Vistas Dataset — over 25,000 images labelled at the pixel level with 37 classes [183] — to contribute to the development of semantic segmentation systems for automatic whole-road scene understanding. The annotated images are captured by a variety of cameras (with varying distortions), all over the world, under different weather and lighting conditions.

The Mapillary application is able to detect traffic signs within the crowdsourced imagery. However, highway agencies will need to collaborate with Mapillary to improve the application if it is to enhance the TAM process. The crowdsourcing tool needs to be retrained, with imagery labelled with assets to enable it to detect multiple asset types, beyond traffic signs. Furthermore, when we tried using the Mapillary application with imagery of the highway, the system failed to detect some roadside traffic signs. An example crowdsourced image taken on a UK highway in which the Mapillary application fails to detect a roadside traffic sign is shown in Figure 3.4.

To be clear, the Mapillary application is not designed as an asset monitoring tool. Rather, it demonstrates the rich potential of crowdsourcing and data aggregation. Perhaps agency analysts of the future will inspect survey imagery of long highway segments (the entire M1 motorway in the UK, for example) which is in fact aggregated from multiple shorter surveys (potentially collected by the TOs considered in Chapter 4).


FIGURE 3.4. The Mapillary application in operation on a UK highway. The system is able to detect traffic signs within crowdsourced imagery, however, the application fails to detect some traffic signs in highway scenes.

### 3.1.2 Summary of Background Material and Research Objectives

In summary, a number of computer vision-based systems have been developed for asset detection, classification or localisation. As large labelled training data sets (such as the GTSDB and LISA data sets) become increasingly available and computational resources improve, it is likely that these systems will only continue to become more and more effective. However, these methods often only consider the imagery data in isolation and do not integrate with other data sets used by an analyst, such as the inventory. Such methods, that do not reflect the current asset management process are not likely to be implemented — a perfect computer-vision system with a classification accuracy of 100% is of little use to the analyst if its outputs cannot be used. Further, to the best of my knowledge, no previous systems have been trained for asset classification with the AVIS survey imagery. As such, our objective in this chapter is to develop an asset monitoring capability that provides automation through computer vision-based techniques while reflecting the role of an analyst to compliment and improve upon the current asset management process. Accordingly, we define the following research questions (RQs, whose numbering convention is described in page 20) to be considered in this chapter:

RQ2.1 *Can a system be developed to automatically detect roadside assets within the AVIS survey imagery?*

RQ2.2 *Can the system integrate with other data sets used by an analyst and thus reflect and improve upon the current asset management process?*

To consider our research questions, we will undertake a study with survey imagery and an inventory of roadside assets installed along an eight-kilometre section of the A27 highway maintained by HE on the south cost of England. We will develop a new computer vision-based decision support tool that is designed to automatically verify the inventory and thus rapidly identify those assets that are recorded incorrectly. To prove the feasibility of our system, three asset types will be considered in this study: traffic signs, reference markers, and matrix signs. The effectiveness of our system will be determined by considering which assets are flagged for further manual inspection.

The rest of the chapter is organised as follows. In Section 3.2 the data sources used to develop our system, and the data pre-processing steps are described. The computer vision methods that underpin our decision support system are described in detail in Section 3.3. The results obtained by our system when used to verify the inventory of a withheld test set of assets are provided in Section 3.4. The results and the system's limitations are discussed in Section 3.5. Finally, future research directions to improve the robustness of the system are proposed in Section 3.6, method justifications are provided in Section 3.7, and the chapter is concluded in Section 3.8.

## 3.2 Data Sources

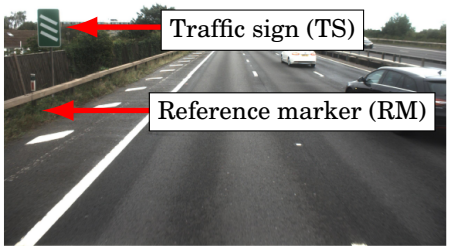
The chapter now proceeds to the development of our new roadside asset monitoring system. First, each data source used to develop the system is described. Samples of the data sources are provided in Figure 3.5.

**Asset Inventory**

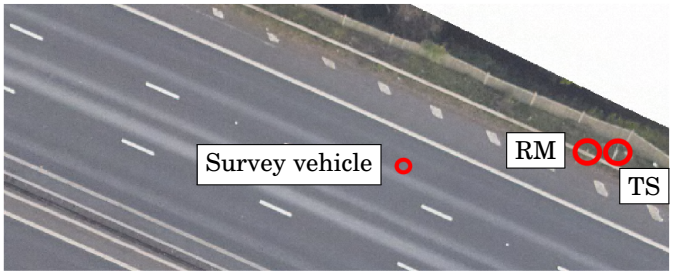
The inventory of roadside assets along the SRN is stored in the Integrated Asset Management Information System (IAM-IS) [130] that contains asset specific information such as its position, condition, installation date and maintenance history. Although the inventory must adhere to standards set by highway agencies [17, 110, 251], TAM and manual inventory updating is often performed by subcontractors and consequently the inventory can become inaccurate and inconsistent.

The position, as an OS grid Easting-Northing $(e, n)$ coordinate [63] of each asset is recorded in the inventory. For some assets, physical attributes such as the size (width and height) and mounting height are also provided. For example, the width and height of traffic signs and matrix signs are recorded, but the size of reference marker posts are not. Therefore, the reference marker posts are assumed to have a 1 m height, 0.01 m width and 0 m mounting height. In total, 590 individual assets (373 traffic signs, 172 reference markers and 45 matrix signs) are taken forward to develop and test the system.

(a)




(b)




(c)

| Asset inventory | | | | | |
|---|---|---|---|---|---|
| **Asset type** | **Easting [m]** | **Northing [m]** | **Width [m]** | **Height [m]** | **Mounting height [m]** |
| Traffic sign | 466524.87 | 104388.58 | 0.6 | 1.1 | 2.1 |
| Reference marker | 466522.47 | 104388.89 | N/A | N/A | N/A |

(d)

| Survey vehicle IMU | | |
|---|---|---|
| **Easting [m]** | **Northing [m]** | **Heading [degrees]** |
| 466509.85 | 104388.61 | 23.81 |

FIGURE 3.5. Samples of the data sources used to develop the asset monitoring decision support system. (a) A traffic sign (TS) and a reference marker (RM) in a survey image and (c) their inventory records. (b) Aerial imagery of the highway. The position of the two assets and the survey vehicle as the survey image is taken are marked. (d) Survey vehicle position, marked in panel (b), and heading provided by the IMU.

**Aerial Imagery**

Overhead imagery of the highway is periodically collected by HE via an airborne survey performed either by a small fixed-wing aircraft or helicopter. The aerial imagery is collected as a set of non-overlapping tiles [16] and stored in the Geotechnical Data Management System [124]. Software tools such as ArcMap [68] may be used to visualise the aerial imagery and obtain the Easting-Northing coordinate to the nearest 10 cm for any point on the highway, however, no height relief is provided.

**Survey Vehicle**

The vehicle is equipped with a GPS-enabled IMU and four cameras: one forward-facing, one backward-facing, one right-facing, and one left-facing [266]. However, in this work, we only consider the imagery captured by the forward-facing camera. Thus, our decision support system is also suitable for drive-by surveys performed by vehicles with more limited sensor setups.

As the survey is performed, an image is taken every 2 m along the highway. Simultaneously, the vehicle's heading angle and position are recorded by the IMU. Thus, each image is accompanied with metadata describing the position on the highway and direction of the vehicle as the image is taken. The vehicle's position as an Easting-Northing coordinate, and its heading are recorded with a nominal resolution of 0.01 m and 0.1° respectively.

### 3.2.1 Data Pre-processing

The position of each asset is given in the world coordinate system $(e, n, z)$ where $z$ is the height above the highway surface. However, the decision support system considers the assets as viewed by the camera on the vehicle. Therefore, the survey data is processed so that the assets may be described relative to the vehicle.

First, the coordinate transformation of the arbitrary coordinate in the world frame, $\boldsymbol{p} = (e_p, n_p, z_p)^\intercal$, to a coordinate relative to the IMU on the vehicle, $\boldsymbol{p}_v = (x_v, y_v, z_v)^\intercal$, is defined. Here, the $x$-coordinate axis points along the heading direction of the vehicle, the $y$-coordinate axis points orthogonally across the width of the highway and the $z$-coordinate axis continues to point vertically upwards. The coordinate transformation is achieved by a translation so that the IMU is at the origin of the $(e, n)$ plane, and a rotation through the vehicle heading angle $\theta$, about the $z$-coordinate axis. Therefore, the coordinate relative to the vehicle is given by $\boldsymbol{p}_v = \boldsymbol{R}_z(\boldsymbol{p} - \boldsymbol{o}_v)$, where $\boldsymbol{R}_z$ is three-dimensional rotation matrix about the $z$-coordinate axis through the heading angle [83]. The translation vector $\boldsymbol{o}_v = (e_v, n_v, 0)^\intercal$ is the origin of the new coordinate system and lies directly beneath the IMU on the highway surface, see Figure 3.6. The coordinates $e_v$ and $n_v$ are respectively the Easting and Northing coordinate recorded by the IMU. The position of the assets relative to the vehicle may then be computed by this transformation.

Second, a smoothing of the vehicle's heading angle is performed. The heading direction is computed by the IMU in real-time from the straight-line between the vehicle's current position and its position as the previous image was taken. The heading angle is then defined as the angle between the heading direction and the Easting-coordinate axis. This method is inaccurate when the vehicle turns a corner, as the vehicle does not have a linear motion, and therefore we re-compute the headings after the survey. Specifically, the vehicle's heading direction as the $i^{\text{th}}$ image is taken is computed via a central difference scheme applied to the position of the vehicle when the $(i-1)^{\text{th}}$ and $(i+1)^{\text{th}}$ image are taken. The corrected heading angle is thus given by

$$\theta_i \approx \arctan\left(\frac{n_{i+1} - n_{i-1}}{e_{i+1} - e_{i-1}}\right). \tag{3.1}$$

This scheme provides a better retrospective approximation for the heading compared to the IMU value that is computed in real time.

## 3.3 Computer Vision-based Methodology

We now describe and develop the computer vision-based methods that underpin our decision support system.

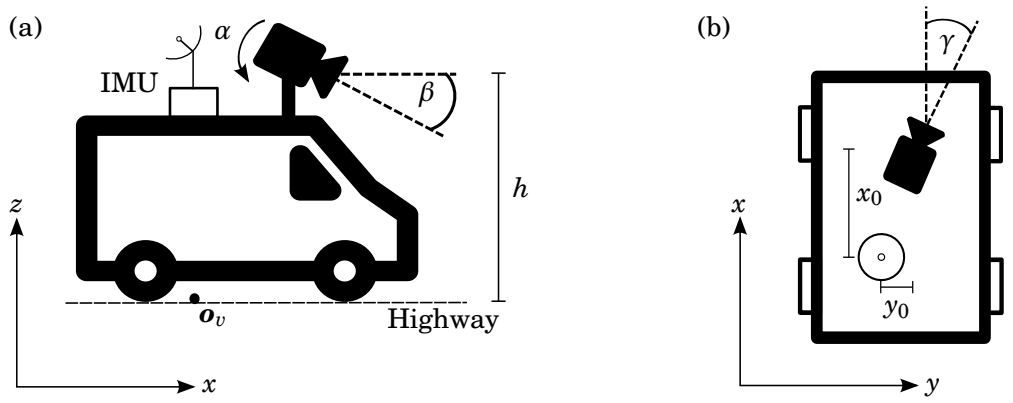



FIGURE 3.6. An illustration of the IMU and camera on the survey vehicle. (a) The vehicle in the $(x,z)$ plane: the IMU and camera are positioned on the vehicle at a height $h$ above the highway surface. The camera has a roll angle $\alpha$ and tilt angle $\beta$. The origin of the coordinate system relative to the vehicle, $\boldsymbol{o}_v$, lies directly underneath the IMU on the highway surface. (b) The offset of the camera relative to the IMU $(x_0, y_0)$ and the camera's pan angle $\gamma$ in the $(x,y)$ plane.

### 3.3.1 Camera Calibration

First, we develop a methodology to locate assets within the survey imagery. This is achieved by calibrating the camera on the vehicle, such that the pixel coordinates of an asset in a survey image may be computed, given the position of the asset relative to the camera.

Formally, during camera calibration, the camera parameters are recovered that project the arbitrary coordinate $\boldsymbol{p}_v$ to a pixel coordinate $(u,v)$ on an image taken by the camera. The camera parameters are categorised into intrinsic parameters; that is, the internal properties of the camera (e.g., the camera's focal lengths), and extrinsic parameters; that is, how the camera is configured on the vehicle relative to the IMU.

It is likely that different surveys will use different cameras, which may be installed onto the vehicle in a number of configurations[1]. Consequently, no assumptions may be made about the camera or its configuration (i.e., how it is installed onto the survey vehicle) during the survey of the A27 considered in this chapter. Therefore, to locate the assets within the survey images, all intrinsic and extrinsic camera parameters must be calibrated.

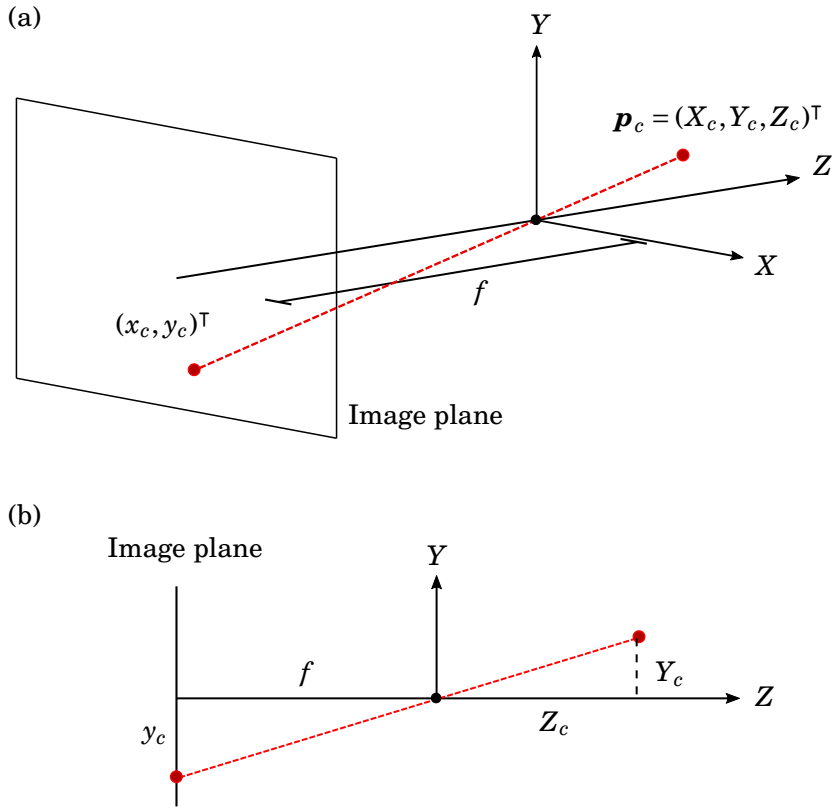



FIGURE 3.7. (a) The ideal pinhole camera model with focal length $f$. The coordinate $\boldsymbol{p}_c$ in the camera coordinate system is projected along the projection line (red) to the coordinate $(x_c, y_c)^\intercal$ in the image plane. (b) the projection in the $(Y, Z)$ plane.

### 3.3.2 Camera Model

Both the IMU and camera are positioned on the vehicle at height $h$. The camera is offset from the IMU at position $(x_0, y_0, h)^\intercal$ and has roll angle $\alpha$, tilt angle $\beta$, and pan angle $\gamma$; namely, rotations about the $x$-coordinate, $y$-coordinate, and $z$-coordinate axes respectively, shown in Figure 3.6. The camera on the vehicle is modelled as an ideal pinhole camera with no radial or tangential distortion, zero skew, and the centre of projection is assumed to be at the centre of the image plane [243]. A background of the pinhole camera model is now provided.

The pinhole camera model projects the three-dimensional coordinate relative to the camera, $\boldsymbol{p}_c = (X_c, Y_c, Z_c)^\intercal$, to the two-dimensional coordinate on the camera's physical image plane $(x_c, y_c)^\intercal$, see Figure 3.7. The coordinate $\boldsymbol{p}_c$ is defined in the coordinate system $(X, Y, Z)^\intercal$, where

[1]During the project, Jacobs (industry sponsor) provided imagery collected on a number of different drive-by surveys. When performing the calibration process, we found that even surveys of the same highway were performed with different cameras (with varying focal lengths) from one year to the next. Some pairs of surveys did use the same camera, however, they were found to have differing extrinsic parameters, suggesting that the camera had either been reinstalled between surveys, or a different vehicle with the same camera was used.

the $X$-coordinate and $Y$-coordinate axes respectively point along the width and height of the image plane, and the $Z$-coordinate axis points out of the image plane.

The projection line from $\boldsymbol{p}_c$ to $(x_c, y_c)^\intercal$ forms the hypotenuses of a pair of similar right-angled triangles in both the $(X,Z)$ and $(Y,Z)$ planes, see Figure 3.7(b). As each pair of triangles is similar, it follows that

$$\frac{x_c}{X_c} = \frac{f}{Z_c} \quad \text{and} \quad \frac{y_c}{Y_C} = \frac{f}{Z_c}$$

where $f$ is the camera focal length. By rearranging and collecting these two equations, we derive the non-homogeneous projection equation:

$$\begin{bmatrix} x_c \\ y_c \end{bmatrix} = \frac{f}{Z_c} \begin{bmatrix} X_c \\ Y_c \end{bmatrix}. \tag{3.2}$$

Given an image plane with width $l_x$ and height $l_y$ that contains $m \times n$ square pixels, the image plane coordinates $(x_c, y_c)^\intercal$ are converted to the pixel coordinates in the digital image $(u,v)^\intercal$ by

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \alpha_u x_c \\ \alpha_v y_c \end{bmatrix} = \frac{1}{Z_c} \begin{bmatrix} f_u X_c \\ f_v Y_c \end{bmatrix}. \tag{3.3}$$

Here, the scaling parameters $\alpha_u = m/l_x$ and $\alpha_v = n/l_y$ have units pixels per unit length and $f_u = \alpha_u f$ and $f_v = \alpha_v f$ are the focal lengths of the camera in pixels. Further, we may rewrite Equation 3.3 in the homogeneous form

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & 0 & 0 & 0 \\ 0 & f_v & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}. \tag{3.4}$$

The homogeneous form is converted back to the non-homogeneous form by multiplying the matrices on the right-hand side of Equation 3.4, and then dividing the first and second rows of the matrices on each side of the equation by their third row. The scaling constant $\lambda$ describes the ray of coordinates along the projection line that project onto the same pixel coordinate.

To project the coordinate $\boldsymbol{p}_v$ to the pixel coordinate $(u,v)^\intercal$ (and thus project the assets into the survey imagery) the transformation from the coordinate system relative to the vehicle, $(x_v, y_v, z_v)^\intercal$, to the coordinate system relative to the camera, $(X,Y,Z)^\intercal$, is substituted into Equation 3.4, such that

$$f(\boldsymbol{p}_v; \boldsymbol{c}) = \lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & 0 & 0 & 0 \\ 0 & f_v & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0}^\intercal & 1 \end{bmatrix} \begin{bmatrix} y_v \\ z_v \\ x_v \\ 1 \end{bmatrix}. \tag{3.5}$$

Here, the translation vector $\boldsymbol{t} = (y_0, h, x_0)^\intercal$ and the rotation matrix $\boldsymbol{R}$ is the product of the three rotation matrices through the camera's pan, tilt, and roll angles, about the $z$-coordinate, $y$-coordinate, and $x$-coordinate axes respectively. Henceforth, for the sake of brevity, the set of camera parameters $f_u$, $f_v$, $\alpha$, $\beta$, $\gamma$, $x_0$, $y_0$, and $h$ are represented by the vector $\boldsymbol{c}$.

### 3.3.3 Control Points

To estimate the camera parameters, $n$ control points are collected; that is, the coordinates $\tilde{\boldsymbol{p}}_{vi}$ relative to the vehicle of an object in an image, and the pixel coordinates $\tilde{\boldsymbol{u}}_i = (\tilde{u}_i, \tilde{v}_i)^\intercal$ of that object in the image for $i = 1, 2, \ldots, n$. The control points are a mixture of assets from the inventory and features on the highway surface. For the assets, the control point coordinates are obtained from the inventory, and the coordinates of surface features are found from the aerial footage of the highway, see Figure 3.8. The coordinates are collected in the world frame in $(e, n, z)^\intercal$ form and validated using the aerial imagery — this is particularly important for control points collected from assets, as their inventory entry may be incorrect. However, the asset height may not be validated from the aerial imagery, and thus the $z$ coordinate of each asset control point is validated from inspection of the survey imagery (and is thus a source of measurement error). Each control points collected from the highway surface has a z coordinate of zero. After validation, the control point coordinates are transformed to the coordinate system $(x, y, z)^\intercal$ relative to the vehicle.

The true camera parameters are then estimated by minimising the total reprojection error; that is, the Euclidean distance between the pixels computed by Equation 3.5 for each control point coordinate, and the corresponding control point pixel coordinates such that

$$\boldsymbol{c}^* = \arg\min_{\boldsymbol{c}} \sum_{i=1}^{n} |\tilde{\boldsymbol{u}}_i - f(\tilde{\boldsymbol{p}}_{vi}; \boldsymbol{c})|. \tag{3.6}$$

TABLE 3.1. The camera parameters found by Equation 3.6. The interior-point method is used to find the optimal set of camera parameters within the lower and upper bounds shown in the table. The directions of the offsets $x_0$ and $y_0$ are respectively towards the front of the survey vehicle and the central reservation of the highway.

| Camera parameter | Value | Lower bound | Upper bound |
|---|---|---|---|
| Focal Length $f_u$ [pixels] | 2058 | 0 | 10000 |
| Focal Length $f_v$ [pixels] | 2267 | 0 | 10000 |
| Roll $\alpha$ [degrees] | −1.604 | −30 | 30 |
| Tilt $\beta$ [degrees] | −5.004 | −30 | 30 |
| Pan $\gamma$ [degrees] | −0.052 | −30 | 30 |
| Offset $x_0$ [m] | 1.765 | −4 | 4 |
| Offset $y_0$ [m] | 0.088 | −2 | 2 |
| Height $h$ [m] | 2.301 | 0 | 10 |


FIGURE 3.8. A sample of control points collected to calibrate the camera on the vehicle. The control point coordinates for assets, such as the traffic sign at control point (a), are collected from the inventory. Control point coordinates for features on the highway surface, such as the lane marker at control point (b), are obtained from the aerial imagery. The green crosses show the manually collected pixel coordinate of each control point. The red circles show the reprojection of each control point coordinate after the calibration process.

The nonlinear optimisation is performed with an interior-point method. In practice, software routines such as Matlab's `fmincon` may be employed to perform the minimisation. The `fmincon` solver is robust, but needs to be constrained by an upper and lower bound on each of the camera parameters. Each control point provides two equations; one for the $u$ pixel coordinate and one for the $v$ pixel coordinate. Therefore, a minimum of four control points are required to solve for the eight camera parameters. However, by using a larger number of control points, and thus over-determining the system, the minimisation becomes more robust to measurement error in the control points (in the $z$ coordinate, for example).


FIGURE 3.9. Bounding boxes of five traffic signs constructed by the perimeter around the pixel coordinates computed by Equation 3.5 for the entire position of the asset relative to the vehicle. The asset width and height are extended by 0.15 m in each direction to ensure the assets appear within the computed bounding boxes.

In total, 24 control points from multiple images are used to calibrate the forward-facing camera on the survey vehicle. Table 3.1 shows the results of the calibration and the upper and lower bounds enforced on each parameter. It is assumed that the camera parameters are constant throughout the single survey considered in this chapter, and therefore the calibration process is only performed once.

### 3.3.4 Asset Localisation

With the survey camera calibrated, roadside assets may be automatically located within the survey images. Consider the asset positioned relative to the vehicle at $\boldsymbol{a}_v = (x_a, y_a, z_a)^\mathsf{T}$, with width $a_w$, height $a_h$ and mounting height $a_m$. The entire asset occupies positions

$$\bar{\boldsymbol{a}}_v = (x_a, \bar{y}_a, \bar{z}_a)^\mathsf{T} \tag{3.7}$$

where $y_a - a_w/2 \leq \bar{y}_a \leq y_a + a_w/2$ and $a_m \leq \bar{z}_a \leq a_m + a_h$. Further, a kind-of bounding box around the asset may be constructed by the perimeter around the pixel coordinates computed by $f(\bar{\boldsymbol{a}}_v; \boldsymbol{c}^*)$. However, the physical attributes for some the of the assets in the inventory were found (by manual inspection) to be inaccurate. Therefore, the asset width and height are extended by 0.015 m in each direction to ensure that the asset appears within the computed bounding box. This very simple capability already assists the analyst; instead of manually cross referencing the images with the inventory to find a suitable image of the asset, assets are now automatically located within the survey images and presented to the analyst, see Figure 3.9.

### 3.3.5 Asset Classification

The method now proceeds to classify the asset within the constructed bounding box and thus, confirm that its inventory record is correct.

The success of convolutional neural networks (CNNs) for image classification tasks is well documented [147, 231, 268]. To classify an image with a CNN, it is fed through a series of convolutional and pooling layers. The convolution layers perform feature extraction with sliding filters of different sizes across the inputted image; smaller filters extract local features in the image (pixel intensity differences, for example) whereas larger filters extract relatively global features such as shape and texture. The pooling layers down-sample the extracted features to improve feature translation invariance; that is, improving the robustness of the classifier to the position of important features in the image. The extracted features are then classified by a fully connected neural network (FCNN).

The CNN is parameterised by a set of weights that are optimised with a labelled training data set via the back-propagation algorithm [224]. During training, each labelled training image is fed forward through the CNN (with sub-optimal weights) and classified. The error, that is, the discrepancy between the classification and image label, is then computed and fed backwards through the CNN. The weights are then optimised via stochastic gradient descent [31] to minimise the total error across the entire training data set.

The architectures and weights for a number of state-of-the-art CNNs, trained on large labelled data sets across multiple GPUs, are freely available online [231]. However, we may only use these pre-trained CNNs to classify an image into one of the classes that the CNNs are originally trained on. To exploit the efficacy of CNNs, and the available pre-trained networks for asset classification, we use transfer learning [230] to retrain a pre-trained CNN with a smaller labelled data set specific to our task. The weights in the convolutional layers are frozen, as the features extracted in those layers are thought to generalise well to any image classification task. The FCNN is then modified so that the number of nodes in the final classification layer is the desired number of

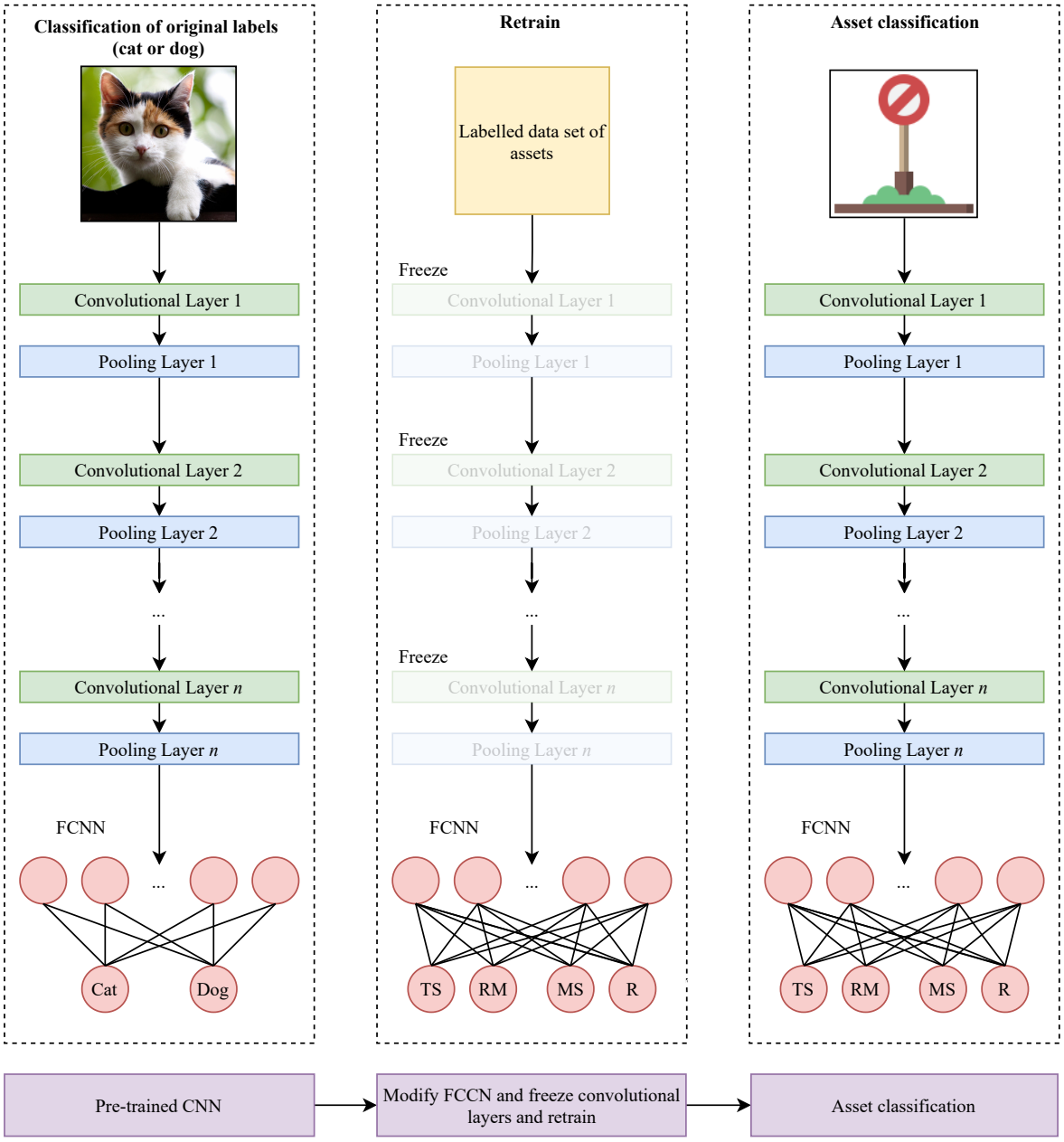



Figure 3.10: An illustration of the transfer learning process used for asset classification. The FCNN in a pre-trained CNN is modified to classify an image as a traffic sign (TS), reference marker post (RM), matrix sign (MS) or random (R). During training, the weights in the convolutional layers are frozen, whereas the weights in the FCNN are re-trained with a labelled training data set of assets.

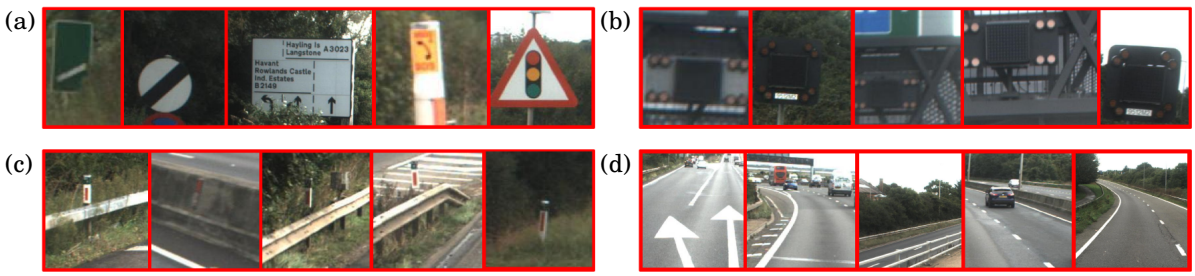



FIGURE 3.11. Examples of the training data set for each class. (a) Traffic signs, (b) matrix signs, (c) reference markers and (d) the empty (random images) class.

classes and retrained with our labelled data set. The transfer learning process is illustrated in Figure 3.10.

We retrain a CNN to classify an image as one of the three asset types considered in this chapter and an additional random class so that the CNN might detect when there is no asset present. This enables the decision support system to find incorrect asset entries in the inventory. To retrain the CNN, we construct a training data set with 60% of the assets in the inventory. Each asset is located within multiple images (with Equation 3.5) and the thumbnail within its bounding box is cropped out of the image. In total, 1,000 thumbnails of each asset type and the additional random class are extracted from the survey images. Furthermore, data augmentation is performed on the training data; that is, copies of the asset thumbnails are randomly sheered, stretched, squeezed, and rotated to create a larger data set. Example thumbnails of each class are shown in Figure 3.11.

To test the classification accuracy of the CNN during training, 20% of the training data is withheld as a validation set. The remaining 40% of assets are taken forward as a test set. The effectiveness of our system will be determined by considering which of those assets in the test set are identified as correct or requiring further manual inspection. A summary of the labelled training, validation, and test data sets is provided in Table 3.2.

TABLE 3.2. Number of each asset type used to train and test the decision support system. The training and validation sets are constructed from multiple views of 60% of the assets in the inventory. The remaining 40% of the assets form a test set to determine to effectiveness of the system.

| Asset type | Train | Validation | Test |
|---|---|---|---|
| Traffic sign | 800 | 200 | 149 |
| Reference marker | 800 | 200 | 69 |
| Matrix sign | 800 | 200 | 18 |
| Random | 800 | 200 | 0 |
| Total | 3200 | 800 | 236 |

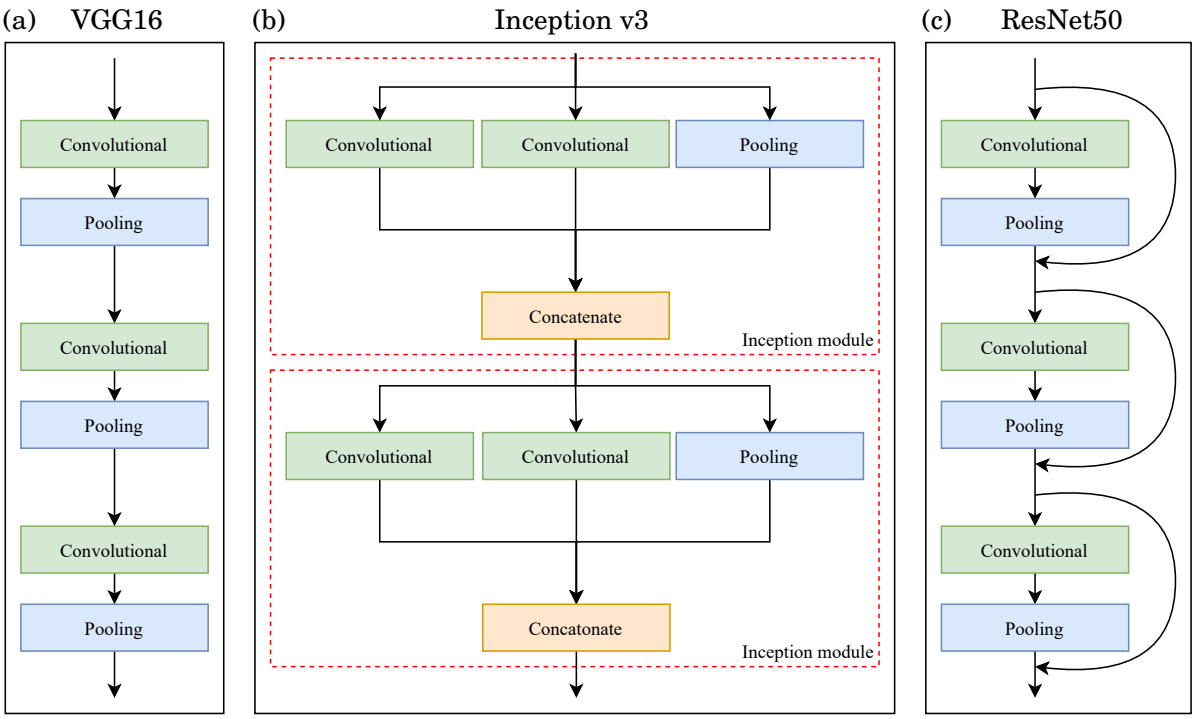



FIGURE 3.12. Pre-trained CNNs retrained for asset classification. (a) The VGG16 classifier has a conventional CNN architecture consisting of convolutional and pooling layers in series. (b) The Inception v3 architecture parallelises convolutional and pooling layers in Inception modules to reduce complexity and improve scalability. (c) The ResNet50 architecture uses skip connections to overcome a vanishing or exploding gradient problem.

There are a number of available pre-trained deep CNNs we may chose to retrain for our roadside asset monitoring system — largely as a result of the large labelled data sets available to researchers. The ImageNet database is one such data set first constructed by the ImageNet project in 2009 [55]. Through crowdsourced labelling, the project has created a database of nearly 15 million images annotated with class labels and bounding boxes. The project holds the annual ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in which researchers are invited to train and test their image classifiers on a trimmed version of the ImageNet data set with $1,000$ classes. The competition not only provides a benchmark for researchers to test their own classifiers, but also allows the community to compare against and improve upon other submissions.

The potential for using CNNs for image classification was first demonstrated at the 2012 ILSVRC, when AlexNet [147] outperformed existing image classification methods (recording a 10.8% classification improvement on second place), kick-starting the deep learning revolution. Subsequent ILSVRCs have accelerated the development of CNNs; in the 2017 competition 29 of

TABLE 3.3. The architecture and number of weights of each CNN retrained for asset classification and the accuracy achieved on the withheld validation data set.

| CNN | Architecture | Weights [million] | Asset validation accuracy [%] |
|---|---|---|---|
| VGG16 | Conventional | 138 | 98 |
| Inception v3 | Inception modules | 23 | 94 |
| ResNet50 | Skip connections | 25 | 98 |

the 38 submissions achieved at least a 95% top-five classification accuracy (i.e., the true label is one of the five most probable classifications) [210].

Thanks to projects such as ImageNet, researchers have developed, trained, and published highly effective CNNs. We consider three notable pre-trained classifiers to use as a base CNN for transfer learning, see Figure 3.12, which are now described.

The VGG16 classifier developed by Simonyan and Zisserman [231] was runner-up at ILSVRC 2014. VGG16 has a conventional architecture consisting of convolutional and pooling layers in series. The key feature of the VGG16 classifier is its depth; the CNN has 13 convolutional layers, whereas AlexNet (the largest existing CNN at the time) has five. This depth enables VGG16 to extract rich visual features, but creates a large computational overhead when training the network.

The Inception network presented by Szegedy et al. [244] employs a novel sub-architecture within the CNN to reduce its complexity and improve scalability without impacting on classification performance. Previous improvements to CNNs had been focused on increasing their depth (to extract more features), however, the authors identified that improvements might instead be achieved by increasing their 'width' and thus designed the Inception module. Instead of a conventional set of convolutional and pooling layers in series, the inception module uses multiple convolutional layers with different filter sizes to extract features at various scales (both local and global), and a pooling layer in parallel. The set of extracted features and downsized inputs are then concatenated and fed into the next layer in the network. The top performing CNN with the Inception module, Inception v3 achieved a 94% top-five classification accuracy on the 2012 ILSVRC data set, improving state-of-the-art performance with at least five times less weights.

The ResNet50 (residual network) architecture developed by Microsoft won the 2015 ILSVRC [99]. The authors challenged the previously accepted idea that CNNs perform better with increased depth (or width), and thus more extracted features. Through experiment the authors found a surprising result — classification performance decreases as CNNs become too deep. This is not caused by over-fitting, but a 'vanishing' or 'exploding' gradient problem in which the error propagated backwards through large CNNs becomes very small or very large and thus the weights are poorly optimised. To overcome this problem, ResNet50 uses 'skip connections' that allow the classifier to propagate the errors backwards through a smaller number of layers while preserving the CNN's deep architecture.

We modify the FCNN of each of the three CNNs and retrain them with the labelled training

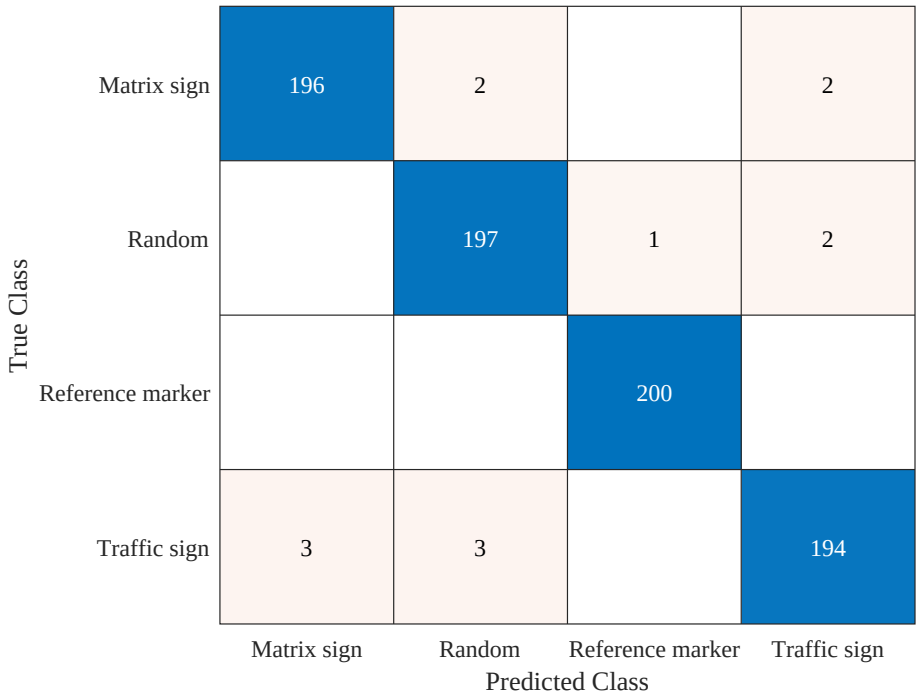



FIGURE 3.13. A confusion matrix that illustrates the inter-class classification accuracy of the CNN on the validation set. The true and predicted asset type are respectively represented by the rows and columns of the matrix. Overall, a validation accuracy of 98% is achieved.

set of assets. Each CNN is trained for 20 epochs with a batch size of 32 and a learning rate of $1 \times 10^{-4}$. ResNet50 and VGG16 both achieve a 98% accuracy on the withheld validation data set whereas Inception v3 achieves a slightly lower 94%, see Table 3.3. We therefore choose the retrained ResNet50 architecture for our decision support system as it is smaller and therefore less computationally intensive than VGG16. A confusion matrix that shows the inter-class prediction performance on the withheld validation set is provided in Figure 3.13.

Although not used for our asset classification problem (as a 98% accuracy on the validation set is already achieved), an optional final training step is to unfreeze some or all of the convolutional layers and fine-tune their weights with a small withheld portion of the training data. Fine-tuning has successfully been used to train CNNs for a number of image classification tasks, including biomedical image analysis [272] and fine art classification [40].

## 3.4 Rapid Inventory Updating

We now test our decision support system by considering those assets in the test set that are verified and those identified for further manual inspection. Each asset is considered individually.

First, the survey image for which the asset is closest to the survey vehicle and contains the asset's bounding box is found. The asset thumbnail is then computed and cropped out of the image and classified by the CNN. For those assets classified as the asset type recorded in the inventory, we verify that the assets are recorded correctly in the inventory. On the other hand, those assets that are classified as a different asset type than that recorded in the inventory, or as the empty class, are identified as assets for further manual inspection.

The inventory may be incorrect if the asset has been moved or removed from the highway, but the inventory has not been updated correspondingly, or the asset's geographical or physical attributes are incorrect in the inventory. In addition, an asset is identified for further manual inspection if the view of the asset is impeded, or if the CNN classifies the asset incorrectly. The system does not determine the reason why the inventory record for an asset has not been verified. Rather, the relatively small number of assets for which this is the case are rapidly identified and presented to the analyst, who may then correspondingly update the inventory. Table 3.4 shows the number of assets that are verified and the reason for incorrect classification. Overall, 91% of roadside assets in the test set are verified automatically. Examples of the system in operation are shown in Figure 3.14.

## 3.5 Discussion

The system automatically verifies 91% of assets in the withheld test set, which would otherwise be manually inspected by an analyst. This reduction in the analyst's manual workload demonstrates the beginnings of a promising operational asset monitoring capability. The system does not consider whether the asset has been correctly verified. However, if an asset is correctly localised (via the calibrated camera) and classified by the CNN (which achieved a 98% classification accuracy on the validation set), it is highly like that asset is recorded correctly in the inventory. The results show that the system performs well for all asset types considered. However, of the 21 assets identified for further manual inspection, 13 were the result of an incorrect classification by the CNN or an impeded view of the asset. Four such assets are shown in Figure 3.15.

An improved system should aim to address these incorrect inventory verifications. The

TABLE 3.4. The number of assets that are verified, and the reason for an incorrect classification for each asset type. Overall, 91% of the assets are verified automatically, thus greatly reducing the workload of the analyst.

| Asset type | Number | Verified | Percentage [%] | CNN error | Impeded | Incorrect inventory |
|---|---|---|---|---|---|---|
| Traffic sign | 149 | 136 | 91 | 5 | 2 | 6 |
| Reference markers | 69 | 64 | 93 | 2 | 2 | 1 |
| Matrix sign | 18 | 15 | 83 | 2 | 0 | 1 |
| Total | 236 | 215 | **91** | 9 | 4 | 8 |

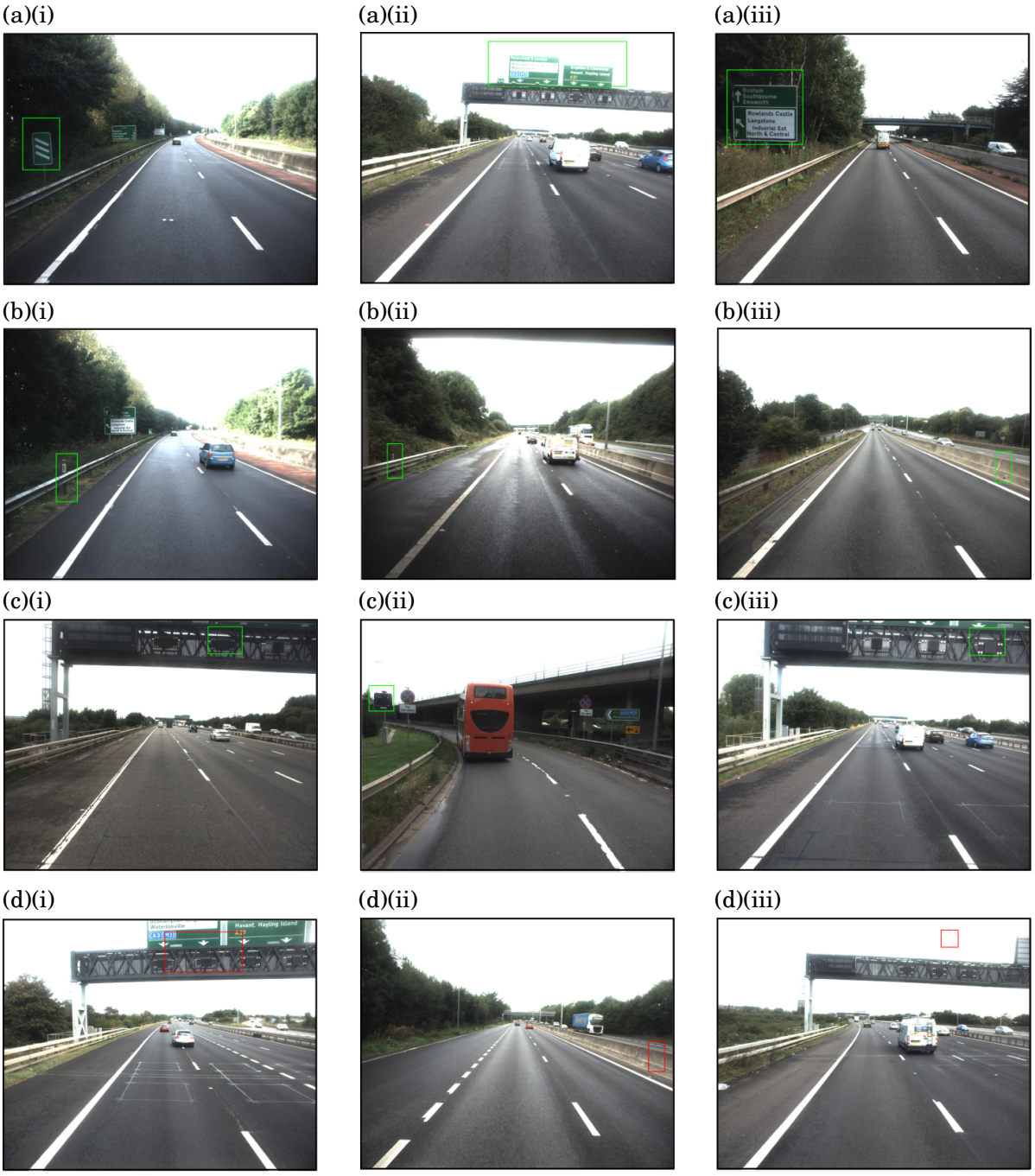



FIGURE 3.14. Examples of our system in operation. (a) Traffic signs, (b) reference markers and (c) matrix signs that are localised and classified as the correct asset type given in the inventory. Consequently we automatically verify the inventory entry for those assets. (d) Assets that are classified as a different asset type, or random. A traffic sign recorded with an incorrect height in the inventory, and thus classified as a matrix sign, is shown in panel (d)(i). Panels (d)(ii) and (d)(iii) respectively show a reference marker recorded with an incorrect position and a matrix sign recorded with an incorrect height: both assets are classified as random by the CNN.

incorrect CNN classifications are likely to be overcome by using a larger labelled training set. A training set containing more asset thumbnails from a number of view points, both close to and far away from the camera, and under different lighting conditions, is likely to improve the classification performance of the CNN.

Usually, an impeded view of an asset is caused by another vehicle on the highway (see Figure 3.15). Therefore, a further improved system might only consider a survey image of an asset when there are no vehicles between the camera and the considered asset. There are a number of available pre-trained vehicle detection networks, such as the YOL0v3 [214] architecture (see page 90) that a future system could use to compute automatically the bounding boxes of other vehicles in the survey imagery. Further, the performance of the system might be improved by considering multiple views of the asset. For example, if the closest survey image to an asset results in an incorrect inventory verification, then the second closest survey image might also be considered to confirm the decision.

There are several other potential refinements that might improve our method. Currently, three asset types have been considered; however, all roadside assets might be monitored by the system by training the CNN with a labelled data set containing all asset types. Should the survey vehicle be equipped with a more sophisticated IMU, a more accurate heading direction might be computed, and thus, the heading correction performed in Section 3.2.1 would not be necessary. In addition, using a camera with known intrinsic parameters might simplify the camera calibration method as only the extrinsic parameters would be unknown. Moreover, the camera calibration process may be completely unnecessary should the extrinsic parameters (that is, how the camera is configured on the survey vehicle relative the IMU) be measured before or after the survey. The camera parameters are assumed to be constant throughout the survey and thus the camera is only calibrated once. However, it is possible that the camera angles may change should the vehicle jolt. Therefore a sequential calibration method might be employed, so that the camera parameters are continually updated.

Our system has several notable differences when compared with pre-existing existing computer vision-based TAM systems. Typically, assets (commonly traffic signs) are automatically detected in images and subsequently processed [22, 258]. In contrast, our system employs a calibrated camera to project the assets (regardless of asset type) into the survey imagery and thus our system is more robust. On the other hand, along with a number of other systems [148, 274], we also exploit the efficacy of CNNs on image classification problems for asset classification.

Second, existing TAM systems consider primarily a data source from an external sensor (camera or LIDAR, for example) and subsequently detect, classify, or map assets. However, our system is built upon the inventory; that is, we firstly consider each asset as it is recorded in the inventory, and subsequently employ computer vision-based methods to verify its entry. Therefore, unlike existing systems, our method reflects the role of an analyst and can thus provide decision support within the current asset monitoring process. Furthermore, to the best of my knowledge,


FIGURE 3.15. (a) Traffic signs and (b) reference markers that are correctly recorded in the inventory but are identified by the system for further manual inspection. The assets in the (i) panels are incorrectly classified by the CNN whereas the assets in (ii) panels are impeded.

our system is the first TAM system that automatically verifies both the asset type and position recorded in the inventory, from a single camera and IMU.

### 3.5.1 System Limitations and Potential Future Improvements

There are several limitations in the currently presented system, and therefore opportunities for further research and development to move the system towards an operational asset monitoring capability.

Currently, the system does not consider the asset's condition. Automatic asset condition evaluation is a difficult problem to solve. As described in Section 3.5, a number of methods have been proposed to model the pavement surface condition, although they usually rely on illuminated

imagery captured by a downwards-facing camera on a specialised vehicle.

However, previous work on automatic condition assessment of roadside assets is more limited. Notable existing methods include systems that automatically evaluate traffic sign retroreflectivity [5] and determine whether a traffic sign is orientated correctly [53]. Yet, to the best of my knowledge, a general framework that might rapidly evaluate the condition of any asset — a capability described by an industry practitioner as 'the holy grail of transportation asset management' (Jacobs, personal communication) — has not yet been developed. In this chapter, we have demonstrated the efficacy of deep CNNs on image classification tasks and thus they are likely to be at the centre of any future computer vision-based asset condition evaluation tool. However, a large labelled data set of images, containing each asset type in every condition, will need to be constructed. As the number of assets in bad condition is likely to be small compared with the number that are in good condition, building a balanced training data set is likely to be a difficult task.

The system only considers those assets on the highway that are recorded in the inventory (correctly or incorrectly). Therefore a future system might also be extended to consider assets that are not recorded in the inventory at all; assets that have recently been installed, for example. To achieve this, improving the CNN so that assets are localized and classified from the whole scene of the highway (providing a bounding box and asset type directly from the imagery), rather than the cropped asset thumbnails, is likely to be the way forward. In this case, the detected bounding boxes may then be compared against those computed from the inventory (via the calibrated camera) to identify assets on the highway that are missing in the inventory.

### 3.5.2 Prototype Asset Monitoring Tool

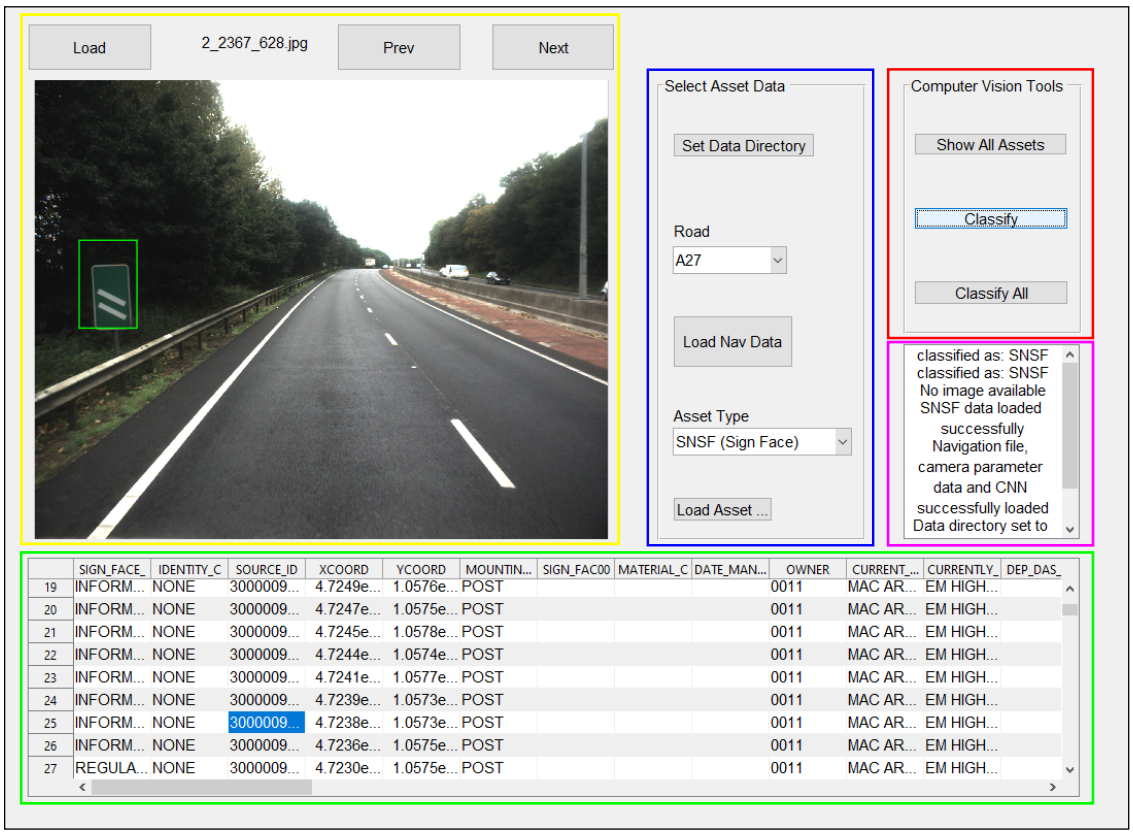
Figure 3.16 shows a prototype asset monitoring tool that demonstrates how the computer vision methods developed in this chapter might assist an analyst in practice. First, the survey imagery and its accompanying IMU data (which is processed via the central difference scheme described in Section 3.2.1), the asset inventory, and the calibrated camera parameters are loaded into the tool. Then, the analyst may either:

1. manually step through the inventory updating process described in Section 3.4 for each asset; that is, automatically compute the bounding box (via the calibrated camera) and then classify the thumbnail within the bounding box (with a trained CNN), thus, verifying its inventory record, see Figure 3.16(a); or,

2. run option 1 for the entire inventory of a chosen asset type (e.g., traffic signs), see Figure 3.16(b), thus rapidly identifying those assets that require further manual inspection, see Figure 3.16(c).

Once an asset has been (automatically) identified for manual inspection, its inventory record may then need to be updated. However, currently, our prototype tool does not allow the analyst to

(a)




| | SIGN_FACE_ | IDENTITY_C | SOURCE_ID | XCOORD | YCOORD | MOUNTIN... | SIGN_FAC00 | MATERIAL_C | DATE_MAN... | OWNER | CURRENT_... | CURRENTLY_ | DEP_DAS_ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 19 | INFORM... | NONE | 3000009... | 4.7249e... | 1.0576e... | POST | | | | 0011 | MAC AR... | EM HIGH... | |
| 20 | INFORM... | NONE | 3000009... | 4.7247e... | 1.0575e... | POST | | | | 0011 | MAC AR... | EM HIGH... | |
| 21 | INFORM... | NONE | 3000009... | 4.7245e... | 1.0578e... | POST | | | | 0011 | MAC AR... | EM HIGH... | |
| 22 | INFORM... | NONE | 3000009... | 4.7244e... | 1.0574e... | POST | | | | 0011 | MAC AR... | EM HIGH... | |
| 23 | INFORM... | NONE | 3000009... | 4.7241e... | 1.0577e... | POST | | | | 0011 | MAC AR... | EM HIGH... | |
| 24 | INFORM... | NONE | 3000009... | 4.7239e... | 1.0573e... | POST | | | | 0011 | MAC AR... | EM HIGH... | |
| 25 | INFORM... | NONE | 3000009... | 4.7238e... | 1.0573e... | POST | | | | 0011 | MAC AR... | EM HIGH... | |
| 26 | INFORM... | NONE | 3000009... | 4.7236e... | 1.0575e... | POST | | | | 0011 | MAC AR... | EM HIGH... | |
| 27 | REGULA... | NONE | 3000009... | 4.7230e... | 1.0575e... | POST | | | | 0011 | MAC AR... | EM HIGH... | |

(b)

| | SIGN_FACE_ | IDENTITY_C | SOURCE_ID | XCOORD | YCOORD | MOUNTING_M |
|---|---|---|---|---|---|---|
| 19 | INFORMATORY | NONE | 3000009361 | 4.7249e+05 | 1.0576e+05 | POST |
| 20 | INFORMATORY | NONE | 3000009359 | 4.7247e+05 | 1.0575e+05 | POST |
| 21 | INFORMATORY | NONE | 3000009364 | 4.7245e+05 | 1.0578e+05 | POST |
| 22 | INFORMATORY | NONE | 3000009356 | 4.7244e+05 | 1.0574e+05 | POST |
| 23 | INFORMATORY | NONE | 3000009363 | 4.7241e+05 | 1.0577e+05 | POST |
| 24 | INFORMATORY | NONE | 3000009351 | 4.7239e+05 | 1.0573e+05 | POST |
| 25 | INFORMATORY | NONE | 3000009352 | 4.7238e+05 | 1.0573e+05 | POST |
| 26 | INFORMATORY | NONE | 3000009360 | 4.7236e+05 | 1.0575e+05 | POST |
| 27 | REGULATORY | NONE | 3000009358 | 4.7230e+05 | 1.0575e+05 | POST |
| 28 | REGULATORY | NONE | 3000009350 | 4.7230e+05 | 1.0572e+05 | POST |
| 29 | INFORMATORY | NONE | 3000009349 | 4.7228e+05 | 1.0572e+05 | POST |
| 30 | INFORMATORY | NONE | 3000009357 | 4.7226e+05 | 1.0574e+05 | POST |
| 31 | WARNING | NONE | 3000009354 | 4.7196e+05 | 1.0574e+05 | POST |
| 32 | WARNING | NONE | 3000009355 | 4.7196e+05 | 1.0574e+05 | POST |
| 33 | REGULATORY | NONE | 3000009353 | 4.7196e+05 | 1.0574e+05 | POST |
| 34 | REGULATORY | NONE | 3000009375 | 4.7187e+05 | 1.0582e+05 | LAMP POST |
| 35 | INFORMATORY | NONE | 3000009362 | 4.7187e+05 | 1.0576e+05 | POST |
| 36 | REGULATORY | NONE | 3000009377 | 4.7184e+05 | 1.0583e+05 | POST |
| 37 | INFORMATORY | NONE | 3000009365 | 4.7181e+05 | 1.0578e+05 | POST |
| 38 | REGULATORY | NONE | 3000009374 | 4.7180e+05 | 1.0582e+05 | LAMP POST |
| 39 | REGULATORY | NONE | 3000009373 | 4.7180e+05 | 1.0582e+05 | LAMP POST |
| 40 | WARNING | NONE | 3000009370 | 4.7179e+05 | 1.0580e+05 | POST |
| 41 | WARNING | NONE | 3000009371 | 4.7179e+05 | 1.0580e+05 | POST |

(c)

FIGURE 3.16. Prototype asset monitoring tool. (a) Main tool interface. (Yellow) survey imagery, (blue) inventory and IMU data loading, (green) inventory, (magenta) output log, (red) computer vision tools. Each asset's bounding box is computed and displayed by clicking on its inventory record, which may then classified by the CNN via the 'classify' button. Clicking the 'classify all' button verifies the entire inventory for a chosen asset type. (b) Inventory of traffic signs that are automatically verified (green) or identified for further manual inspection (red, e.g., traffic sign 37). (c) Bounding box computed for traffic sign 37: the asset appears to have been removed from the highway.

update the inventory; it is only loaded (so that each asset's physical properties may be used to compute its bounding box) and displayed in the main tool window (Figure 3.16(a), green box). As such, embedding an additional interface into the tool, so that an analyst may directly update the inventory would mature the prototype tool towards a deployable capability.

### 3.5.3 Human-Computer Interaction

The system (and accompanying prototype tool) presented in this chapter is a semi-automated roadside asset monitoring capability; that is, it combines an automatic process performed by a computer (assets are automatically identified for manual inspection) and a manual process performed by a human (the inventory is manually updated) [197, 275]. As computers and software applications proliferate throughout organisations, systems (such as ours) with a level of automation, will continue to streamline and make business processes (such as asset management) more efficient.

Ultimately, the goal of automation is to save time (i.e., to perform a task several times faster than a human). However, human performance is usually the gold standard and fully automatic systems rarely improve the quality of a task [89]. Semi-automated systems aim to strike a balance between time and performance — a human might validate the automated components of a system, for example. However, in some cases, it has been shown that semi-automatic systems can in fact decrease the performance of a human, compared with a similar task in a manual setup [196]. This effect is widely studied in the human factors research area of human-computer interaction (HCI) [141].

In a study presented by Ghani and Deshpande [79], 149 professionals were interviewed to determine the relationship between the experience of computer users and the scope of the task that they perform. The authors defined that optimal task performance is obtained by maximising the user 'flow'; that is, the user's concentration and enjoyment level. From the interviews it was determined that the greatest contributor to user flow is the perceived sense of agency (SOA) — humans perform better when they feel in control of the tasks they are set [176]. A popular measure of user flow (and therefore agency) is the perceived length of time between a user action and its outcome. Users with a high SOA perceive this 'time of action' to be shorter than those with a lower SOA. This result is known as intentional binding (IB) [95] and has been used to explore user SOA for a number of case-study systems with varying degrees of automation. For example, a flight simulator and a computer-assisted point and click task are respectively considered by Berberian et al. [27] and Coyle et al. [48]. In both studies the same conclusion is reached — a system with increased autonomy decreases the user's SOA.

Therefore, an analyst using the tools developed in this chapter might inaccurately update the inventory for those small number of assets that are not verified by our system. As such, the interface design and HCI will require careful consideration when deploying an operational system based on our work or prototype tool. For example, a system where the analyst must confirm all or

some of the automatically verified assets (and thus reduce the level of automation) may result in a more accurate inventory overall, as well as reducing the time taken to complete the task.

## 3.6 Improving the System Robustness in GPS-denied Environments

The system as presented in this chapter relies on a GPS-enabled IMU installed on the survey vehicle and therefore may perform poorly in closed environments such as tunnels or urban canyons where a strong satellite signal is unavailable. In this chapter so far have not considered such environments and the system is assumed to be in operation on an open, multi-lane highway. However, to make the system more robust, the data provided by the IMU might be fused with another source to estimate the vehicle's position and heading direction in closed environments. Such an estimate might be provided by a wheel odometer (assuming the vehicle is travelling in a broadly straight line) installed on the survey vehicle, or alternatively, via a computer vision-based method in which the relative camera motion is estimated purely from the imagery.

A position estimation computed directly from the survey imagery is an attractive capability as no additional sensors than those already installed on the survey vehicle are required. As such, we now explore how the active research area of monocular (i.e., not stereo) visual simultaneous localisation and mapping (vSLAM) might be employed to estimate the position and heading direction of the survey vehicle, and thus, make a future automatic asset monitoring capability more robust.

The remainder of this section is structured as follows. In Section 3.6.1 we provide a background on vSLAM systems. In Section 3.6.2 we identify two shortcomings when using an existing state-of-the-art vSLAM system with the survey imagery. The existing system is then respectively modified and deployed in Sections 3.6.3 and 3.6.4. Finally, in Section 3.6.5 we discuss how the vSLAM system might be embedded within our asset monitoring capability.

### 3.6.1 vSLAM Background

Given a sequence of images, vSLAM systems compute the full six degree-of-freedom pose (position and orientation) of the camera as each image is taken and concurrently construct a map (the three-dimensional coordinates) of the world around the camera [100, 245].

Feature-based vSLAM systems are built upon reliably extracting feature matches; that is, the pixel coordinates, or keypoint, of the same feature (corners for example) in multiple images [21]. Features are generated by first transforming the local image patch around a keypoint into a descriptor vector, then matched using a nearest neighbours-based algorithm to match similar descriptors [98]. Initially, scale-invariant feature transform (SIFT) features, that compare Histogram of Oriented Gradient-based (HOG) descriptors [164], and speeded-up robust features (SURF) that compare Haar wavelet distribution-based descriptors [25] were extracted. However,

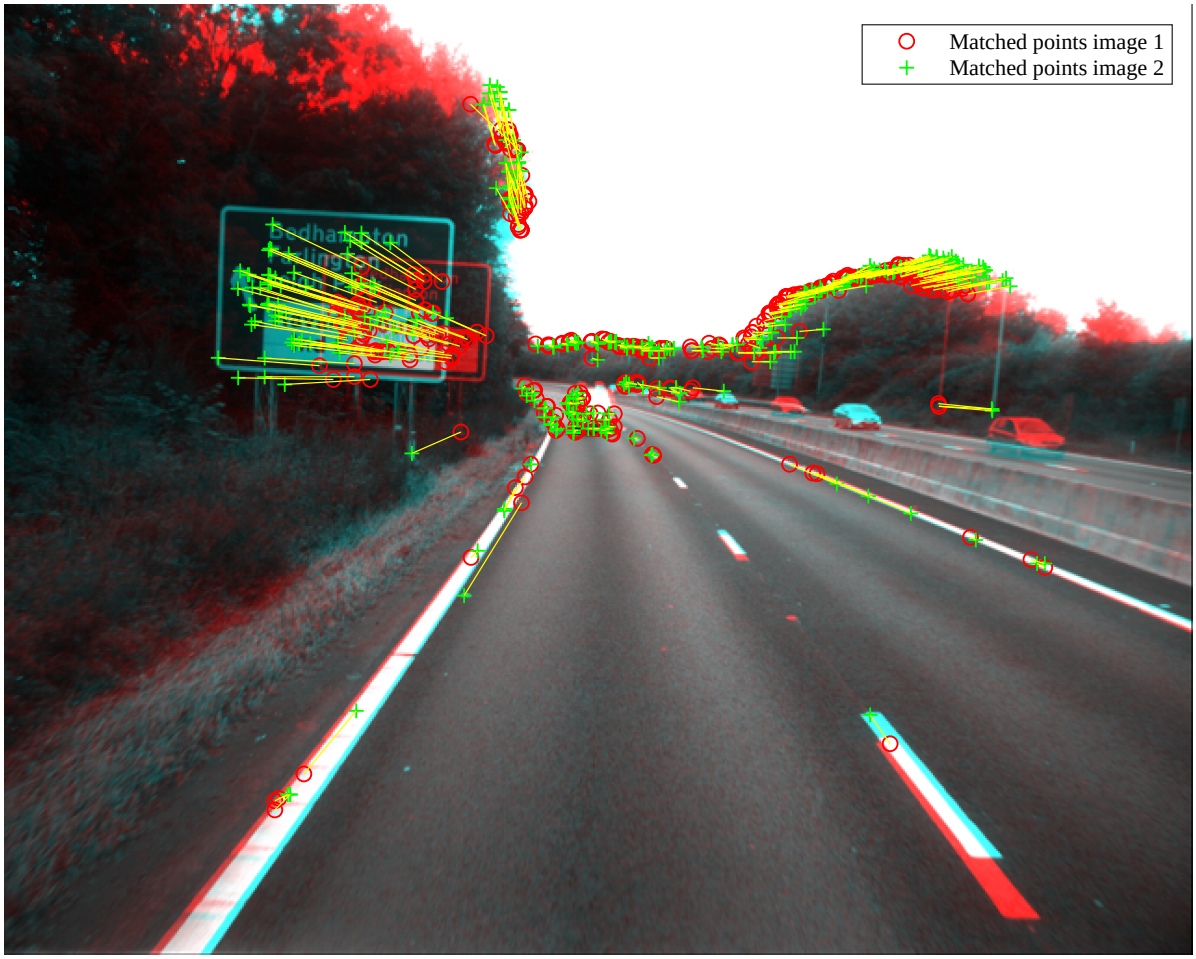



FIGURE 3.17. Matched ORB features in two survey images taken of the same section of highway. The two images are overlaid to show the relative position of each feature in image 1 (red circles) and in image 2 (green crosses).

modern systems use features that are faster to compute and which are invariant to rotation and noise, such as oriented FAST and rotated BRIEF (ORB) features [98, 223]. A sample of ORB feature matches extracted from the survey imagery is shown in Figure 3.17.

With a sufficient number of features and images, Bundle Adjustment (BA) [249] is performed to compute the camera pose as each image is taken (localisation), and the three-dimensional coordinate in the world of each feature (mapping). Typically, BA is used in conjunction with a random sample consensus (RANSAC) [73] routine to remove anomalous feature matches. Specifically, given $n$ features extracted across $m$ images, BA minimises the reprojection error

$$\underset{\boldsymbol{x}_i,\boldsymbol{R}_j,\boldsymbol{t}_j}{\arg\min} \sum_{i}^{n}\sum_{j}^{m} \delta_{ij} |\pi(\boldsymbol{x}_i,\boldsymbol{R}_j,\boldsymbol{t}_j) - \boldsymbol{u}_{ij}|. \tag{3.8}$$

Here, $\boldsymbol{x}_i = (x_i, y_i, z_i)^\mathsf{T}$ denotes the three-dimensional coordinate of the $i^{\text{th}}$ feature, $\boldsymbol{R}_j$ and $\boldsymbol{t}_j$

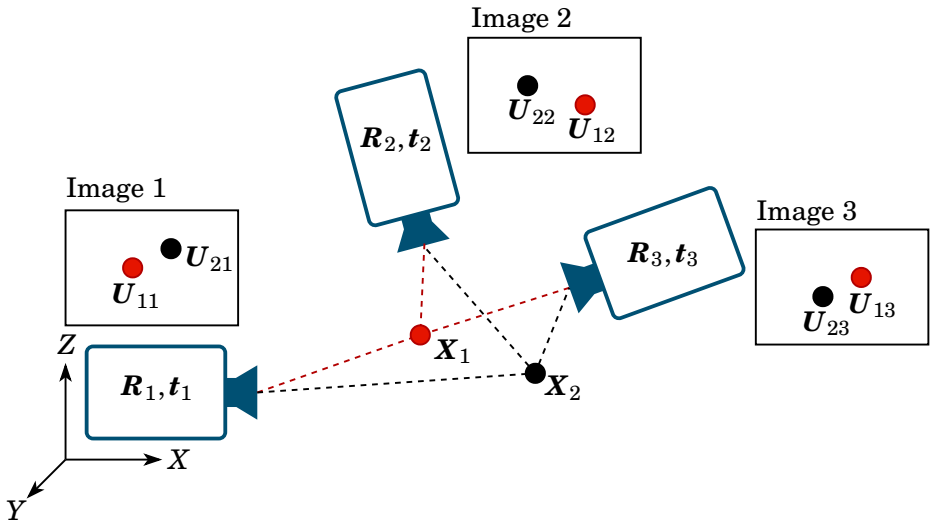



FIGURE 3.18. An illustration of a feature-based visual SLAM system. The two-dimensional pixel coordinates of features, $\boldsymbol{U}$ are matched across multiple images. With a sufficient number of matches and images, vSLAM systems simultaneously compute the three-dimensional coordinate of each feature in the world (mapping), $\boldsymbol{X}$, and the orientation and position (pose), $\boldsymbol{R}$ and $\boldsymbol{t}$, of the camera as each image is taken (localisation).

denote respectively the orientation (as a rotation matrix) and position vector of the camera as $j^{\text{th}}$ image is taken, and $\boldsymbol{u}_{ij} = (u_{ij}, v_{ij})^\intercal$ denotes the two-dimensional pixel coordinate of the $i^{\text{th}}$ feature in the $j^{\text{th}}$ image. The constant $\delta_{ij} = 1$ if the $i^{\text{th}}$ feature appears in the $j^{\text{th}}$ image and 0 otherwise, and the non-homogeneous projection function

$$\pi(\boldsymbol{x}, \boldsymbol{R}, \boldsymbol{t}) = \frac{1}{z'} \begin{bmatrix} f_u x' \\ f_v y' \end{bmatrix} \tag{3.9}$$

is derived in page 67. The coordinates $\boldsymbol{x}' = \boldsymbol{R}(\boldsymbol{x} - \boldsymbol{t})$ are transformed from the world coordinate system to the coordinate system relative to the camera (see page 66). An illustration of a feature-based vSLAM system is provided in Figure 3.18.

Historically, vSLAM systems were filter-based and the camera pose and map were updated for each new image. Such approaches have previously employed Kalman filters [44, 50, 51] and particle filters [81], for example. However, filter-based approaches can be inefficient if consecutive images are similar and therefore uninformative. Alternatively, keyframe-based vSLAM approaches update the map using selected, informative images (keyframes) [155]. Strasdat et al. [242] showed that keyframe-based approaches outperform filter-based approaches for the same computational cost and are now the de facto method for monocular vSLAM systems.

Real time performance for keyframe-based vSLAM was first achieved by the innovative parallel tracking and mapping (PTAM) system [145]. Their key contribution was to compute the camera poses and map in two parallel threads, allowing for real time performance on a dual-core

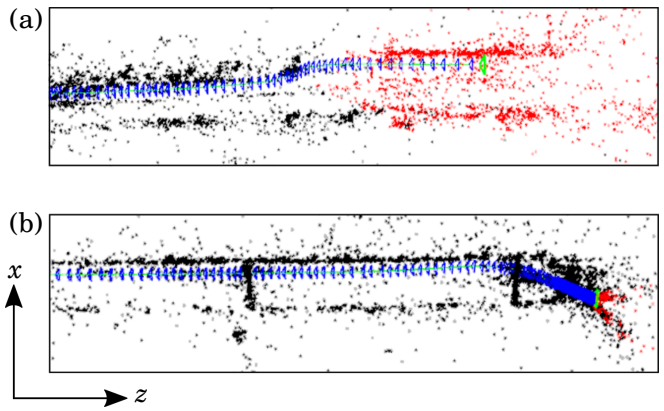



FIGURE 3.19. Two examples of poor camera poses and maps computed by ORB-SLAM for a straight segment of the A27. (a) The estimated camera pose and map of the highway shift to the left of the direction of travel, and (b) appear to follow the highway, however become *bunched*, possibly due to low parallax. Each panel shows a broadly top-down view of the camera pose on the highway as the survey vehicle travels from left to right. The panels show (blue) the estimated camera pose, (black) the generated map, (red) the map that is currently being estimated by the local BA, and (green) the current camera pose estimate, and co-visibility graph (which indicates how many feature correspondences are in each image pair).

computer. Later, PTAM was improved upon by the ORB-SLAM system [178] in which a third thread is proposed to allow for loop detection (i.e., the camera has returned to a previous location) and closing in parallel.

Monocular vSLAM systems have a number of limitations. The camera poses and map are computed relative to the first image and to a scale. Furthermore, vSLAM can suffer from drift, in which the global error grows despite a locally accurate estimation of the camera pose and map. In addition, it is assumed that camera's intrinsic parameters are known.

### 3.6.2 Standard ORB-SLAM Implementation

To estimate the camera pose, and thus the position of the survey vehicle, we use the ORB-SLAM system [178] which achieves state-of-the-art performance on monocular vSLAM problems. Furthermore, the system performs in real-time and is reportedly suitable for large environments such as the highway [178]. ORB-SLAM consists of three parallel threads: tracking, local mapping, and loop-closing, which are now described.

The tracking thread extracts ORB feature matches to perform BA and estimates the camera pose for each new image. In addition, the tracking thread decides if a new image is a keyframe. The local mapping thread uses the keyframes to optimise both the camera pose and the map. However, a *local* BA is performed in which only positions of those features close to the current camera pose are optimised. The loop-closing thread decides if a loop has occured, that is, the

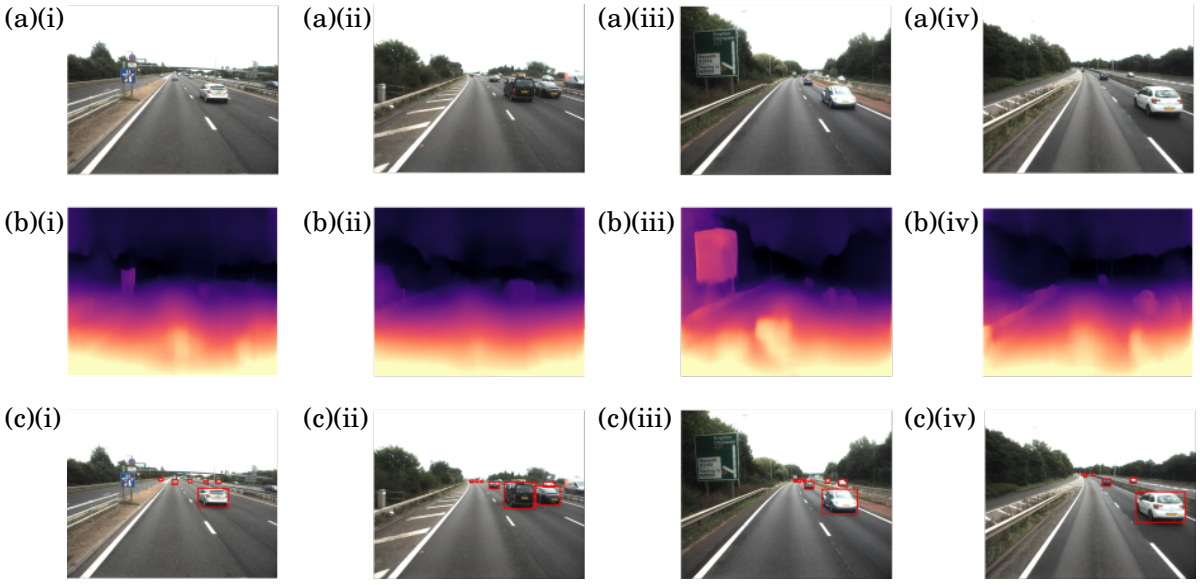



FIGURE 3.20. Example outputs from the CNNs with which we have augmented ORB-SLAM. (a) Survey imagery and corresponding depth estimation (b) and vehicle detection (c).

current camera pose is similar to a previous pose. ORB-SLAM uses this constraint to perform 'loop closing' in which the loop information is exploited by the BA to optimise all camera poses, and the entire map, and thus correct drift.

However, using such a vSLAM system with the survey imagery presents two problems.

1. Low parallax. The BA optimisation performs poorly if there is low parallax where the pixel coordinates of feature matches are similar across multiple images. Survey images captured on straight sections of the highway are likely to suffer from low parallax, see Figure 3.19.

2. Dynamic environment. Visual SLAM systems assume that the camera moves within a static environment so that feature correspondences are the same three-dimensional coordinate in each image. Clearly, features found on moving vehicles on the highway will break this assumption.

Consequently, we find that ORB-SLAM performs poorly on the A27 survey imagery. Figure 3.19 shows two examples in which the estimated camera pose and map is inaccurate.

### 3.6.3 Modified ORB-SLAM Implementation

To overcome these two challenges, and thus provide an improved camera pose estimate, we modify ORB-SLAM with two state-of-the-art CNN tools. The two CNNs and our modifications to ORB-SLAM are now described.

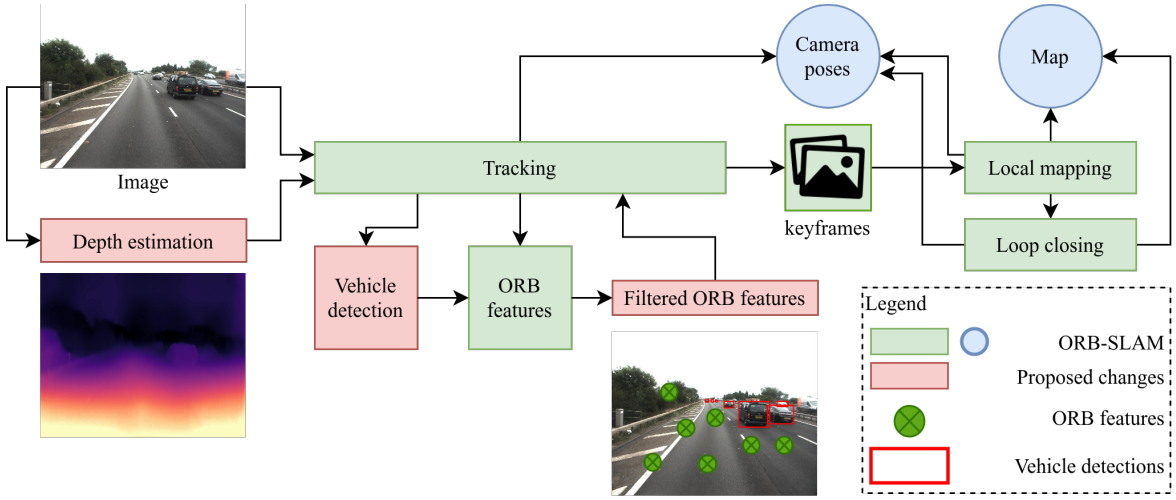



FIGURE 3.21. A diagram of our vSLAM system. The original ORB-SLAM system is shown in green and our modifications are shown in red. The camera poses and map coordinates computed by the system are shown in blue. Two CNNs are built into the system; a depth estimator to overcome low parallax on the highway, and a vehicle detector to remove ORB features found on vehicles, in order to create a static environment.

### Depth Estimation Tool

Low parallax can be overcome if the depth (the distance along the optical axis of the camera) of the each feature is known. Typically this distance is estimated with an additional sensor such as LIDAR or a stereo camera. In our case, no such secondary sensing system is available, so instead we use a novel deep learning architecture to provide depth estimation. Unlike standard CNNs (described in detail in Section 3.3.5), the output is an upsampled convolutional layer and a relative depth is estimated across the the whole image. There are number of such architectures ([76, 158, 169], for example), however, on the KITTI data set (which includes highway imagery) state-of-the-art performance is achieved by [28] and we therefore use this CNN for depth estimation.

### Vehicle Detection Tool

We create a static environment by removing ORB features found on dynamic objects in the survey imagery. All dynamic objects on the highway are likely to be other vehicles, and therefore we need only detect the vehicles in the survey imagery and remove those ORB features found on them.

Object detection in images was first achieved by sliding window-based methods, that classify features extracted from the window [182]. Later, with the advent of deep learning, large CNNs improved classification performance. However, detection and classification were historically performed separately, that is, regions in the image likely to contain an object were identified and then subsequently classified, resulting in slow performance [260].

State-of-the-art performance is now achieved by the YOLOv3 (you only look once) architecture [214]. The detector consists of a single conventional CNN that encodes a bounding box and classification label for each object in the output layer. YOLOv3 performs detection in real time and is available pre-trained on the COCO data set [157] that contains trucks, cars, and motorbikes. Therefore we may easily detect vehicles in the survey images without re-training the network. The YOLOv3 detector has been used to remove dynamic features in other vSLAM systems [139, 261], however, to the best of my knowledge there are no other vSLAM implementations that use the architecture to remove dynamic features on the highway. Examples of the depth estimation and vehicle detection are shown in Figure 3.20.

To perform vSLAM on the survey imagery, the two CNNs are built into ORB-SLAM. The depth estimation is performed as the image is loaded. The estimated depth for each extracted ORB feature is then subsequently used by all three threads to overcome the low parallax when optimising the camera poses and map. The vehicle detection is performed within the tracking thread: ORB features found within the vehicles' bounding boxes are filtered and removed. The system continues to run in real time, and the original ORB-SLAM inputs and commands remain unchanged. A system diagram of the modified ORB-SLAM is shown in Figure 3.21.

### 3.6.4 Survey Vehicle Localisation

We now employ the modified ORB-SLAM system to compute the camera pose and thus localise the survey vehicle along the A27 highway directly from the survey imagery. The values of intrinsic camera parameters (the focal lengths $f_u$ and $f_v$) are those given in Table 3.1 that are computed via the calibration process. To consider the effectiveness of the system, the estimated camera poses and map are inspected visually. We found that unlike the examples shown in Figure 3.19, the modified system produces smooth trajectories that appear to follow the expected route along the highway. In addition, we compared the camera pose estimated by our system against the position and heading provided by the IMU. Figure 3.22 shows two visualisations of the estimated camera pose and map from our modified ORB-SLAM system, and a comparison of the estimated camera pose against the corresponding IMU readings for two segments of the A27 highway.

Figure 3.22 shows that, at a qualitative level, our modified vSLAM system provides a good estimate for the camera poses, to a relative scale. Unfortunately, when implemented over a long segment of the highway, the system suffers from drift, which is an unavoidable side-effect in vSLAM systems with no dead-reckoning correction.

The vSLAM system represents the beginnings of a promising survey vehicle localisation capability, but more work needs to be done to further develop the system into an operational capability. Firstly, the system should be refined to estimate the scale of the camera poses and map. To achieve this, collecting measurements of known features in the images (the width of the highway for example), or an accurate depth estimation provided by a secondary sensor (a calibrated stereo camera or LIDAR) installed on the survey vehicle is likely to be the way forward.

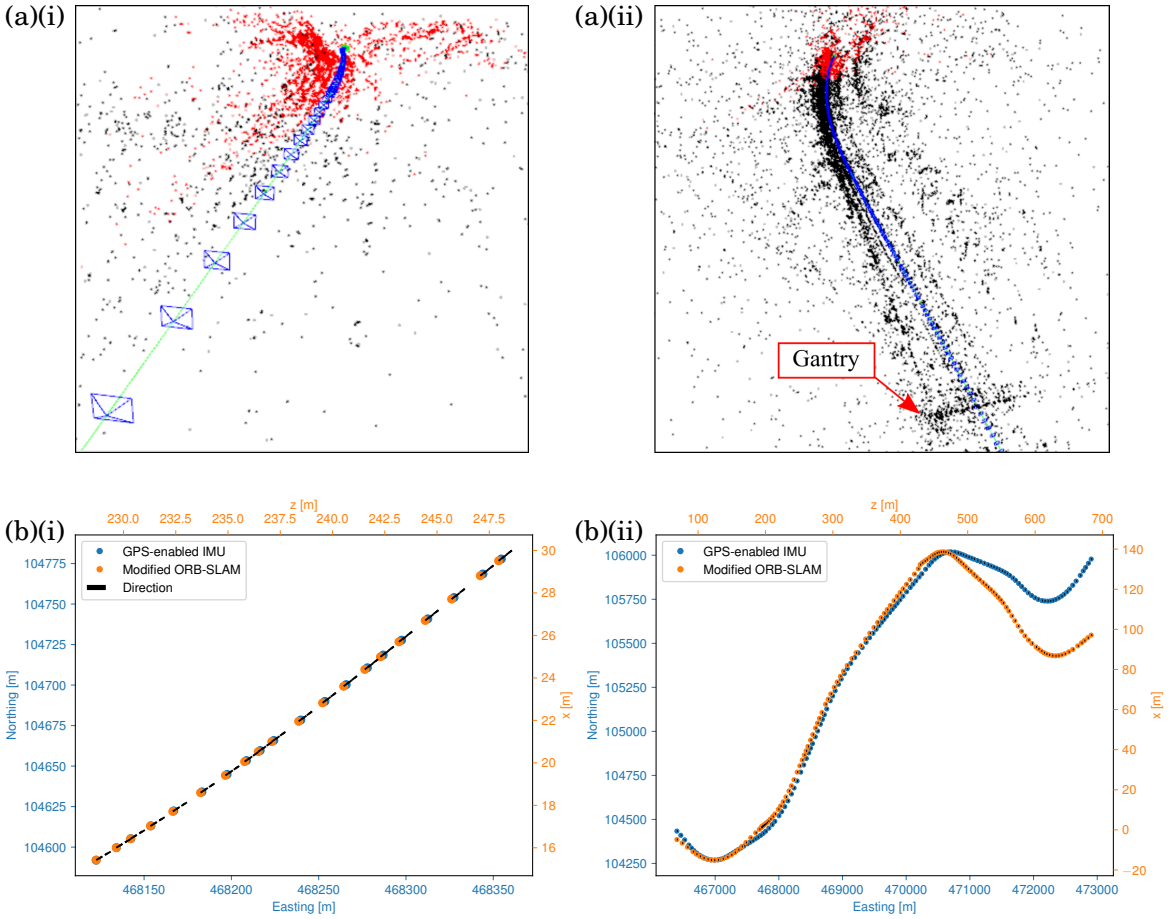



FIGURE 3.22. The modified ORB-SLAM system in operation. (a) camera poses and map estimated by our vSLAM system for two sequences of the A27 survey imagery. Both sequences show a smooth trajectory and in panel (a)(ii) highway assets such as a gantry can be identified in the map. The two panels are coloured as described in Figure 3.19. (b) The camera pose and corresponding position and heading recorded by the IMU. An 800 m and six km segment of the A27 are shown respectively in panels (b)(i) and b(ii). We see that our system compares well to the IMU, although the poses are estimated to an arbitrary scale. Over the six km segment, we see that system beings to suffer from drift, possibly due to over-rotating around a bend.

Secondly, a future system should consider how to correct the drift. One way of achieving this is through loop closing, however, this is would not be possible for a survey vehicle travelling in one direction along the highway. Alternatively, a dead-reckoning system, in which features with known positions are identified in the images might be employed instead. In low-feature, closed environments such as a tunnel, dead-reckoning might be achieved via a sensor (e.g., radar) installed (at a known position) on the roadside that is able to communicate both the position of

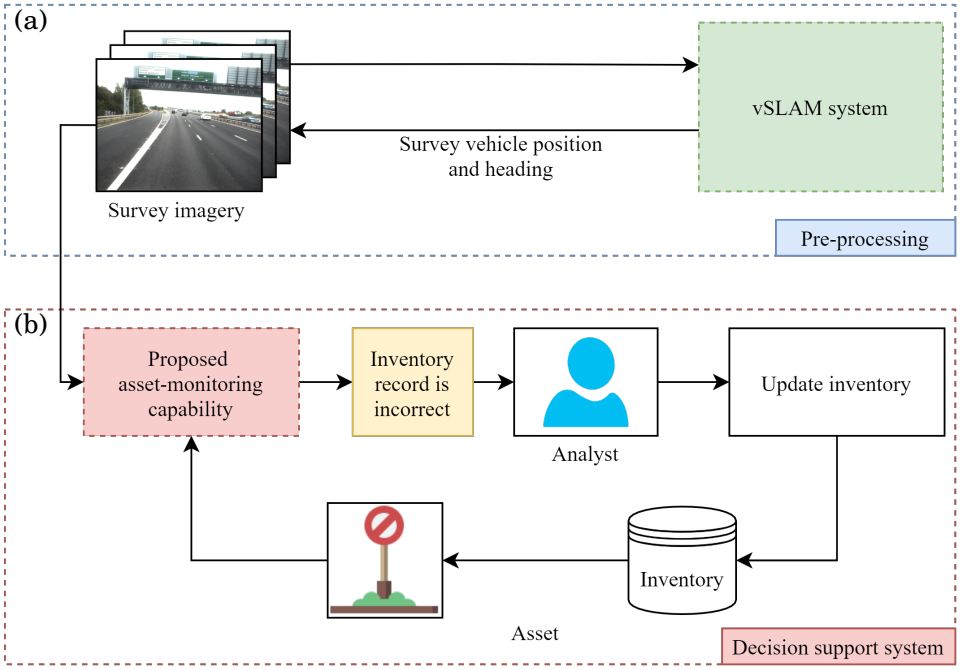



FIGURE 3.23. Envisioned future asset monitoring capability. (a) The vSLAM system is used as a pre-processing tool to compute the vehicle position and heading as the imagery is taken. The TAM decision support system (b) may then use the processed survey imagery to verify the inventory of assets installed in both open and closed environments, such as tunnels.

the sensor and the relative position of the survey vehicle (i.e., the vehicle's position with respect to the sensor). Thus, a small device installed on the survey vehicle may compute its global position, and correct the accumulated drift.

### 3.6.5 Envisioned Asset Monitoring System

We now consider how the vSLAM localisation capability may be embedded within the TAM decision support system previously developed and tested in this chapter. The inventory is updated by our system post-survey, and assuming that the camera is calibrated, the system requires the survey imagery with the corresponding survey vehicle position and heading as the image is taken, as well as an inventory of assets (to be verified). As there is no time-critical constraint on the inventory updating process (an asset is likely to remain incorrectly recorded in the inventory from one day to the next) it is logical to use the vSLAM system as a kind-of pre-processing tool, in which the accompanying vehicle position and heading for each survey image are refined or computed for the closed environments where a satellite signal is not available, thus making our decision support system more robust. The envisioned future capability is sketched out in Figure 3.23.

## 3.7 Method Justification

The work presented in this chapter enhances the important (and widely practiced — see Section 3.1) TAM activity of inventory updating and thus the rationale for our research is sound; essentially, the analyst may now perform their task many times faster, providing cost savings and potentially reducing human error.

Our method exploits a camera calibration process to localise each asset in the survey imagery; the alternative approach would be to use a trained object detection system (e.g., the YOLOv3 network) to detect the assets instead. While this approach may detect assets not recorded in the inventory (i.e., directly from the imagery), this alternative would require an extensive labelling exercise (where imagery is annotated with asset bounding boxes), and may introduce new errors if the network is poorly trained. Furthermore, the inventory is a rich and important data source, used by agency analysts across the globe, that describes each asset in a physical context (i.e., its position and size). Therefore, our system, that exploits the inventory, (a) makes use of this existing data source, and (b) compliments and easily integrates alongside the current role of the analyst.

We used the Inception v3 CNN architecture to classify assets localised in the survey imagery. Two other CNNs were considered, each employing different novel sub-architectures (e.g., residual blocks and skip connections). Overall the Inception v3 CNN performed best on the withheld validation data set, and was thus used in our system. We implemented a transfer-learning framework whereby a pre-trained model was retrained with a relatively small set of assets. The alternative would be to train a CNN from scratch, however, this would require a larger labelled training set and more computing resources, with no guarantee of improved performance — the 98% validation set prediction accuracy achieved via transfer learning would be hard to beat!

To assess the effectiveness of our system the assets were split into a training set and a test set; the first group was used to perform the camera calibration and CNN training, and the second group was withheld to test the system, thus ensuring the validity of our results (i.e., no model bias).

In this work, we aimed to prove the feasibility of our system, and thus three different asset types (traffic signs, matrix signs, and reference markers — i.e., common roadside assets) were only considered. However, our system may easily handle a larger number of asset types and is thus suitable for different roadside inventories — an expanded training set (with extra labels) and modified output layer (with more classification nodes) are only required.

The work presented in Section 3.6 explored how a computer vision-based localisation system might improve the robustness of our method. We considered a system that relies only on the data collected from a single camera (i.e., typical survey vehicle sensor setups), and is thus a sensible approach. We embedded two CNN tools, each one state-of-the-art in their respective applications (object detection and depth estimation). In this initial investigation, we assessed the localisation performance against ground-truth data collected from the GPS-enabled IMU on the

survey vehicle via visual inspection. While our qualitative assessment is suitable for our initial work, future research must test the system at a quantitative level.

## 3.8 Conclusion

In this chapter, a decision support system to assist those responsible for the maintenance of highway asset inventories is developed and tested. Our system applies computer vision-based methods to integrate previously disparate data sources used by an agency analyst; namely, survey data captured by a camera and GPS-enabled inertial measurement unit (IMU) installed on a survey vehicle, aerial imagery of the highway, and the asset inventory.

Our new roadside asset monitoring capability consists of two distinct computer vision contributions. Firstly, the camera on the survey was calibrated by collecting control points. As such, the position of each asset as it is recorded in the inventory, may be projected into the survey imagery. Secondly, transfer learning was used to retrain a state-of-the-art convolutional neural network (CNN) to classify a localised asset in the survey imagery, thus automatically verifying the asset's inventory record. The transfer learning process was applied to three different base CNNs, however the ResNet50 network achieved the highest classification accuracy of 98%, with a relatively small architecture (25 million weights), on a withheld validation set of assets.

To prove the feasibility of our system, three different asset types were considered: traffic signs, reference markers, and matrix signs. The effectiveness of our system was determined by considering those assets in a withheld test set that are verified and those that are flagged for further manual inspection. Overall, 91% of the assets are automatically verified, which would otherwise generate manual work for the analyst. As such, we have demonstrated the feasibility of our system and its benefit to an analyst.

A number of potential refinements to our system as presented in this chapter are proposed, as well as opportunities for future research to further mature and improve the capability. In particular, we explored the beginnings of a promising tool to localise the survey vehicle directly from survey imagery collected by the camera. Specifically, we considered how a visual simultaneous localisation and mapping-based (vSLAM) methodology might be embedded into a future capability to refine, or replace missing IMU position data required by our system. Such a tool would improve the robustness of our decision support system and enable the system to automatically verify the inventory of assets installed in closed environments where GPS may be unavailable.

The key novelty of our system lies in the perspective of the analyst. Unlike previous computer vision-based TAM systems, our decision support tool is designed to compliment the current asset monitoring process. This is achieved in two ways: firstly, the system integrates multiple data sets used by the analyst. Secondly, the system is built upon the inventory; just like the analyst, each asset is considered as it is recorded in the inventory and subsequently verified or flagged for inspection.

CHAPTER 4

# ROLE OF TRAFFIC OFFICERS FOR TRANSPORTATION ASSET MONITORING

*Sections 4.2–4.5 are based on works summarised in the paper 'Role of Traffic Officers in Transportation Asset Monitoring' which has been accepted for publication in the proceedings of the 2022 Transportation Research Board (TRB) conference and selected for review in the Transportation Research Record (TRR, the conference's associated journal) [240].*

**Problem Statement**

Agency analysts rely on remotely captured survey data, usually collected on an annual basis, to monitor highway assets. This frequency of data collection is not suitable for those assets such as vegetation, street lights, or guardrails whose condition might rapidly change. Therefore, we consider via simulation, and as a proof-of-concept, whether traffic officer vehicles might also collect real-time, network-wide asset data (via dashboard cameras) alongside their primary function of incident response.

**Chapter Glossary**

| Term | Definition |
|---|---|
| AIRT | Average incident response time |
| TAIRT | Time-averaged AIRT |
| AIVT | Average edge inter-visit time |
| TAIVT | Time-averaged TAIVT |
| APV | Average percentage of edges visited by at least one traffic officer |
| APVW | APV at the end of the time window |
| ALR | All lane running |
| BWNN | Bell and Wong nearest neighbour |
| Dash cam | Dashboard camera |
| DfT | UK Department for Transportation |
| ERA | Emergancy refuge area |
| FTO | Future traffic officer |
| GA | Genetic algorithm |
| HA | Highways Agency |
| HATRIS | HA traffic information service |
| HE | Highways England (UK highway agency, formally HA) |
| ILD | Inductance loop detector |
| KPI | Key performance indicator |
| NTIS | National traffic information service |
| OR | Operational research |
| MCLP | Maximal covering localation problem |
| PPRP | Police patrol routing problem |
| PSP | Persistent surveillance problem |
| PRT | Personal rapid transport |
| RIS | Road improvement strategy |
| RP | Road period |
| RWS | Rijkswaterstaat (Dutch highway agency) |
| SRN | Strategic road network |
| SVD | Stopped vehicle detection |
| TO | Traffic officer |
| TOFV | TO fleet vehicle (data set) |
| UAS | Unmanned aerial system |

| Term | Definition |
|---|---|
| VRP | Vehicle routing problem |
| VRPTW | VRP with time windows |
| VRPPD | VRP with pickup and delivery |
| DVRP | Dynamic VRP |
| OVRP | Open VRP |
| VANET | Vehicular ad-hoc network |
| VDS | Vehicle dispatching system |
| VMS | Variable message sign |
| WMIRT | Weighted minimum incident response time |

Traffic incidents are a major contributor to congestion on highways. Vehicle breakdowns, collisions, and debris can cause highway lanes to close while an incident is cleared and thus result in major delays; Schrank and Lomax [227] found that between 52 to 58 percent of all delays can be attributed to traffic incidents. In order to keep traffic moving freely, incidents are often attended by traffic officers (TOs) on patrol who usually arrive first at the scene.

Traffic officers are usually part of a police sub-unit, who are expected to manage incidents alongside law enforcement duties. For example, as a result of heavy traffic and high-speed collisions on the autobahn highway in Germany (where some sections have no speed restrictions), the German police force created the Autobahnpolizei; a police unit assigned to manage traffic flow and respond to incidents along the autobahn alongside duties in reducing crime, such as speeding and drink-driving [204]. The Highway Patrol units in the U.S. (known as state troopers or state police in some states) [10], Garda National Roads Policing Bureau in Ireland [11], and Polizia Stradale in Italy [205] are three other exemplar police sub-units deployed on highways across the world.

Alternatively, TO fleets deployed in some countries form a dedicated incident management capability, separate from the police and under the direct control of a highway agency. One such fleet are the Weginspecteurs (road stewards) introduced in 2004 by Rijkswaterstaat (RWS), the national highway agency in the Netherlands [220]. The RWS weginspecteurs are trained by the Dutch police in areas of incident management and traffic control, thus enabling the police force to focus on law enforcement. Also in 2004, Highways England (HE), the largest agency in the UK, created the HE TO fleet deployed specifically to attend and manage incidents on the UK highways. The fleet has proven to be an effective presence since its introduction; reportedly contributing to a 12% reduction in incident-related congestion, and freeing up 44% of police time on UK highways [206]. These dedicated TOs are less common than their police sub-unit counterparts; only two (HE and RWS) out of 16 highway agencies across Europe interviewed by Steenbruggen et al. [237] employed a dedicated fleet of TOs.

Traffic officer vehicles are equipped with a range of communication systems and equipment to safely and quickly clear incidents on the highway. The HE TO vehicles are fitted with a hands-free mobile phone, radio, and telematics [74, 121] that allow the fleet to communicate with one another and with control centres, as well as providing live position data. The radio communications systems used by the HE TOs were upgraded in 2020 to make the devices robust to weather (namely, wet and dusty conditions) and noise (mostly caused by other vehicles on the highway) [150]. Each HE TO carries portable warning lights, cones, and lane diversion arrows to temporarily redirect traffic around an incident, and tow bars, straps, and slings to move a vehicle off the highway [112]. The Weginspecteur vehicles are fitted with a notable additional (compared against HE TO vehicles) piece of equipment; a large vehicle-mounted variable message sign (VMS), see Figure 4.1(f). The VMS allows the Weginspecteurs to give direct instruction to drivers on the Dutch highways, without relying on a roadside or gantry-fitted VMS. In addition, the HE

TO and Weginspecteur vehicles are equipped with a dashboard camera (dash cam) and capture imagery data of the highway while on patrol [118], see Figure 4.1(c).

Asset management, discussed in detail in Section 3.1, is another key area of operation for a highway agency [113, 252] that ensures transportation assets (the pavement surface, traffic signs and street lights, for example) are properly monitored and maintained. To capture asset data, highway agencies usually deploy specialised vehicles, equipped with sensors such as a camera or LIDAR to perform periodic (often annual) surveys of the highway and its assets (see Section 3.1). From this remotely captured survey data, agency analysts may determine the assets' condition and suitable maintenance schedules can be designed. These periodic surveys are sufficient for those assets whose condition does not degrade between surveys (the pavement surface for example), but are clearly not suitable for assets such as vegetation, drainage systems, street lights, and guardrails whose condition might rapidly change.

The apparent opportunity is to use those TOs with a dedicated incident management capability (such as those deployed by HE and RWS), to also collect asset data. Ultimately, these TOs already operate across highway networks (with a dash cam), and so they represent a potential frequent asset data collection capability.

To consider the feasibility of our proposed system, a proof-of-concept study is undertaken. A simulator is developed to deploy TOs on the highway who patrol and respond to dynamically generated incidents. Two patrol regimes are considered, one that tries to minimise the fleet's incident response time, and one that tries to maximise the fleet's coverage (to capture asset data across the entire highway network). By considering the second regime's incident response time against the first, we will determine the feasibility of our proposed system.

The remainder of this chapter is organised as follows. In Section 4.1 we review some of the previous related vehicle routing work, and the current operational role and environment of dedicated TO fleets are discussed. In Section 4.2 the data sources with which we build our study are described and pre-processed. The concepts of operation (i.e., the TO patrol regimes and incident response mechanism) of the proposed system are described in Section 4.3. The simulation in which we deploy our proposed system (under different patrol regimes) is described in Section 4.4, and the results are provided in Section 4.5. Finally, the results, future improvements, and considerations for a deployed system are discussed in Section 4.6 before the chapter is concluded in Section 4.7.

## 4.1 Background

Highways England was established in 1994, although it was initially known as the Highways Agency (HA) — an executive agency attached to the UK Department for Transport (DfT). In 2010, the DfT commissioned an independent review of HA's approach to managing the strategic road network (SRN, the highway network in England). The review concluded that the HA was


FIGURE 4.1. Highways England (HE) traffic officers (TOs) (a) and Weginspecteurs (f) respectively patrol along the strategic road network in England, UK (e) and the Dutch highway network. Each TO vehicle is equipped with a dashboard camera, shown in panel (c). Traffic officers are assigned to attend and manage traffic incidents (b) by regional control centre operators (d). Panels reproduced from: (a), (b), and (d), Auto Express website [19]; (c), online video frame [30]; (e), HE [105]; and (f), Ebo Van Weel website [64].

(organisationally) too close to government, and thus, there was little pressure on the government to make, or fulfill long-term decisions on SRN funding. Furthermore, the electoral cycle caused 'stop-start' investment (as government manifestos and priorities changed) which resulted in

inefficient development and improvement of the SRN [59]. Consequently, in 2015 the HA was restructured and rebranded as HE — a government owned company with an economic structure more aligned to the private sector[1].

In 2004, HE (then HA) introduced the HE TO fleet that patrols on the SRN, depicted in Figure 4.1(e). While on patrol, HE TOs either drive along designated patrol routes (typically the busier sections of highway) looking for incidents or remain stationary at waiting areas. TOs are instructed by operators in regional control centres (RCCs) to respond to incidents via the radio communications system installed in the vehicle. Although the TO fleet is not part of the UK emergency services, HE TOs have a number of extra powers derived under the Traffic Management Act of 2004 [56] that enable them to quickly and safely clear incidents. For example, a TO on the SRN may stop a vehicle, close highway lanes, and place temporary traffic signs. Furthermore, TOs are trained in first aid and CPR and can provide medical assistance at the roadside [184]. A TO is often first to arrive at an incident and usually takes lead command, unless there is a loss of life where they fall under the command of the emergency services. In total, the TO fleet consists of 234 vehicles that begin and end their patrol shifts from 32 regional outstations [116].

Similarly, the Weginspecteur fleet patrols on 5,200 km of highway in the Netherlands. Just like the HE TOs, the core responsibility of the fleet is to attend and safely clear incidents on the highway (via lane closures or vehicle removal, for example) [181]. In addition, the Weginspecteurs look out for potentially dangerous situations, such as oil spillages, and report them back to control centres. Although the Weginspecteur fleet is completely separate from the police, some TOs in the Netherlands are enrolled as 'Extraordinary Investigation Officers' who may compile official reports against drivers for a number of driving offences, including ignoring red crosses on VMSs (indicating a lane closure) or driving on the hard shoulder, for example.

Returning to the UK, the SRN is a key transportation network made up of 2,984 km multi-lane highway and 4,114 km of trunk A-road highway in England [57], see Figure 4.1(e). The SRN is a vital asset valued by the UK government at £128 billion [115] that connects people and businesses across the country. The network carries a third of all vehicle miles travelled in England; reportedly 90% of drivers in England rely on using the SRN at least once a month and 96% of the English population live within a one-hour drive from an SRN junction [103]. Further, one billion tonnes of freight in England (two thirds of all freight traffic) is transported along the SRN, and thus it is a key contributor to the economy. Highways England report that 75% of businesses in England consider a free-flowing SRN critical to their operation [105], and identify a number of SRN reliant sectors that contribute £314 billion to the economy [105], such as logistics, retail, and construction [106].

The number of vehicle miles travelled on the SRN has increased by 14.8% from 2009 to 2019 and a further 21% increase is forecast over the next 20 years. Consequently, in 2014 HE and

[1]In August 2021, HE was again rebranded as National Highways, see page 8.

the UK government announced the road investment strategy (RIS) to improve the SRN. The RIS outlines the long-term vision for the SRN to be realised by 2050, and focuses on improving driver safety and user experience. The RIS funds a set of improvement and maintenance schemes across the SRN that are delivered in five-year 'road periods' (RPs); the first (RP1) began in 2015 and the second (RP2) in 2020. The SRN improvement schemes are designed and tested against key performance indicators (KPIs) to be met by the end of each RP. The UK government has invested £15.2 billion across RP1 that has funded over 100 different schemes, including building new lanes, improving cycling sections, and reducing noise pollution [119]. By the end of RP1, HE had missed only two of 10 target KPIs — a target 90% user satisfaction score was missed by 0.8%, and a KPI on the number of killed or seriously injured drivers that is waiting on data to be published. Currently, HE are in the second year of RP2 with a further £27.4 billion invested by UK government to meet 12 target KPIs by 2025 [114] that include incident time-to-clear [96] and average delay; clearly TOs will play a critical role if HE are to meet their targets.

The SRN is split into seven regions each managed by a regional sub-division of HE. Each region is monitored from an RCC by operators to ensure that the highway remains free flowing and safe. Usually, operators identify incidents from video feeds captured by CCTV cameras installed across the SRN but may also be alerted to incidents by drivers either by mobile phone or telephones on the roadside. The operators coordinate and deploy a response plan (dispatching TOs or liaising with emergency services, for example) with the aim to quickly clear each incident.

In addition, HE provide SRN users with information on real-time traffic conditions through the National Traffic Information Service (NTIS). The NTIS connects to various traffic data feeds across RCCs including CCTV and inductance loop detectors (ILDs, see Section 2.1.1), as well as information on planned incidents such as roadworks or sporting events to determine live traffic conditions [126]. The NTIS has access to VMSs across the SRN to deliver up-to-date advice to drivers, as well as providing an online map-based service [127] and social media feeds.

To increase the capacity of the SRN and thus reduce congestion, HE have introduced 'smart motorways' along particularly busy sections of highway that employ variable speed limits, a dynamic hard shoulder (that can be temporarily used as an extra lane), or have no hard shoulder at all: the latter is known as 'all lane running' (ALR) highway.

Despite ALR highway reducing the number of fatal causalities (0.12 per hundred million vehicle miles travelled) compared with the conventional SRN highway (0.16 per hundred million vehicle miles travelled) [128], smart motorways have proven to be a controversial scheme. The removal of the hard shoulder forces drivers involved in an incident (e.g., a collision or a vehicle breakdown) to wait in live highway lanes while TOs or the emergency services arrive. Fatalities on ALR highway that might have otherwise been avoided with a hard shoulder have attracted a lot of media attention [26, 185, 225]. In response, HE are rolling out stopped vehicle detection (SVD) technology — a radar-based system that alerts an RCC to a stationary vehicle in a live lane — across all ALR highway by 2022 and plan to install additional emergency refuge areas (ERAs,

where a vehicle may safely remain stationary on the roadside) such that an ERA is available every three-quarters of a mile along the SRN [128].

### 4.1.1 Vehicle Routing Problems

The problem of designing optimal routes for vehicles is a widely studied operational research (OR) question known as the vehicle routing problem (VRP). The VRP was first considered by Dantzig and Ramser [49] who developed a method to route a fleet of fixed-capacity homogeneous petrol trucks from a central depot to a series of delivery sites, before returning back to the depot. The method finds a set of shortest paths (one for each truck) along a transportation network defined as a graph of edges (road links) between demand nodes (places where petrol must be delivered). The VRP formulation proposed by Dantzig and Ramser constrains the trucks to visit each node only once, and can therefore be viewed as a specialisation of the travelling salesman problem (TSP) [75] with the additional constraint that each route must begin and end at the depot.

Today, there a number of variations of the VRP employed to solve specific transport logistic problems. One popular extension is the VRP with time windows (VRPTW) [65] in which nodes must be serviced within a given time window — store deliveries that must arrive within its opening hours, for example. Another variation is the VRP with pick up and delivery (VRPPD) where deliveries are collected on route (rather than at the depot), and in the open VRP (OVRP) the delivery vehicles do not return to the depot. An example application of the OVRP is a delivery company that does not own enough (or any) delivery vehicles, and thus must rent a number of extra vehicles for the day.

Solving the VRP and its variations is an NP-hard problem, and optimal solutions can only be found for relatively simple applications (around 100 demand nodes) [151]. Instead, for real-world applications, that are likely to be much more complex, the VRP is often solved via a set of heuristics. Typically, these heuristics are performed in two stages; construction and improvement. The first stage computes an initial first sub-optimal solution that is then refined in the second stage. One simple construction method is the Clark and Wright [45] heuristic that begins with $n$ 'back and forth' routes for $n$ demand nodes; that is, a vehicle leaves the depot then visits a single demand node and returns back to the depot. The set of back and forth routes are then merged into longer routes that visit multiple nodes. Typically, improvement heuristics are built upon relatively simple inter-route operators. For example, the RELOCATE operator removes consecutive nodes from one route and places them into another, and the 2-OPT* operator removes two edges from two routes and reconnects them differently [151].

In practice, optimal routes are often computed via metaheuristics, that is, a high-level and problem-independent framework. Metaheuristics can be broadly categorised into two groups; local search and population-based methods. The first group begin at an initial solution and iteratively explore the local neighbourhood of the solution space while employing strategies to avoid local minima and solution cycling. Commonly used LS methods include simulated

annealing that incorporates randomness into the solution space exploration [144], and tabu search that restricts the search so that similar solutions are not considered [82]. The second group of methods are inspired by natural processes such as animal foraging and evolution of species. These methods employ high-level strategies that use the concept of memory (i.e., the success of previous iterations) when exploring the solution space. Genetic algorithms (GA) are a population-based method that have previously been used to solve the VRP [208, 209]. Genetic algorithms iteratively mutate a large set of solutions (the population) and score each member of the population with a fitness function. Fitter solutions survive and repopulate, and thus the population converges to an optimal solution.

Standard VRPs assume that the demand is static and known in its entirety before the vehicles begin their routes. However, in most real-world scenarios, the demand is dynamic (hail-and-ride taxi trip requests, for example) and vehicle routes may need adjusting while in operation (i.e., not at the depot); this extension to the VRP is known as the dynamic VRP (DVRP).

In a recent review provided by Pillac et al. [201], the DVRP is categorised into two classes; deterministic and stochastic. In this first class of problems, the demand is revealed over time. Thus, an optimal solution may only be obtained for the current state, which then becomes sub-optimal as new demand is realised. Typically, deterministic demand problems are solved by invoking a standard VRP (with the demand considered static at that time point) and correspondingly updating the vehicle routes.

The second class of problems consider demand with known statistics (computed from historical data, for example), and thus future demand may be inferred. To solve the stochastic demand problem, an ensemble of future demand scenarios are sampled from the known demand distributions, and are then solved as a set of static VRPs. Optimal routes are then found by computing the associated statistics across the entire ensemble.

When assigning a TO to respond to an incident, a control centre operator considers the position of the entire fleet and makes a strategic decision to assign TOs that minimise the response time. Such centralised vehicle dispatching systems (VDS) are a specialisation of the DVRP (with the constraint that demand should be serviced as soon as possible) and are found in a number of demand-driven transportation systems such as taxis [264] and emergency repair vehicles [262].

Both a deterministic and stochastic VDS for a personal rapid transit (PRT) system are considered by Lees-Miller and Wilson [152]. The deterministic method assigns the nearest available vehicle as trips are requested in sequence, whereas the stochastic method moves idle (or empty) vehicles to stations to service future demand. On two test networks the stochastic method achieved up to a 96% reduction in passenger waiting time compared against its deterministic counterpart.

The DVRP becomes complex for large transportation systems, such as the highway networks considered in this chapter, and rerouting a vehicle multiple times throughout one journey

may be unpractical and an impossible burden on the driver. Furthermore, standard DVRP-based approaches may not be appropriate for vehicles with complicated operational constraints (beyond time, fuel, or mileage limits, for example) such as a TO or police officer. Instead, such complicated transportation problems are often solved through simulation that may incorporate rich application-specific objectives.

The police patrol routing problem (PPRP), concerned with designing patrol routes to minimise the response time to crime while maximally covering a patrol area to deter criminals, is often studied via simulation. The PPRP has a strong analogy with our TO problem: both aim to maximally cover an area (to deter criminals or capture asset data) while minimising response time (to crime or traffic incidents).

One simulation-based approach to the PPRP, is the GAPatrol system developed by Reis et al. [217]. The authors developed a simulation, based in the Fortaleza area of Brazil, in which a group of criminals randomly try to commit crimes at predetermined targets while avoiding a fleet of police officers. Their GA-based method determines a set of police patrol routes that minimise the number of crimes in the simulation, and identifies a set of locations for each police officer to visit in sequence. A similar simulation is presented by Melo et al. [173], however, in their work they consider various 'physical reorganising' strategies; namely, how the police officers drive between crime hot-spots. Through their simulation the authors found that a short routing strategy, where police officers drive along the shortest route between hot-spots, minimised the amount of crime committed.

A recent in-depth review of the PPRP problem and various proposed solutions is provided by Dewinter et al. [62]. The authors identify that current solutions to the PPRP do not consider an important aspect of police patrol — the presence of police vehicles in fact deters and proactively reduces crime. Thus, the PPRP should also try to maximise patrol length as well as minimise crime response time. Future research into the PPRP problem should address this shortcoming to more accurately reflect true police patrols.

Ambulances and fire trucks differ from TO vehicles in that they are usually stationed at and dispatched from a facility (a hospital or fire station) and then return (either with a patient or once a fire has been put out). The OR questions concerned with such stationary vehicles do not try to design optimal routes, but rather, they consider where a number of facilities should be placed to best service a population in a given geographic region. For example, ambulances in the UK are stationed to meet a target average response time of 8 minutes to category one calls (life threatening conditions, such as cardiac arrest) across the entire country [186].

The OR problem of facility placement was first proposed as the maximum covering location problem (MCLP) considered by Church and ReVelle on a graph of demand nodes and edges [42]. A methodology was developed to identify those nodes where $p$ facilities should be placed by maximising the percentage of the 'covered' population (nodes) within a service radius, $s$, around each facility. Two test networks were considered and curves were computed to show the effect

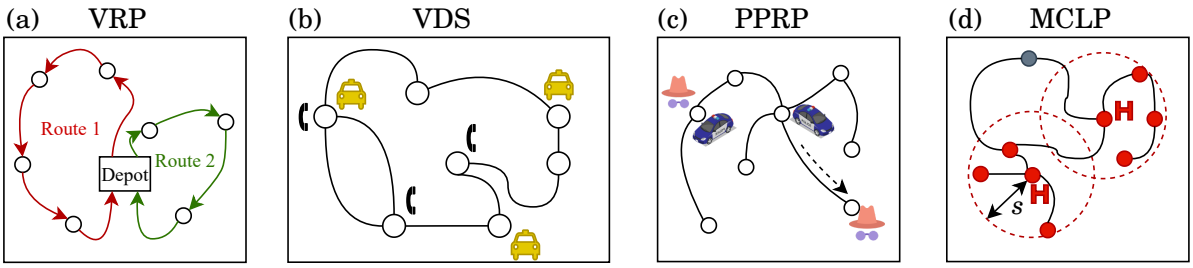



FIGURE 4.2. Existing vehicle routing and dispatching problems. (a) The classical vehicle routing problem (VRP) determines a set of delivery routes to demand nodes. (b) Vehicle dispatching systems (VDS) assign vehicles to meet dynamic demand as it is requested. (c) The police patrol routing problem (PPRP) in which police units patrol an area and respond to crime is both a VRP and a VDS. (d) The maximal covering location problem (MCLP) identifies where vehicles should be stationed (such as ambulances at hospitals (H)) to maximise the number of nodes within a serviceable radius $s$ of a station.

of changing both $p$ and $s$, thus decision makers can choose whether to alter either (or both) the number of the facilities or the distance each vehicle travels from its facility. Subsequent methods have extended the MCLP to capture more complex applications, such as the capacitated MCLP that constrains the workload on each vehicle [203], and applying the MLCP on a graph whose nodes have uncertain demand [24].

Currently, TO outstations are placed to cover only the multi-lane highway of the SRN (i.e., not the trunk A-road highway) (HE, personal communication). However, HE are currently considering where to place a number of new outstations, as well as refuge areas where TOs may remain stationary along the SRN, to improve coverage across the entire SRN. It thus seems that an MCLP-based solution is likely to be the way forward. Existing vehicle routing and dispatching problems are depicted in Figure 4.2.

### 4.1.2 Mobile Sensing and Maximal Coverage Systems

The use of mobile sensing capabilities for transportation asset management is not a novel idea (see page 58). However, mobile sensing solutions that are exploited for a secondary purpose alongside their primary function, such as the system proposed in this chapter, are less widely deployed. One example is the pothole patrol system [67] that uses accelerometer data collected from taxis across the city of Boston. The primary function of the taxis is not changed — they pick up and drop off passengers as they normally would. However by aggregating accelerometer data collected from the entire fleet, the system is able to automatically detect potholes across the entire city.

Similarly, Carnelli et al. [38] developed a simulator to determine whether a taxi fleet may

be employed as a vehicular ad-hoc network (VANET) to deliver data packets across an urban environment. Each taxi is modelled as a mobile node that may transfer data to another taxi in close proximity, or download/upload data to a source/sink node. Historical taxi trip data collected in San Francisco and Rome is considered to explore the performance of a taxi-based VANET. Overall, their simulations found that doubling the size of the taxi fleet decreased the mean sink-to-source data delay (i.e., the time between data arriving at the source node to arriving at the sink node) by a factor of 0.6–0.7.

The objective of routing agents (a broad term often used in simulation that refers to the vehicles or individual people in question) to maximally cover an area is a widely studied problem. In the pioneering work of Machado et al. [167] the architectures and system considerations for maximal coverage multi-agent systems are tested within a simulation. The authors deconstruct a geographical area into a graph of nodes and edges, and introduce the concept of node 'idleness'; that is, the time between successive node visits from an any agent. Several subsequent works have continued to adopt the term of site or target idleness [4, 136].

The agents in [167] try to minimise the total node idleness of the graph, and thus maximally cover the graph. Various routing strategies, including a greedy algorithm in which each agent only considers the immediately neighbouring nodes, and more complex strategies in which the agents consider a larger neighbourhood of nodes, are tested. In addition, the authors tested different communication styles; that is, how the agents communicate the node idleness with each other. This work included decentralised methods where agents leave environmental 'flags' at nodes for other agents to sense, centralised communication with a control centre (and thus potentially, centralised coordination too), and communication via agent interaction. Our objective is slightly different in that the TO fleet considers the idleness of the highway segments (or edges), rather than nodes. Nonetheless, many of the topics covered in [167] will be relevant considerations in our work.

In the context of Unmanned Aerial Systems (UAS), a maximal coverage objective is known as the Persistent Surveillance Problem (PSP) [187]. One solution to the PSP is provided by Kent et al. [142] who developed a simulation in a 2-dimensional hexagonal world. Multiple UAS platforms aim to minimise the 'score' of each hexagonal cell that increases the longer it was last surveilled (i.e., the cell's idleness). Each platform is modelled as a single agent that can sense the environment (i.e., the idleness of each cell) and is given a policy that controls its routing decision. The authors considered five policies with varying complexity, however a relatively simple heuristic in which each platform travels to the neighbouring cell with the highest idleness was found to work as well as any other policy.

### 4.1.3 Summary of Background Material and Research Objectives

In summary, the existing relevant literature has two common themes; one, the transportation network or area to be patrolled is represented as a graph with nodes and edges, and two,

agents/vehicles that service dynamic demand (such as TOs) are deployed within a simulation. As such, in this chapter, we will build upon the rich literature and develop a simulation to consider the following research question (RQ, whose numbering convention is described in page 20):

RQ3.1 *Can a TO fleet cover the entire highway network under its jurisdiction, and thus perform asset management without reducing its incident response time?*

On a graphical representation of the highway network, we will deploy a TO fleet in a simulation where traffic incidents, that the fleet attend, occur across the network. Two different patrol regimes that alter the routing decision made by each TO will be considered. The first will try to minimise the fleet's incident response time, whereas the second will try to minimise the inter-visit time of each edge; that is, the time between successive visits from any TO (i.e., the edge idleness) and thus monitor assets across the highway network. From the simulation, under either regime, the incident response times and the edge inter-visit times will be compared to determine the feasibility of our proof-of-concept study.

## 4.2 Data Sources and Preparation

Having considered the international situation, we focus our study on the TO fleet deployed by HE on the SRN in England. We chose the HE TO fleet because data sets on the fleet, outstations, and the SRN are available with which to build our simulation. We now proceed to describe each of the data sources, see Table 4.1 for a summary.

**Traffic Officers**
Each vehicle in the HE TO fleet is recorded in the traffic officer fleet vehicles (TOFV) data set published by Highways England [120] in 2018. For each vehicle, the TOFV data set records its fleet (unique identification) number, make and model, and the outstation from which the vehicle begins and ends its patrol. In total, the TO fleet consists of 234 vehicles that patrol from 32 outstations across the SRN.

**Outstations**
The name of each of the 32 TO outstations across the SRN is listed on the HE website [122]. To determine the positions of the outstations, their addresses were obtained via an online government building database [88], which were then converted to an Easting-Northing [63] coordinate to the nearest metre.

**SRN**
The SRN is represented by the Highways Agency (now HE, see page 99) traffic information service (HATRIS) links [107] that describe junction-to-junction sections of the SRN. The HATRIS links were initially used by HE to report traffic flows between each junction until the service

TABLE 4.1. (a) Three links in the HATRIS data set. The position of junctions M40 J12 and M40 J13 are recorded differently in rows 2 and 3 as the links represent different sides of the carriageway. (b) Six TO vehicles recorded in the TOFV data set. (c) Positions of three HE outstations retrieved from a UK government building database [88].

(a)

| Row | Description | Start junction | End junction | Start Easting [m] | Start Northing [m] | End Easting [m] | End Northing [m] |
|---|---|---|---|---|---|---|---|
| 1 | A38 between A513 and A5127 | A38 A513 | A38 A5127 | 417216 | 314346 | 414414 | 310549 |
| 2 | M40 between M40 J12 and M40 J13 | M40 J12 | M40 J13 | 437249 | 254903 | 430537 | 260388 |
| 3 | M40 between M40 J13 and M40 J12 | M40 J13 | M40 J12 | 430534 | 260412 | 437262 | 254918 |

(b)

| Vehicle fleet number | Make | Model | Outstation |
|---|---|---|---|
| HE 1033 | Land Rover | Discovery SDV6 3.0 DSE | Heston |
| HE 1034 | Land Rover | Discovery SDV6 3.0 DSE | Heston |
| HE 0851 | Mitsubishi | Shogun 3.2 DI-DC M | Heston |
| HE 0921 | Mitsubishi | Shogun 3.2 DI-DC M | Samlesbury |
| HE 0945 | Mitsubishi | Shogun 3.2 DI-DC M | Samlesbury |
| HE 0981 | Land Rover | Discovery SDV6 3.0 DSE | Milnrow |

(c)

| Outstation | Easting | Northing |
|---|---|---|
| Knutsford | 373433 | 378337 |
| Milnrow | 393036 | 411899 |
| Watford | 458602 | 270068 |

was upgraded in 2016 to provide live traffic information along shorter inter-junction highway sections [104].

For each link, its highway name (e.g., M6), and the start and end junction name (e.g., J1) and position as an Easting-Northing coordinate (to the nearest metre) are recorded. Junctions along multi-lane highway and major A-roads are named by their junction number (M6 J1, for example), however junctions along smaller trunk A highways are defined by the connecting road at the junction. For example, junction A38 A515 represents the junction where the A38 connects to the A515 (which is not part of the SRN). In total, 2,510 links are recorded in the HATRIS data set.

**OpenStreetMap**

All map-based figures presented in this chapter are generated via the OpenStreetMap (OSM) service — a free online and collaborative mapping resource [191].

### 4.2.1 Graphical Highway Network Model

We now proceed to construct a graphical representation of the SRN from the HATRIS links. The recorded positions for each HATRIS junction with the same name are collected, and their pairwise distances are computed. Those pairs for which the distance is greater than 4 km are then flagged for manual inspection and cleaning, since it seems likely that these are in fact different junctions with the same name. This duplication may arise for one of two reasons:

1. the junction has been renamed and the HATRIS link has not been updated; or
2. pairs of 'A' roads meet at a number of different places.

Overall, four junctions were found to have an incorrect name (reason 1) and were subsequently renamed: J43, J44 and J45 on the A1(M) motorway (that were all renamed in 2009 [102]) and junction J8 on the M42.

A further 47 'A' road junctions were identified for cleaning due to reason 2. In each case, these junctions are given an additional suffix to distinguish between them (i.e., to represent each location at which the pairs of 'A' roads meet). For example, the A64 highway meets the York circular highway (A1237) at two different junctions both named A64 A1237 either side of the city. Thus, the two junctions are renamed A64 A1237(a) and A64 A1237(b), see Figure 4.3(b).

A graphical highway model of the SRN is then constructed from the cleaned HATRIS links and junctions. The junctions provide the graph's nodes and each HATRIS link represents a directed edge between those nodes. The driven distance for each edge is then modelled crudely by the straight-line distance between its nodes, based on their coordinates.

Unfortunately, the HATRIS data does not provide unambiguous junction coordinates: see Table 4.1(a), rows 2 and 3, which represent the opposing directions of the same dual carriageway section. Because of the lateral physical scale, different coordinates are recorded for the junction position, for each side of the carriageway. Our approach is thus simply to take the average of all of the coordinates provided for a given junction, see Figure 4.3(a).

The resulting graph, see Figure 4.4, consists of 1,108 nodes and 2,376 directed edges, with a total length of 13,048 km. Highways England report a total network length of 13,760 km [117]; the small difference can be accounted for by our neglection of link curvature.

## 4.3 Concepts of Operation

We now proceed to describe and simulate the proposed operational mode of the TOs. We will use the term FTO (future traffic officer) to distinguish between the new concepts of operation in the new system that we propose, and the current real-world practice.

Multi-lane highways usually have a central reservation and TOs can thus only re-route at junctions. Thus, we describe their operation in terms of a node-to-node routing methodology on the graphical model of the highway developed in Section 4.2.1.

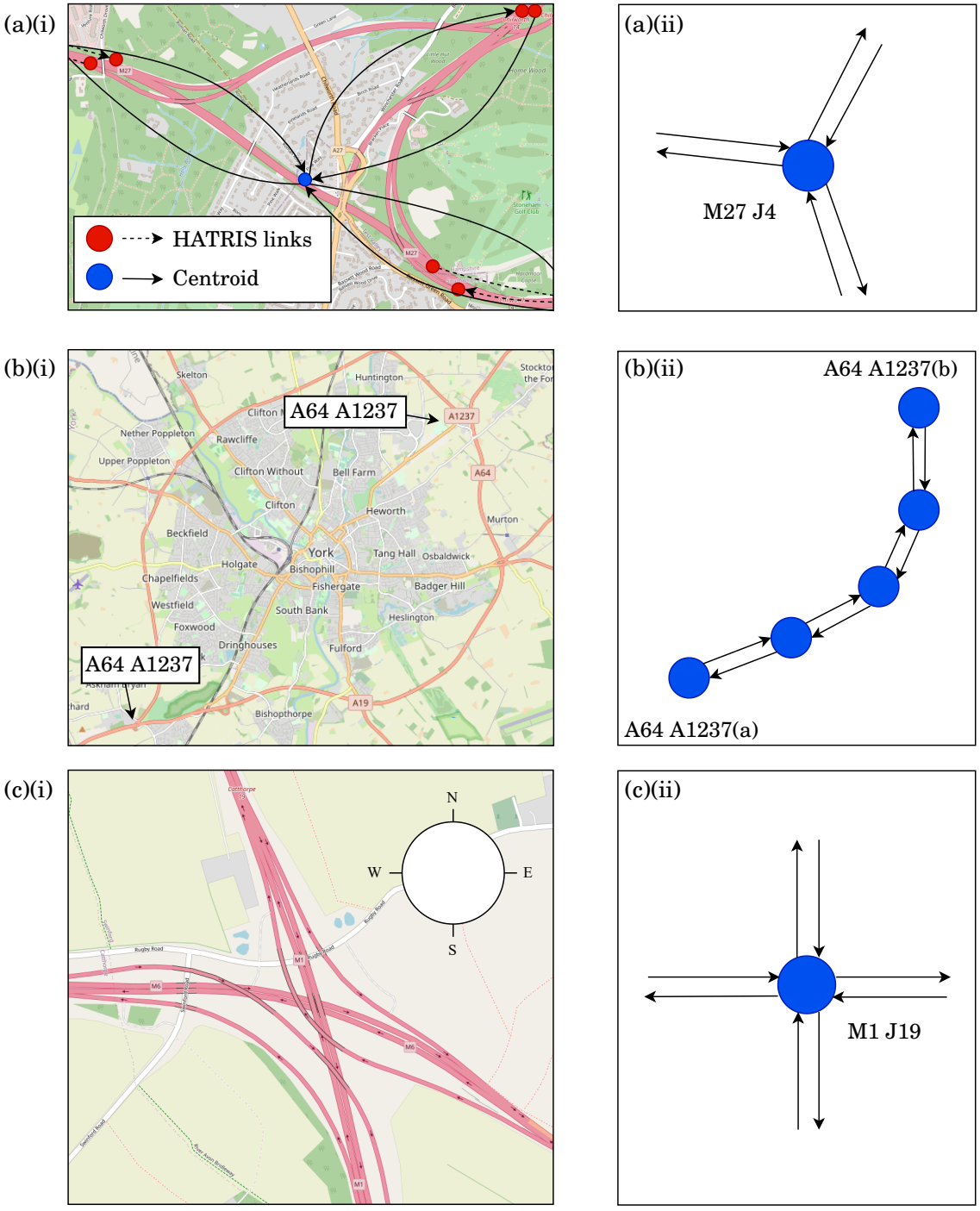



FIGURE 4.3. HATRIS juntions and links (i) and their graphical representation (ii). (a) HATRIS links that begin or end at M27 J4. The recorded position of the junction in all six links is set to their centroid (blue). (b) The A64 highway meets the A1237 highway at two different junctions both named junction A64 A1237. The two junctions are subsequently renamed. (c) Junction M1 J19, whose drive-through structure is not captured in the graphical model. A driver travelling northbound or southbound towards junction M1 J19 may only merge onto the westbound and eastbound carriageways respectively, or vice versa.

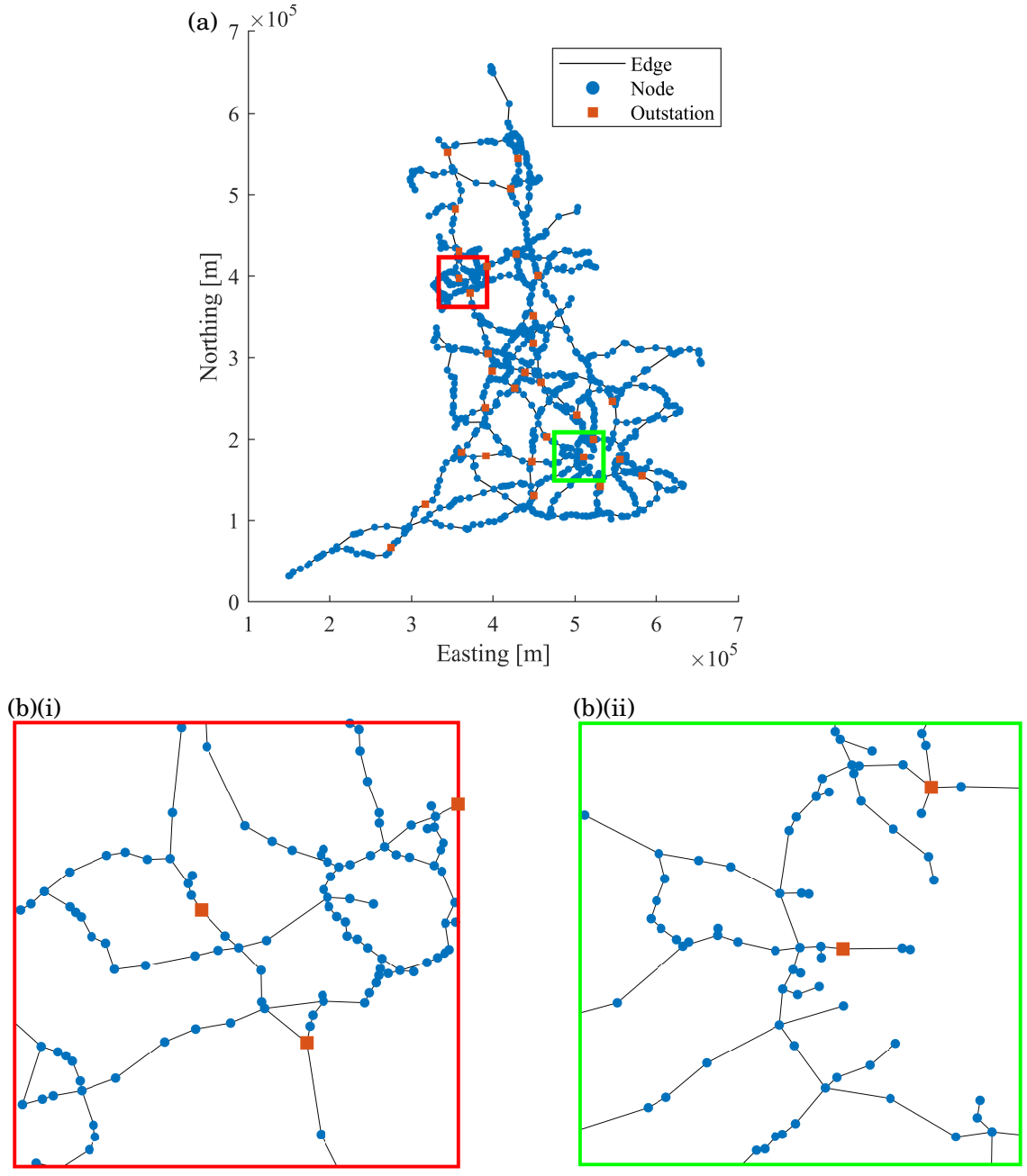



FIGURE 4.4. (a) The SRN graph constructed from the HATRIS links and (b) two zoomed areas. For illustration purposes, each edge in this figure represents two directed edges (one for each direction of travel). The nodes closest to each outstation are depicted with orange squares.

Note that as each junction is collapsed onto a single node, neither its size or drive-through structure is modelled in this work (these generalisations remain for future work). As an FTO reaches a node, it is assumed that they may instantaneously chose which of the connecting edges to patrol along next. In reality, it will take the TO some time to travel around the junction and furthermore, some junctions are constructed such that different highways can be joined depending on the direction of travel, see Figure 4.3(c). However, junctions are small compared with the length of highway between them and maximising the fleet's coverage on the SRN (for asset management) can be achieved by considering the edge coverage only.

Upon reaching a node, we suppose that each FTO will make one of the following decisions:

1. continue on the same highway;
2. join a new highway;
3. turn around and drive along the same highway in the opposite direction; or potentially,
4. stop for a rest period, albeit remaining available during this time for incident response.

The key idea is that different patrol regimes (that prioritise either incident response or asset management) will influence this routing decision.

In notation: the state of FTO $k$ is fully described by the node it is driving to, and the time it will arrive (and simultaneously depart from) there, respectively denoted by $n[k]$ and $a[k]$. The edge from node $i$ to $j$ has inter-visit time (idleness), $v(i,j)$, and driven distance, $d(i,j)$ (recall that this is modelled by the node-to-node straight-line distance). The distance matrix, $\boldsymbol{D}$ describes the pairwise node shortest path length, and the trip time matrix is given by $\boldsymbol{T} = \boldsymbol{D}/s$, where we suppose that $s = 100$ kph is the constant FTO speed (i.e., the effect of congestion on travel times is not taken into account — again, a generalisation that remains for future work).

### 4.3.1 Traffic Officer Patrol Regimes

We analyse three patrol regimes:

1. response (R1),
2. coverage (R2), and
3. random (R3).

The idea is that regime R1 aims to minimise the fleet's collective incident response time, whereas regime R2 aims to maximise the fleet's coverage for asset management. In regime R3, FTOs will select their next edge at random, and this thus acts (in the absence of breadcrumb data from the current HE TO system) as a base case which regimes R1 and R2 must out-perform.

Under regime R1, see Figure 4.5(a), FTO $k$ considers a neighbourhood $\mathscr{N}$ of close nodes, within a driven distance $\delta_n$ of node $n[k]$, and a set $\mathscr{F}$ of close FTOs, within a driven distance $\delta_f$

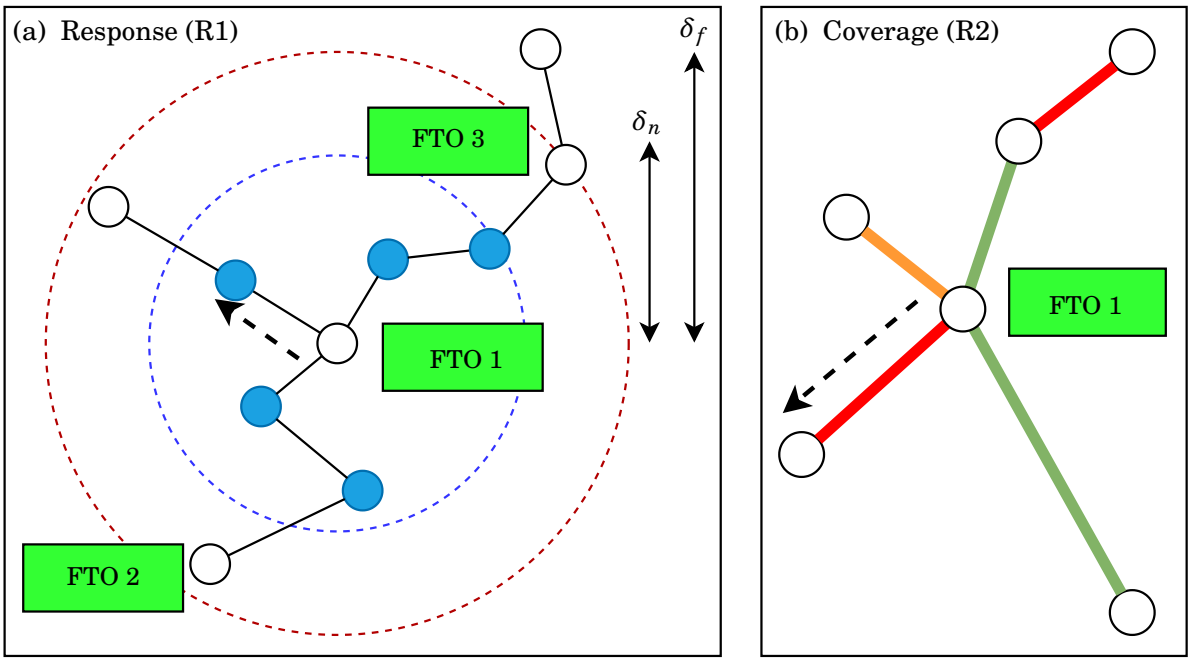



FIGURE 4.5. (a) Under regime R1 FTO 1 selects the edge to minimise the total weighted minimum response time to a nearby node neighbourhood (blue), from either itself, FTO 2, or FTO 3. (b) Under regime R2 FTO 1 selects the edge with the largest inter-visit time (red).

of node $n[k]$. The idea is that FTOs continuously attempt to collectively minimise their drive time to any node within their near neighbourhood, in case an incident should occur there in the near future. This is achieved in a distributed way by each FTO selecting their next edge $e^*$ according to

$$e^* = \arg\min_{e} \sum_{n \in \mathcal{N}} \bar{r}_n, \tag{4.1}$$

where

$$\bar{r}_n = w_n \min(\{r_e(k,n) \cap r(f,n) : f \in \mathcal{F}\}) \tag{4.2}$$

is the weighted minimum incident response time (WMIRT), and

$$r_e(k,n) = \boldsymbol{T}(n[k], n_e) + \boldsymbol{T}(n_e, n), \tag{4.3}$$

where $n_e$ denotes the destination node of edge $e$, is the response time of FTO $k$ to node $n$ if it chooses to patrol along edge $e$. The response time of TO $f$ to node $n$ is denoted by $r(f,n)$. Note $w_n$ is a weight attached to node $n$ that models its relative frequency in experiencing incidents.

In contrast, in regime R2, see Figure 4.5(b), FTO $k$ selects the connecting edge with the largest inter-visit time, that is

$$e^* = \arg\max_e [v(n[k], n_e)]. \tag{4.4}$$

In regime R3, the edge $e^*$ is chosen uniformly randomly from the allowed options. Once FTO $k$ has chosen which edge $e^*$ to patrol along, their state is updated according to $n[k] = n_{e^*}$ and $a[k] = a[k] + \boldsymbol{T}(n[k], n_{e^*})$, and the inter-visit time of the edge is updated such that $v(n[k], n_{e^*}) =$ 0 minutes.

### 4.3.2 Incident Response Mechanism

From time to time, incidents will occur and the FTOs will break off from their usual patrolling pattern. We suppose that the Bell and Wong Nearest Neighbour (BWNN) heuristic [264] is employed to assign the nearest FTO to the incident and thus minimise the response time. In notation, given an incident at time $t$ along an edge from node $i$ to node $j$, BWNN assigns a single FTO, $k^*$, by

$$k^* = \arg\min_k [r(k, i)]. \tag{4.5}$$

Here,

$$r(k, i) = \max(0, a[k] - t) + \boldsymbol{T}(n[k], i) + t^i_{\text{inc}} \tag{4.6}$$

is the time for FTO $k$ to drive to node $i$, plus the time taken to drive from node $i$ along the edge to the incident, denoted $t^i_{\text{inc}}$. Correspondingly, $t^j_{\text{inc}}$ denotes the time to drive from the incident to the edge's destination node.

We suppose that asset monitoring continues while the selected FTO drives to the incident, and thus the inter-visit times of the edges along the shortest path from node $n[k^*]$ to node $i$, and the edge from node $i$ to node $j$, are set to zero as the FTO passes them. Moreover, we suppose that the assigned FTO is unavailable to respond to further incidents before clearing its current assignment. (Of course, one might consider much more sophisticated operations, where for example a multi-vehicle collision requires that multiple FTOs are deployed and taken from other less urgent assignments — this is beyond our current scope).

Once the incident is cleared, we assume that the FTO $k^*$ drives to node $j$, and thus $n[k^*] = j$ and

$$a[k^*] = t + r(k^*, i) + t_{\text{TTC}} + t^j_{\text{inc}}, \tag{4.7}$$

and then resumes their usual patrolling regime. Here, $t_{\text{TTC}}$ models the incident's time-to-clear (TTC) [96].

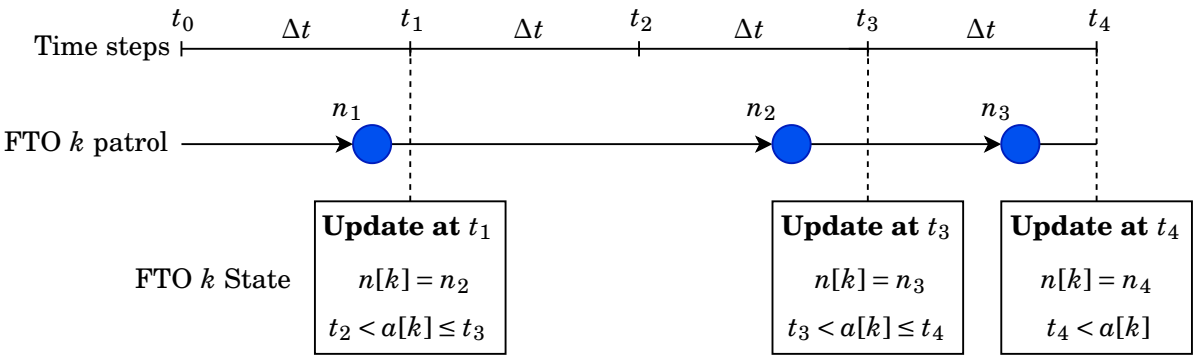



FIGURE 4.6. FTO update logic. FTO $k$ patrols from node $n_1$ to node $n_2$ to node $n_3$ (blue) to node $n_4$ (not shown). The FTO's state ($n[k]$ and $a[k]$) is updated retrospectively at times $t_1$, $t_2$, and $t_3$.

## 4.4 Simulation Methodology

We now proceed to describe the simulation in which we deploy our proposed system.

### 4.4.1 Initialisation and Time Stepping

At the beginning of the simulation, the inter-visit time of each edge is set to zero minutes; essentially, assets across the entire highway network begin in a perfect state (as if they have just been visited) and now require monitoring throughout the new patrol. Then, inspired by the day-night patrol pattern employed by HE and RWS [19, 221], each simulated FTO begins a 12-hour patrol shift at $t = 0$ minutes from the node closest to its outstation. Each FTO must return to this node by the end of the shift and is thus constrained by an operational radius that shrinks throughout the simulation. However, we allow an FTO to leave their operational area to respond to an incident.

While on patrol, HE TOs drive for half of their shift and rest at junctions or parking areas for the other half (HE, personal communication). To model this, our simulated FTOs repeatedly drive for one hour and then rest at a node for one hour. The first rest time of each FTO is randomly assigned within the first hour of the simulation to ensure that the entire FTO fleet is not resting at the same time. All simulated FTOs patrol under the same regime throughout the 12-hour patrol.

The simulation time step is denoted as $\Delta t$ and in each time step the update rules (see Section 4.3) are applied to each simulated FTO in sequence. At time step $m$, provided $m\Delta t < a[k]$, FTO $k$ will proceed along its current edge, provided it is not called to an incident. The node decision logic is then applied when $m\Delta t \geq a[k]$, see Figure 4.6, and the resulting excess time, $m\Delta t - a[k]$, is subtracted from the arrival time at its next node.

In Section 4.5.4, the incident response times and edge inter-visit times collected from an ensemble of simulations, for various time steps between 5 seconds to 1.5 minutes, are considered. Overall, the simulations were found to be robust (i.e., the incident response times and edge inter-visit times were unchanged) when the time step was refined. Thus, a time step of $\Delta t = 1$ minute, that provides an appropriate simulation run time (around 20 seconds to simulate the 12 hour patrol with 234 FTOs) is chosen. A top-level simulation architecture is provided in Figure 4.7.

The sequential updating methodology used in our simulation may introduce artefacts as we intend that the FTOs make independent decisions, but, in any given update, those FTOs who update last consider different edge inter-visit times and FTO states than those FTOs who update first. Thus, it is good practice that sequential updating processes are parallelised. However, over an ensemble of simulations (20 runs per regime, R1 with $\delta_n$=$\delta_f$=80 km) with time step $\Delta t = 1$ minute, the FTOs make a routing decision every 2.5 minutes, or 4.2 km on average (the average edge length is 5.49 km) and an FTO makes a single routing decision (rather than multiple ones) at each time step update 94% of the time. Hence, the choice of update protocol is likely to be inconsequential.

Moreover, we find, on average, an FTO makes a routing decision in isolation (that is, there are no other FTOs rerouting at the same time at a node within a 10 km driven-distance radius) 55% of the time. Furthermore, only one other FTO simultaneously reroutes within this driven-distance radius for 36% of updates.

As the number of FTOs that make multiple routing decisions in one time step is relatively small, and the majority of routing decisions are made in isolation, or when only one other FTO is in close proximity, updating the FTOs in sequence is an appropriate simulation methodology. Future work, that considers other TO fleets and transportation networks (rather than the HE TOs and SRN considered here) may seek to improve the simulation methodology. This might be achieved by continueing to perform the routing decisions in sequence, but only updating the edge inter-visit times and FTO states once all FTOs have made their decision.

### 4.4.2 Incident Model

We now describe how incidents are generated within our simulation. Vehicle incidents on highways are often modelled by a Poisson process in space and time [36, 140, 160] that models each incident as an independent event. Furthermore, the rate at which a given edge suffers incidents is proportional to its length [36, 258].

In any time step, the probability of there being one event on an edge, from node $i$ to node $j$, is approximately $\lambda d(i,j)\Delta t$. Since this scales with small $\Delta t$, we take the usual approach and suppose that the probability of more than one event on a given edge in one (small) time step is zero. Each incident is randomly positioned along the edge and is given HE's target time-to-clear (TTC) of $t_{\text{TTC}} = 60$ minutes [188].

In 2015, HE TOs attended 215,568 incidents, equal to 0.41 incidents per minute [20]. Di-

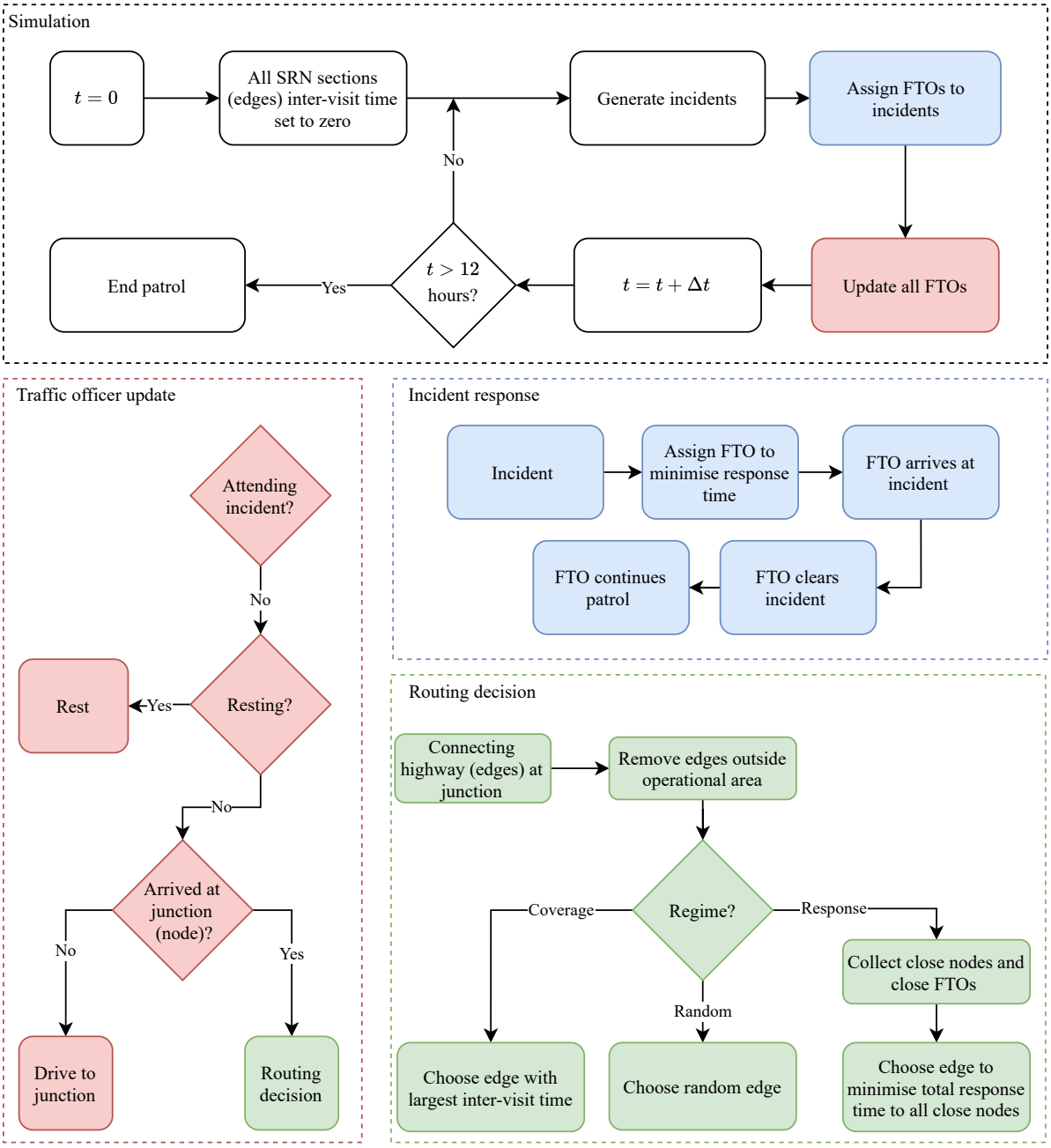



FIGURE 4.7. Simulation architecture. The FTOs patrol for a 12-hour shift and respond to incidents as they are generated (blue). Upon reaching a node, provided they are not resting, each FTO chooses which edge to patrol along next via the routing decision process (green) that is influenced by the FTO's patrol regime.

viding this quantity by the total length of the SRN yields an incident rate $\lambda = 3.14 \times 10^{-5}$ incidents per km per minute.

### 4.4.3 Inspection of the Response Regime and Node Weightings

Under regime R1, each FTO considers the position of other nearby FTOs, to minimise the total WMIRT to a nearby node neighbourhood, see Equations 4.1 and 4.2. As the number of incidents on an edge is proportional to its driven distance, and each incident is randomly positioned along the edge, we model the weight $w_n$ of node $n$ by the total driven distance of its outgoing edges.

FTOs patrolling under regime R1 emerge into a kind-of formation where each FTO patrols close to a response node; namely, a node from which the total WMIRT to all nodes in the neighbourhood is minimised.

When an FTO is assigned to an incident, they leave their response node and break the formation, then, the remaining unassigned FTOs emerge into a new formation. This reformation process is demonstrated in Figure 4.8 with a small fleet of three FTOs on a grid-like test graph, with 17 nodes and 28 edges each with length 10 km. For this test case, $\delta_n$ is set such that each FTO's node neighbourhood is the entire graph, and $\delta_f$ is set such that each FTO may always communicate with the other two.

Three nodes in the test graph are given an increased weight of 100 whereas all other nodes have weight 10, and thus, the FTOs weight the minimum response time to these three nodes more heavily. Note that in this test case scenario, the weights are assigned independently of the edge length.

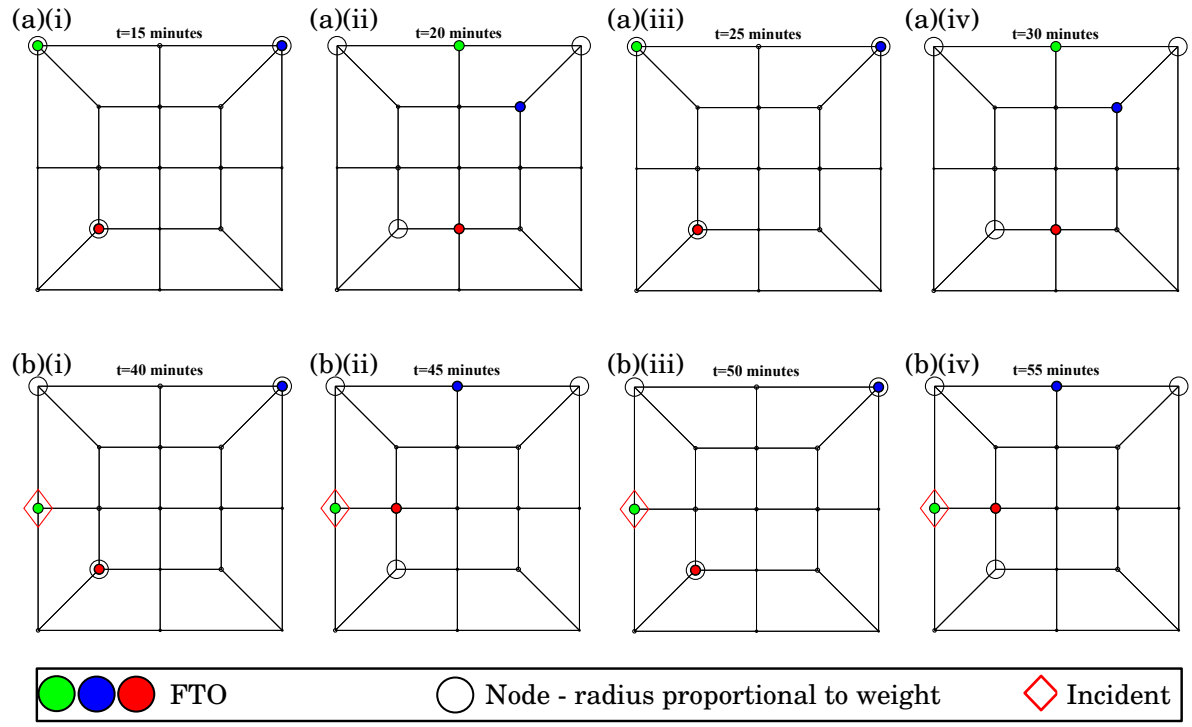
### 4.4.4 Designing the Response Regime Parameters

We now design the R1 regime's two free parameters, $\delta_n$ and $\delta_f$, that are set before the simulation is initialised and are not changed.

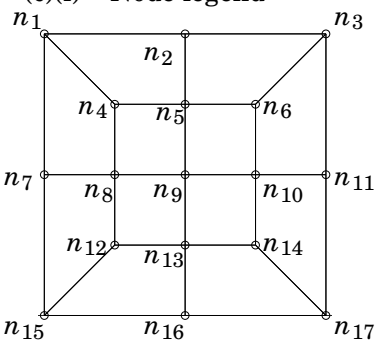
Intuitively, optimal parameters choices will minimise the total WMIRT (see Equation 4.2) across the entire SRN graph such that an FTO may quickly respond to any incident. Unfortunately, a large number of simulations (with dynamically generated incidents) are required to compute robust statistics on the fleet's incident response time to every node. Instead, for every node, we consider the response time as if an incident were to occur there. The FTO fleet is deployed in our simulation as described in Section 4.4.1, however, the incident generation model (see Section 4.4.2) is not implemented. Then, the total WMIRT of every node, over the entire 12-hour patrol,

$$R = \sum_{t} \sum_{n \in \mathcal{N}_{\text{SRN}}} w_n r(k^*, n), \tag{4.8}$$

is computed for various $\delta_n$ and $\delta_f$. Here, $\mathcal{N}_{\text{SRN}}$ is the set of nodes in the SRN graph and $k^*$ is the FTO with the minimum response time to node $n$, see Equation 4.5. For each pair of parameter values, the patrol is simulated 10 times (without incidents) and the average, $\hat{R}$, is computed.




(c)(i) Node legend




(c)(ii)

| node | $n_1$ | $n_2$ | $n_3$ | $n_4$ | $n_5$ | $n_6$ |
|---|---|---|---|---|---|---|
| weight ($w_n$) | 100 | 10 | 100 | 10 | 10 | 10 |
| node | $n_7$ | $n_8$ | $n_9$ | $n_{10}$ | $n_{11}$ | $n_{12}$ |
| weight | 10 | 10 | 10 | 10 | 10 | 100 |
| node | $n_{13}$ | $n_{14}$ | $n_{15}$ | $n_{16}$ | $n_{17}$ | |
| weight | 10 | 10 | 10 | 10 | 10 | |

FIGURE 4.8. Three FTOs on a test network under regime R1. Nodes $n_1$, $n_3$, and $n_{12}$ are assigned weight 100, whereas the rest have weight 10 (see node legend and weights in the (c) panels). At the beginning of their patrol (a), the FTOs emerge into an initial formation. At $t = 40$ minutes the green FTO responds to an incident, and the two unassigned FTOs arrange themselves into a new formation (b).

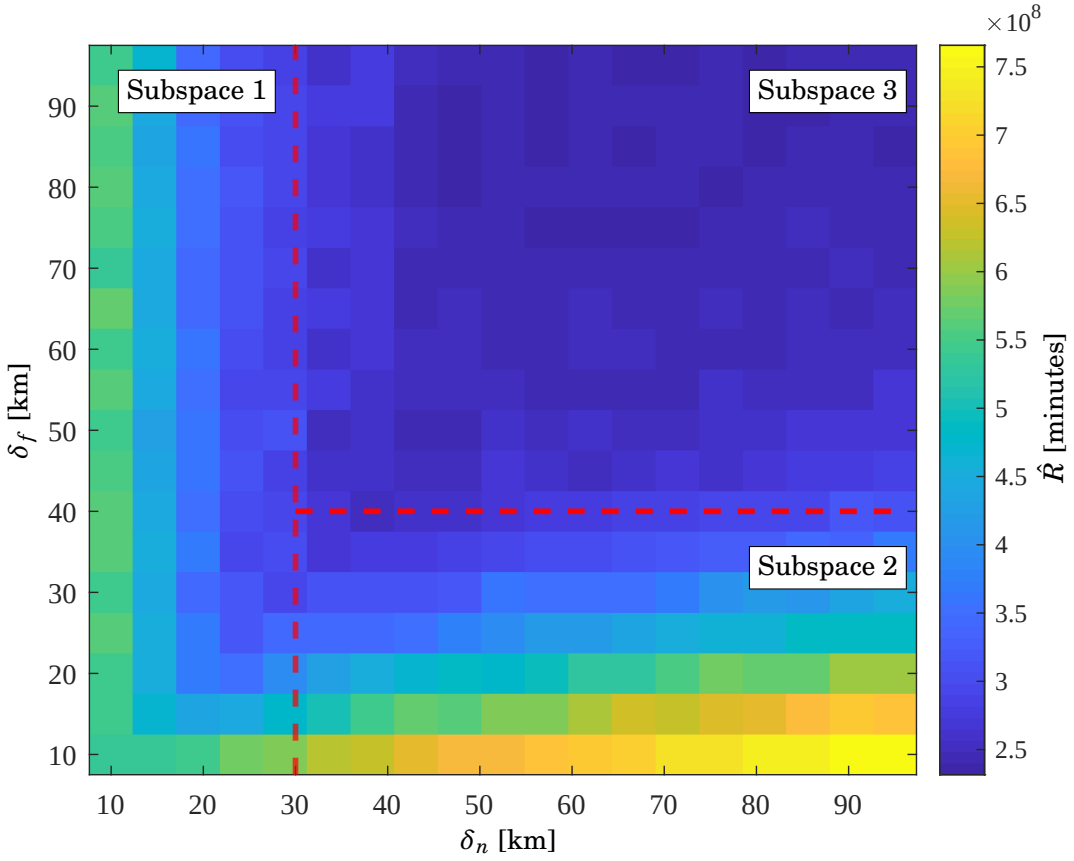



FIGURE 4.9. The approximation to incident response time, $\hat{R}$, for varying $\delta_n$ and $\delta_f$. Small values of $\hat{R}$ indicate that the FTO fleet is distributed at response nodes across the SRN graph. In subspace 1, $\mathcal{N}$ is relatively small and thus FTOs patrol at local response nodes. In subspace 2, where the FTOs have a low level of communication, $\hat{R}$ increases with increasing $\delta_n$. As the size of $\mathcal{N}$ increases, the FTOs patrol at the same response nodes. $\hat{R}$ is minimised in subspace 3 where $\delta_n$ and $\delta_f$ are greater than their critical values (red).

As discussed in Section 4.4.3, when an FTO responds to an incident, the formation around response nodes is broken and the remaining unassigned FTOs rearrange themselves into a new formation. However, the approximation to incident response time, $\hat{R}$, is computed without incidents and thus it does not capture this reorganisation. Rather, the approximation considers the first emergent formation that the FTOs arrange themselves into at the beginning of the patrol. In addition, the approximation does not consider that an FTO may not respond to an incident if it is already assigned to an incident elsewhere. Despite these simplifications to the FTO patrol strategy, $\hat{R}$ is a useful quantity with which to examine regime R1 under various parameter values, and moreover, it allows us to compute robust statistics on the response times

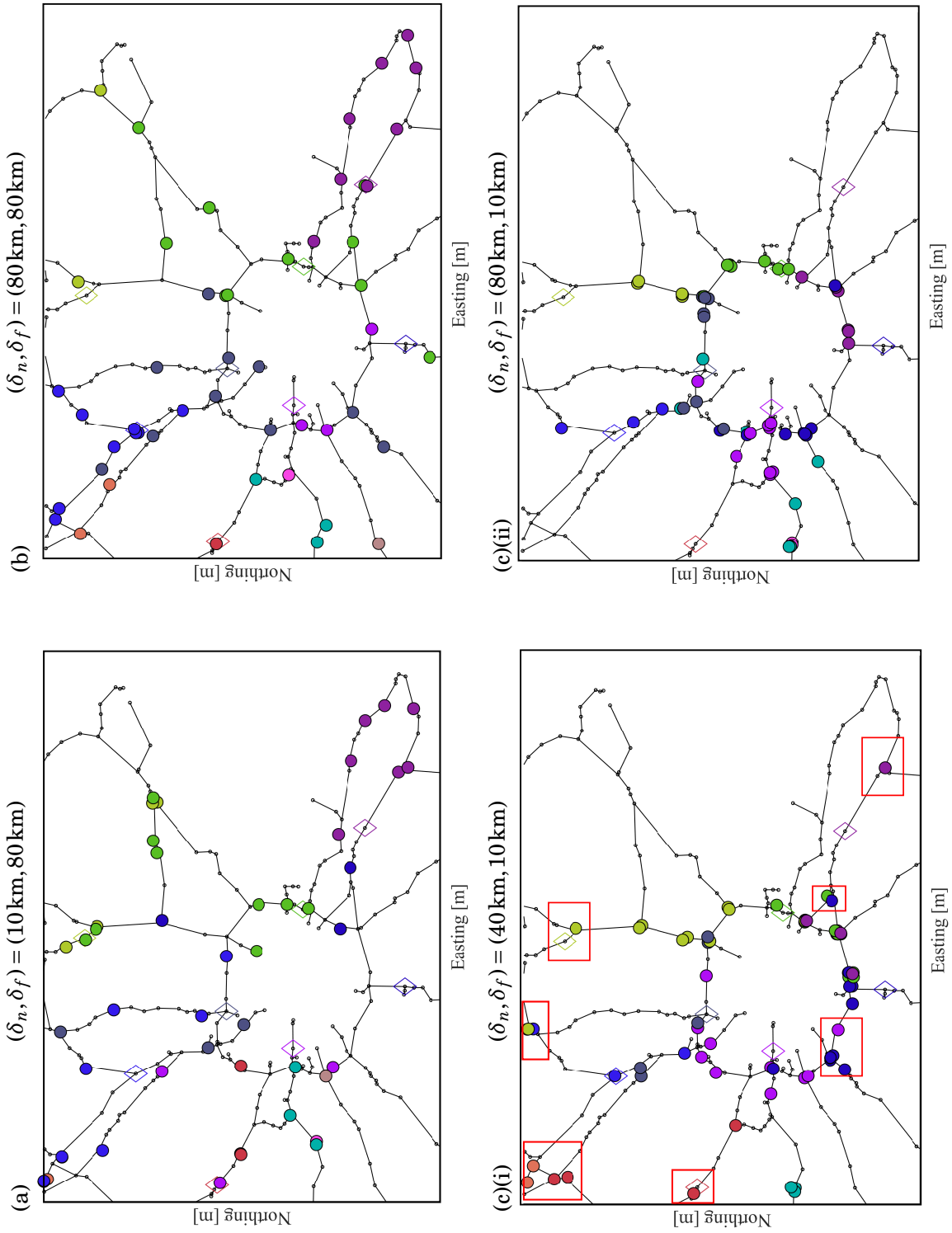



FIGURE 4.10. FTO emergent formations at $t$ = 200 minutes in the South East of England for various $\delta_n$ and $\delta_f$. (a) Subspace 1 and (b) Subspace 3. In Subspace 2, smaller values of $\delta_n$ (c)(i) cause the FTOs to patrol at relatively local response nodes (red boxes), compared with larger values of $\delta_n$ (c)(ii). The colour of each FTO corresponds to their outstation (diamonds).

to all nodes.

The approximation $\hat{R}$ for various values of $\delta_n$ and $\delta_f$ is shown in Figure 4.9. The figure can be broadly split into three subspaces. In subspace 1, the neighbourhood of close nodes considered by each FTO $\mathscr{N}$ is small ($\delta_n < 30\,\text{km}$), thus, each FTO minimises the incident response time to a relatively local area and the fleet is unable to distribute across the graph.

In subspace 2, where the size of $\mathscr{N}$ is larger ($\delta_n > 30\,\text{km}$) but the number of other FTOs that each FTO communicates with $\mathscr{F}$ is small ($\delta_f < 40\,\text{km}$), $\hat{R}$ increases with increasing $\delta_n$. This, perhaps counter-intuitive, result is a consequence of the low level of communication between the FTOs. Consider the extreme case where $\mathscr{N}$ is every node in the SRN graph and the FTOs do not communicate at all. In this scenario, every FTO patrols at the same central response node. As $\delta_n$ decreases, each FTO patrols at an increasingly local response node and thus the fleet distributes across the SRN graph, and thus, $\hat{R}$ decreases.

In subspace 3, the parameters are larger than the critical values $\delta_n = 30\,\text{km}$ and $\delta_f = 40\,\text{km}$. Ultimately, each FTO performs an edge-by-edge routing decision, and thus increasing the size of $\mathscr{N}$ or $\mathscr{F}$ does not result in an improved routing decision at this relatively local scale. In the subsequent sections of this chapter, the parameter values are set in subspace 3, such that $\delta_n = \delta_f = 80\,\text{km}$. Illustrations of the R1 regime emergent formations for parameter values in each subspace are provided in Figure 4.10.

## 4.5 Simulation Metrics and Analysis

We now deploy and consider the performance of the FTOs under all three patrol regimes in our simulation. The incident response times and edge inter-visit times are compared over an ensemble of 50 simulations under each of the R1, R2, and R3 regimes (150 in total). Four example patrol routes from the ensemble are illustrated in Figure 4.11. Specifically, at simulation time $t$, we consider the ensemble-average incident response time (AIRT) and edge inter-visit time (AIVT), respectively computed by

$$\text{AIRT}(t) = \frac{1}{|\mathscr{N}_{\text{inc}}|} \sum_{n \in \mathscr{N}_{\text{inc}}} r(k^*, n) \tag{4.9}$$

and

$$\text{AIVT}(t) = \frac{1}{|\mathscr{E}_{\text{SRN}}|} \sum_{(i,j) \in \mathscr{E}_{\text{SRN}}} v(i,j). \tag{4.10}$$

Here, $\mathscr{N}_{\text{inc}}$ denotes the set of start nodes of those edges where an incident occurs at time $t$, (and attended by FTO $k^*$, see Section 4.3.2). Nodes $i$ and $j$ respectively denote the start and end node of each edge in the set of SRN graph edges $\mathscr{E}_{\text{SRN}}$.

The AIVT is a suitable metric to consider for those assets that require immediate management (and thus continuous monitoring), such as debris on the highway surface. However, a daily asset data capture was specified by HE as a suitable monitoring frequency for assets whose condition

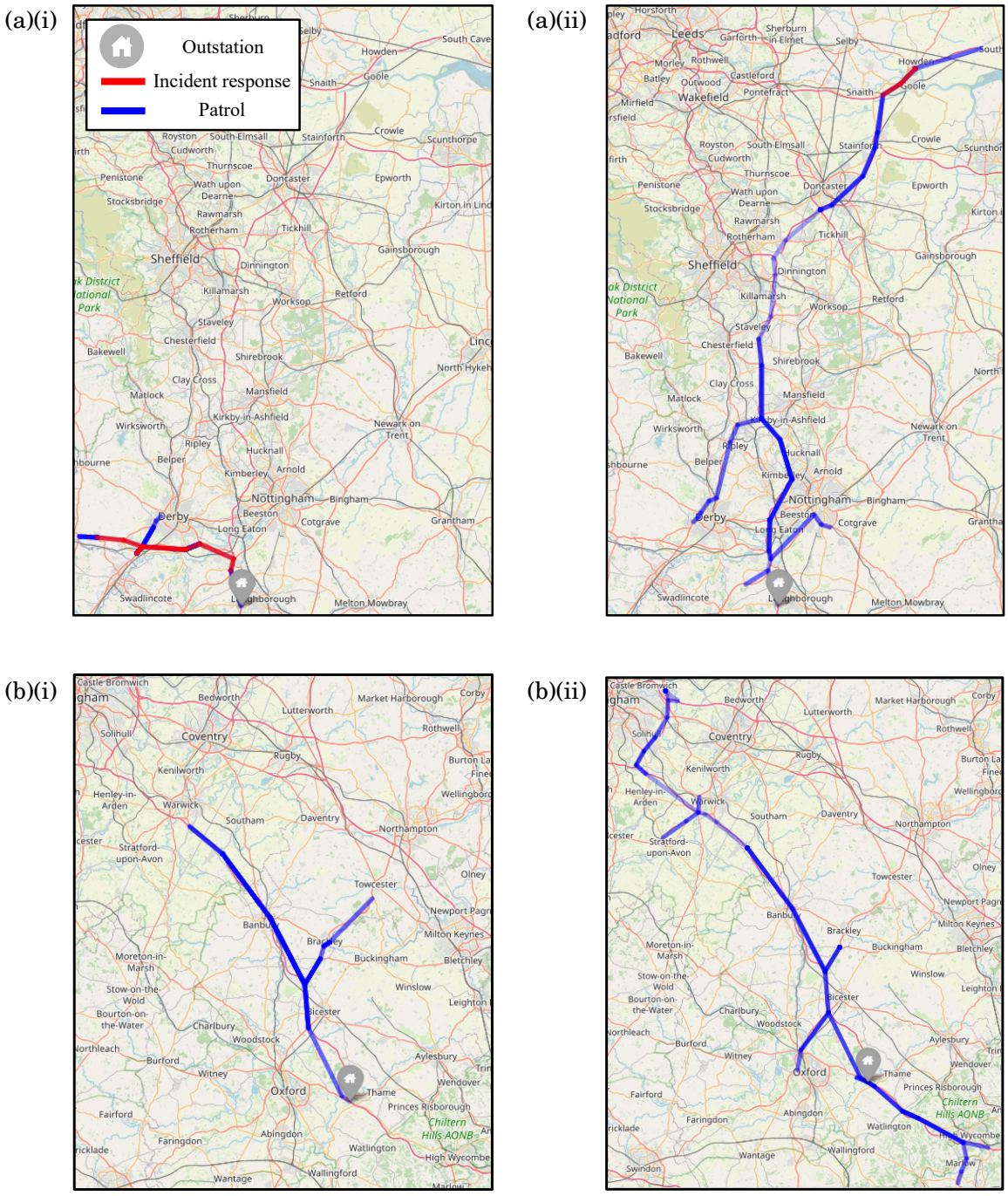



FIGURE 4.11. FTO patrol routes from the (a) Shepshed and (b) Milton Common outstations. The (i) and (ii) panels respectively correspond to patrols under regimes R1 and R2. Blue sections of highway indicate the FTO's usual patrolling pattern whereas red indicates incident response: a darker colour illustrates where the FTO has regularly patrolled.

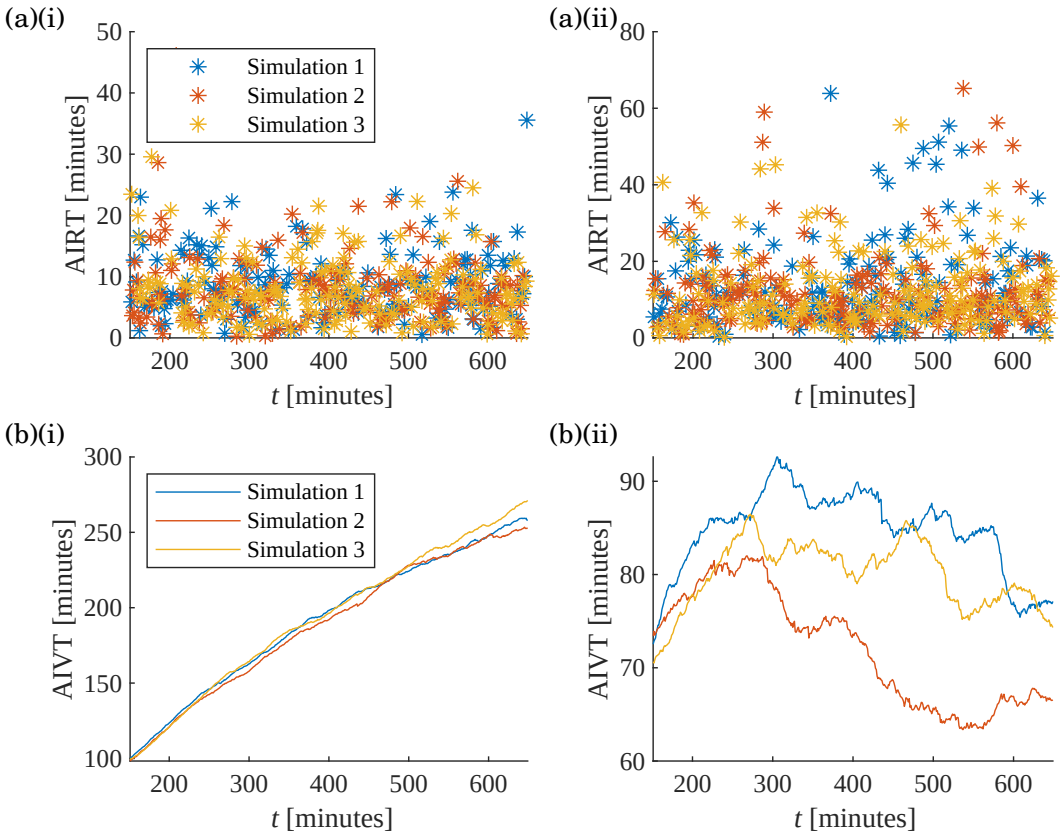



FIGURE 4.12. (a) AIRT and (b) AIVT for three simulations within the ensemble. Panels (i) and (ii) correspond respectively to patrols under regimes R1 and R2. For illustration purposes, the AIRTs at each simulation time step are depicted with a scatter point.

might rapidly change but do not require immediate management, such as a broken street-light for example (HE, personal communication). Therefore, in addition to the AIVT, we also compute the ensemble-average percentage of edges that are visited at least once by an FTO (APV) during the 12-hour patrol. Specifically, at simulation time $t$, the APV is computed by

$$\mathrm{APV}(t) = 100 \times \frac{1}{|\mathscr{E}_{\mathrm{SRN}}|} \sum_{(i,j)\in\mathscr{E}_{\mathrm{SRN}}} V(i,j) \tag{4.11}$$

where

$$V(i,j) = \begin{cases} 1 & \text{if } v(i,j) < t, \\ 0 & \text{otherwise.} \end{cases} \tag{4.12}$$

The AIRTs and AIRVs for three simulations under R1 and three simulations under R2 are

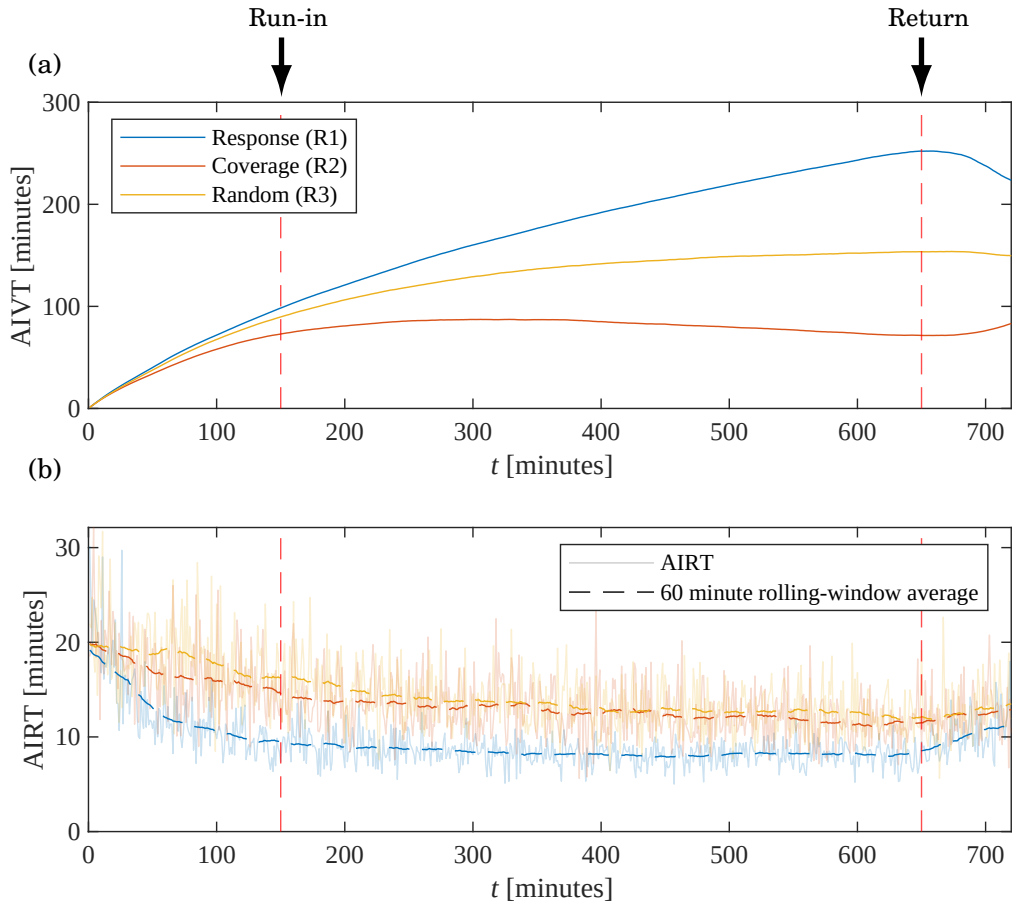



FIGURE 4.13. (a) AIVT and (b) AIRT along with a 60 minute rolling-window average under each regime. The simulations exhibit a run-in period ($t < 150$ minutes) and a return period ($t > 650$ minutes) as the FTOs leave and return to their outstations.

shown in Figure 4.12. Each simulation represents a different 12-hour time period with a unique set of generated incidents, and thus, the metrics vary within the ensemble.

### 4.5.1 Simulation Run-in and Return Period

The ensemble AIVT and AIRT at each simulation time step are shown in Figure 4.13. The simulations exhibit a run-in period during the first 150 minutes of the patrol as the FTOs depart from their outstations and patrol on a small proportion of the SRN graph. The FTOs return to their outstations during a return period ($t > 650$ minutes). Consequently, under regime R2, the AIVT is increased as each FTO is restricted by their operational radius. On the other hand, under regime R1, the AIVT is decreased as the FTOs break their emergent formations. The AIRT is increased under all regimes during both the run-in and return period.

To assess the performance of the regimes while avoiding the impact of the run-in and return periods, we compute and compare the time-averaged AIRT (TAIRT) and AIVT (TAIVT) in the

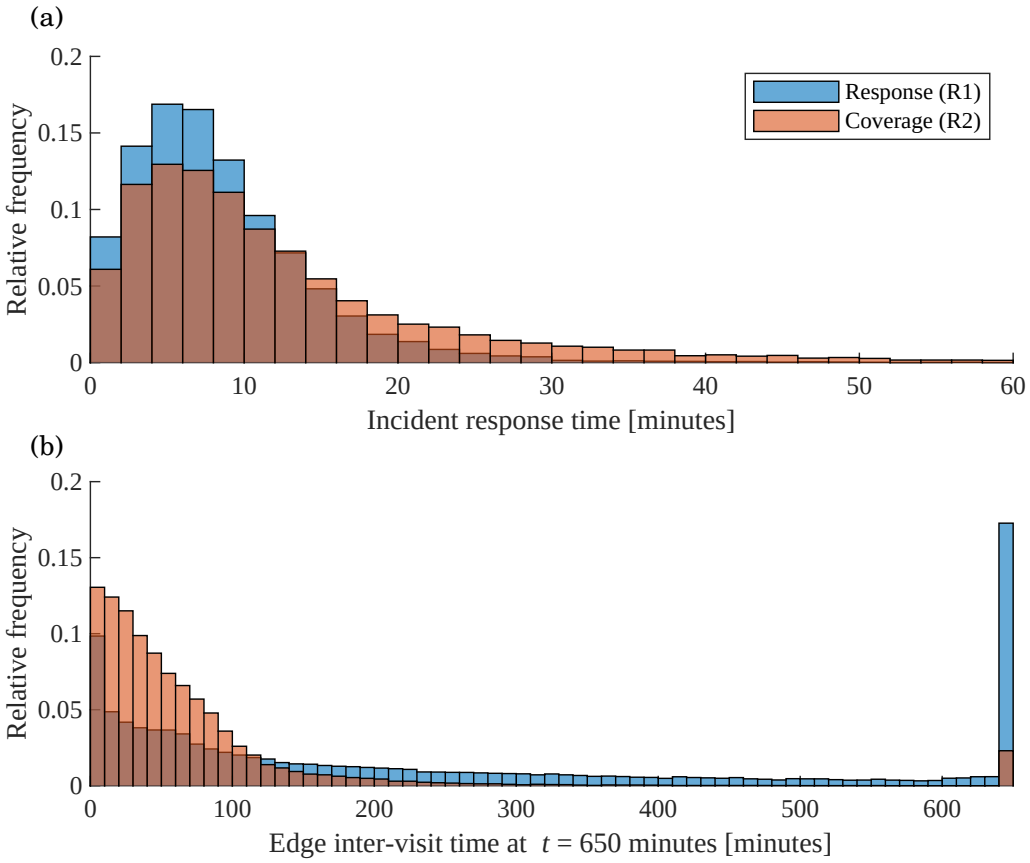



FIGURE 4.14. (a) Ensemble incident response times and (b) edge inter-visit time distributions at $t = 650$ minutes under regimes R1 and R2. The incident response mechanism (BWNN) is identical under both regimes and thus the response time distributions have a similar profile. Under regime R1, the FTOs frequently visit edges close to response nodes, thus, there are an anomalously large number of inter-visit times from $0-10$ minutes, and a large proportion of edges that are not visited at all.

time window $150 \leq t_w \leq 650$ minutes, such that,

$$\text{TAIRT} = \frac{1}{|t_w|}\sum_{t_w}\text{AIRT}(t_w) \tag{4.13}$$

and

$$\text{TAIVT} = \frac{1}{|t_w|}\sum_{t_w}\text{AIVT}(t_w). \tag{4.14}$$

Here, $|t_w|$ denotes the number of time steps in the time-window (500 for $\Delta t = 1$ minute). Similarly, we consider the APV at the end of the time-window (APVW),

$$\text{APVW} = \text{APV}(650 \text{ minutes}); \tag{4.15}$$

TABLE 4.2. TAIRT, TAIVT, and APVW achieved by an FTO fleet of 234 vehicles. The difference row shows the difference between regimes R2 and R1. Overall, regime R2 reduces the TAIVT by 106.46 minutes and increases the APVW by 15%, whereas, the TAIRT is only increased by 3.93 minutes.

| Regime | TAIRT [minutes] | TAIVT [minutes] | APVW [%] |
|---|---|---|---|
| Response (R1) | 8.52 | 186.74 | 83 |
| Coverage (R2) | 12.45 | 80.28 | 98 |
| Difference | 3.93 | -106.46 | 15 |
| Random (R3) | 13.41 | 135.16 | 93 |

essentially, this is the percentage of edges visited by at least one FTO before the return period.

### 4.5.2 Simulation Results

The TAIRT, TAIVT, and APVW for each regime are given in Table 4.2. Overall, regime R2 decreases the TAIVT by 106.46 minutes and the APVW by 15% compared with regime R1, whereas the TAIRT is only increased by 3.93 minutes. The incident response time and edge inter-visit time distributions at $t = 650$ minutes across the ensemble are provided in Figure 4.14.

The incident response mechanism (BWNN) is identical under all patrol regimes and thus the incident response time distributions have a similar profile for both regimes R1 and R2. In contrast, regime R1's inter-visit time distribution exhibits a relatively heavy tail compared against regime R2, and 17% of edges are not visited at all during the patrol. In addition, an anomalously large number of inter-visit times from $0 - 10$ minutes are exhibited under regime R1, because those edges connecting to response nodes are visited more regularly.

### 4.5.3 Varying Simulation Time Steps

Figure 4.15 shows the TAIRT and TAIVT achieved by regimes R1 and R2 for varying simulation time steps. For each time step, the metrics are respectively computed via Equations 4.13 and 4.14 over an ensemble of 20 simulations under each regime. Overall, neither metric is significantly affected by the simulation time step. Note that all other results are computed with time step $\Delta t = 1$ minutes as discussed in Section 4.4.1.

### 4.5.4 Varying Fleet Sizes

Figure 4.16 shows the TAIRT, TAIVT, and APVW achieved by each regime for varying fleet sizes (averaged across 30 simulations per fleet size). When the fleet size is small, the number of incidents overwhelms the fleet and a queue of incidents for each FTO to attend builds. Consequently, the FTO patrol is identical under all regimes; each FTO attends and clears an incident, and then attends the next and so on. As a result, all regimes achieve an asymptotically large TAIRT, and small TAIVT and APVW. At the other extreme, for large fleet sizes, the number of FTOs

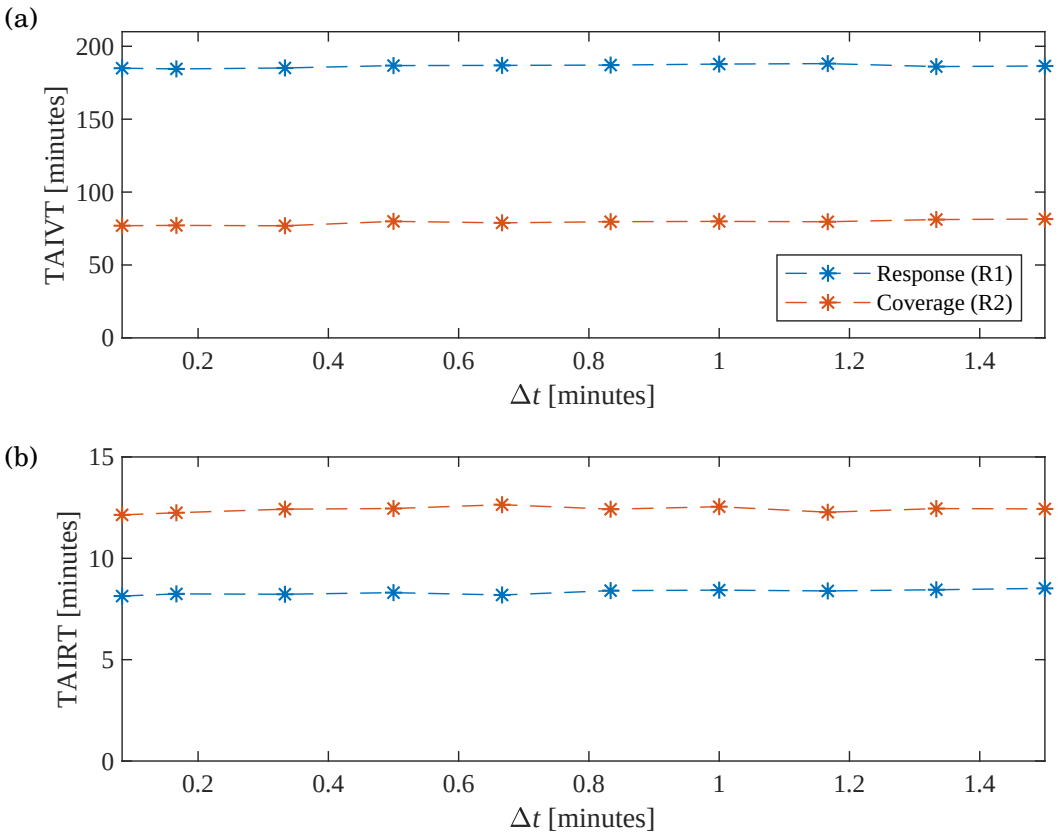



FIGURE 4.15. (a) TAIRT and (b) TAIVT achieved under regimes R1 and R2 for varying time steps between 5 seconds and 1.5 minutes. Overall, neither metric is significantly affected by refining the simulation time step.

approximates the number of nodes in the graph (1,108). In this case, the fleet is equally well distributed under either regime and the metrics tend to an asymptotic value. The key result shown in Table 4.2 is reproduced for all realistic fleet sizes — the FTO fleet can be deployed to efficiently collect asset data across the SRN, while only increasing the fleet's incident response time by a few minutes.

## 4.6 Discussion

The TAIRT achieved by regime R2 is only a 0.96 minute improvement on regime R3 (random) — this result is unsurprising as neither regime tries to position the fleet for fast incident response. However, regime R2 still achieves a TAIRT well under HE's current average incident response time of 17 minutes, and only 2.45 minutes over their target response time of 10 minutes [128].

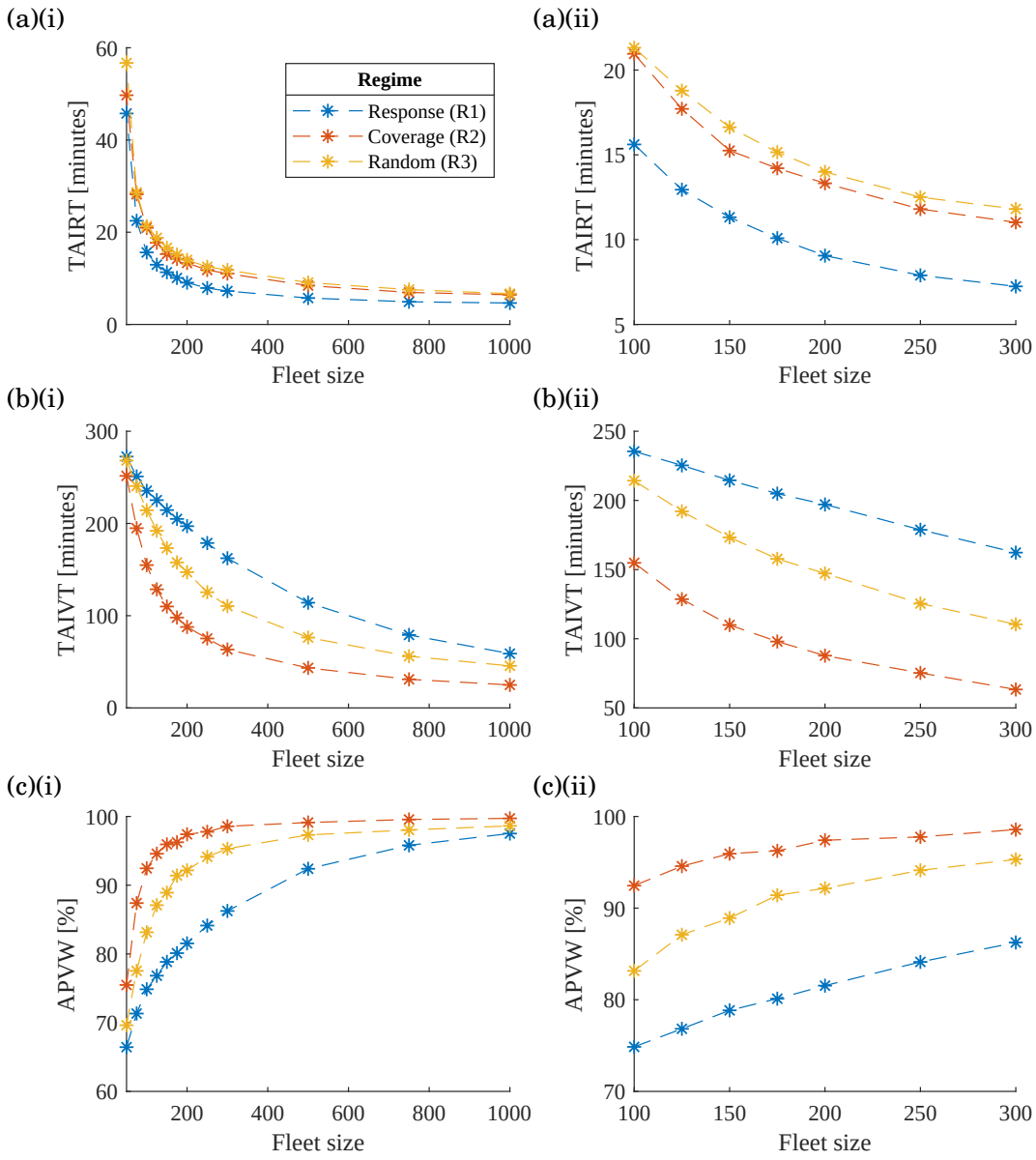



FIGURE 4.16. Panels (a) to (c) respectively show the TAIRT, TAIVT, and APVW for varying fleet sizes. The (i) panels show each metric across a wide range of fleet sizes whereas the (ii) panels are cropped to show more realistic fleet sizes.

Regime R1 reduces the TAIRT by 3.93 minutes (compared against regime R2) but only minimises the inter-visit times of edges close to response nodes or along the shortest path to an incident. Consequently, the TAIVT is significantly larger under regime R1, compared with regimes R2 and R3. Furthermore, on average, 17% of edges are not visited at all under regime R1, whereas only 2% of edges are not visited under regime R2. The key result, that regime R2 significantly reduces the TAIVT and APVW but only increases the TAIRT by a few minutes, is reproduced for all realistic fleet sizes (see Figure 4.16), and thus, our simulations show the potential to use TO fleets for asset management.

In this work, we have considered a base-case regime where each FTO chooses an edge at random (R3). Of course, there are other regimes that may result in longer incident response times and reduced coverage — every FTO remaining at their outstation, for example. However, regime R3 employs the same node-to-node routing methodology as that in regimes R1 and R2, and thus seems an appropriate base case.

The simulation as currently presented has a number of simplifications, and future work should refine the simulation to further reflect realistic TO patrols. For example, the FTOs are assumed to drive with a constant speed, however, it is likely that the TO's speed will be affected by traffic conditions on the highway. Therefore, historical traffic data should be built into future simulations to reflect realistic driving conditions. In addition, the FTOs make an instantaneous routing decision as they arrive at a node and do not change their speed. While this feature of the simulation is relatively realistic for those TOs who chose to remain on the same highway, those TOs who chose to merge onto a different highway at a junction may reduce their speed on a slip road or even come to a complete stop at a signalised junction (with traffic lights).

Presently, we model the number of incidents on an edge as being proportional to its driven distance, with a constant of proportionality that is common to the entire network. However, an improved incident model might be calibrated with historical data concerning the number of incidents attended by TOs on each highway section. While public road safety data sets exist that record the number of vehicle crashes and collisions across the SRN [58], TOs respond to a number of other types of incidents (breakdowns and removing abandoned vehicles, for example). Therefore, efforts should be made to liaise with highway agencies to determine the number of incidents attended by TOs at a section-by-section level (or via the NTIS for the HE TO fleet on the SRN considered in this chapter).

Our proposed concepts of operation assume that the SRN and FTO vehicles are equipped with the necessary communications infrastructure for the FTOs to patrol under each regime. When an FTO makes a routing decision (under either regime R1 or R2) they require real-time data; regime R2 requires the edge inter-visit times, and regime R1 requires knowledge of the nearby FTOs and nodes around each FTO. Through dialogue with HE, it is understood that each RCC already captures live telemetry data for HE TO vehicles via an installed GPS-enabled device (HE, personal communication). Therefore, the data required by either regime can be computed in

real-time (the inter-visit times would be updated as a TO drives along a highway section) with the current technology.

To implement an operational system, an agency must decide how to compute and communicate the navigation instructions to each TO. This may be achieved by one of two ways:

1. the routing decision for each TO is computed in the RCC and then communicated back to the TO, either by radio or a connected (vehicle to RCC) navigation system; or,

2. the required data is sent back to the TO vehicle and a routing decision is computed by a small device installed in the vehicle.

There are a number of other possible operational constraints on the TO patrols that are not considered in our simulations, such as maximum mileage constraints, for example. In addition, the FTOs in our simulations are all assumed to patrol under the same regime. In reality, TOs may have varying roles; for example, only some of the RWS Weginspecteurs may compile official incident reports. Therefore, future work might incorporate non-homogeneous FTO fleets, using different patrol regimes.

Note that, regime R1 is designed for fast response to incidents that the RCC has been alerted to (e.g., via CCTV cameras), which then assigns the nearest TO. However, TOs patrolling under regime R2 may provide a faster response to minor incidents that are unknown to the RCC (e.g., in a CCTV blind spot or not reported by the public), which are not modelled in this paper. Such incidents are only discovered when TOs drive past them — an event promoted by maximising the fleet's coverage for TAM. Therefore, future simulations should consider mixtures of known and unknown incidents and whether mixtures of different patrol regimes result in reduced incident response times overall.

### 4.6.1 Method Justification

The work presented in this chapter is motivated by a highway agency's (who currently rely on annual survey data) requirement for regular (e.g., daily) asset data. This need is demonstrated by the ongoing conversations between HE and national coach services (each coach is equipped with a dash cam, see page 58), and personal communication with HE. Our research considers a data source already available to, and under the jurisdiction of an agency (i.e., TO dash cams), which is thus a feasible future asset monitoring capability. Moreover, our engagement with and inputs from HE (e.g., requested daily asset capture, TO rest patterns, and outstation placement) validate our rationale behind our research direction and the proof-of-concept work undertaken in this chapter.

The simulation incorporates the operational details of the HE TO patrol wherever possible; the day-night shift pattern and 50-50 rest-patrol split, for example. However, as described in Section 4.6 we make a number of assumptions. In the absence of granular traffic (e.g., intra-junction, collected every minute) data, we assume a constant TO speed. In addition, in the

absence of incident TTC distribution data (i.e., only summary statistics are published), we assume a constant incident TTC. However, these simulation parameters do not affect the patrol regimes, and may thus be viewed as a separate module of the simulation to be improved in future iterations.

The response and coverage patrol regimes must outperform a base case where TOs route at random. Ideally, we would replace this base case with breadcrumb GPS data collected from each TO vehicle. In this case, we may then compute the true current response and coverage metrics and inspect the gains or losses introduced by our regimes. However, due to sensitivity and privacy issues (we may have computed TO driving speeds, or lunch break times, for example) we were unable to gain access to the TO GPS data. Therefore, in this initial work, we compared against the random base case, and the current and target incident responses times published by HE (respectively 17 and 10 minutes, see page 129).

In Section 4.5, we consider averaged metrics (TAIRT, TAIVT, and APVW); that is, the response time and coverage distribution means obtained by each regime. This approach is valid as the distributions have a similar profile under each regime, see Figure 4.14.

## 4.7 Conclusion

In this chapter, patrol routing strategies are proposed and simulated to consider whether a fleet of traffic officers (TOs) may be used for TAM alongside their primary role of incident management. A case study of the Highways England (HE) TO fleet on the strategic road network (SRN) in England, UK, is considered to investigate, as a proof-of-concept, whether data from dash cams installed in TO vehicles might be used to frequently capture asset data across an entire highway network.

Junction-to-junction links in the HATRIS data set were used to build a graphical model of the SRN. A simulator was built to deploy the HE TO fleet who respond to dynamically generated incidents. Within the simulation, TOs patrol under one of three regimes that influences their routing decisions at each node (junction). Three regimes were considered: response, coverage, and random. The first aims to minimise the fleet's incident response time and acts as a best case to compare the other regimes' incident response times against. The second aims to maximise the fleet's coverage for asset management. The third regime, where TOs make random routing decisions, acts as a base case.

To determine the feasibility of the proposed patrol routes, the incident response times and the edge inter-visit times were computed. Overall, our simulations showed that the TOs (with varying fleet sizes) can monitor assets across the SRN while only increasing the incident response time by a few minutes.

We have thus demonstrated that TOs can be used for frequent asset data capture, alongside their primary role of incident response. To employ our proposed patrol routing strategies in

operation, the increase in incident response time must be weighed up by decision makers. That said, in 2019, only 18% of accidents on major SRN highways (i.e., those roads with an 'M' prefix) were categorised as 'fatal' or 'serious' [58], and thus, the majority of incidents that HE TOs attend (that also include breakdowns and other minor issues) are not time critical. Highway agencies also need to consider the costs of the required sensor and communications infrastructure. However, note that for the HE TO fleet considered in this paper, the required sensing capability (the dash cams) and much of the communication infrastructure (radio and telemetry data sent to a regional control centre) is already in place.

CHAPTER 5

# CONCLUSION

In this thesis, three case studies (one per chapter) on intelligent transport technologies deployed for transportation asset management (TAM) are examined that aimed to address the following research questions (RQs) initially defined on page 7.

RQ1. *Can extra value be derived from pre-existing fixed sensors; that is, can they be used for a secondary purpose in addition to their primary function?*

RQ2. *To what degree can TAM be automated; to what degree is there scope for decision-support systems to assist agency analysts?*

RQ3. *Are there ad-hoc sensing capabilities available to an agency, so that they may regularly capture asset data, rather than relying on scheduled remote survey data?*

In Chapter 2, structural health monitoring (SHM) systems, that capture data from sensors installed onto the structural components of civil structures, were initially investigated to address research question RQ1. While these fixed sensor deployments are widely used, they usually have a single specific function (i.e., monitoring the health of structural components), and rarely have a secondary purpose that extends beyond that for which they were designed for at the point of installation. There is, therefore, great potential to add value to these existing SHM setups.

Data from a single accelerometer along with toll barrier data captured during a previous trial [94] on the Clifton Suspension Bridge (CSB) was considered to develop and test a method to estimate the number of vehicles that travel across the bridge: thus, providing an alternative to traditional systems (e.g., inductance detector loops) that can be costly to install and maintain. Initially, our system achieved a 74% prediction accuracy, however, it was restricted to low vehicle counts due to inaccuracies in the ground-truth vehicle counts provided by toll barriers. Therefore,

to further develop and test the system, we conducted a second trial in which accurate ground-truth vehicle counts were collected by human observers.

The higher vehicle counts collected in the second trial motivated us to consider a regression-based framework. As such, we explored the state-of-the-art in time series regression (TSR), and investigated a leading generalised TSR method, ROCKET (random convolutional kernel transform). In addition, we proposed our own relatively crude spectrogram thresholding-based method, whereby those powers greater than a tuned threshold are summed. Over a set of withheld two-minute windows, our proposed method achieved an encouraging 1.36 vehicle mean absolute error (MAE), which outperforms the ROCKET method. Furthermore, as we increased the window length, and thus improved the accuracy of the ground-truth counts (by reducing the structural confusion at the window boundaries), we achieved a mean absolute percentage error (MAPE) of 6.29%.

The work presented in Chapter 2 clearly answers RQ1 — our proposed systems adds value to data collected from a pre-installed, existing sensor (i.e., an accelerometer). The methods we have developed and tested allow bridge maintenance teams to monitor the structural health of a bridge, and estimate the number of vehicles that travel across it, from the same data source. The encouraging performance of our system (1.36 vehicle MAE) quantitatively establishes that we have answered RQ1 — SHM data can, very effectively, be used for a secondary purpose.

In Chapter 3, we developed and tested a computer vision-based decision support system designed to assist an analyst in roadside asset inventory. Typically, inventory updating is performed via manual inspection of survey imagery captured by a specialised vehicle. Our system automatically verifies those assets that are correctly recorded in the inventory, and thus identifies those assets that require updating, reducing the manual workload of the analyst. In developing our system, we address research question RQ2.

Our proposed asset monitoring capability integrates multiple data sources used by the analyst; namely, the inventory, aerial imagery, and survey data (imagery and position data) collected by a camera and GPS-enabled inertial measurement unit (IMU) installed on a vehicle. The system is built upon two distinct computer vision contributions. Firstly, the assets are projected (as a function of their positional and physical description in the inventory) into the survey imagery via a camera calibration process. Then, the localised area of the survey imagery (that contains the asset) is classified by a state-of-the-art convolutional neural network (CNN).

To prove the feasibility of our decision support system, we considered three types of roadside asset: traffic signs, reference marker posts, and matrix signs installed along an eight-kilometre section of the A27 highway on the south coast of England. Overall, 91% of assets in a withheld test set were automatically verified by our system, thus (potentially) significantly reducing the workload of an analyst.

In addition, we presented a prototype tool that enables the user to easily use our computer vision-based methods for inventory updating, and suggested improvements to mature the proto-

type towards a fully operational system; namely, an additional interface that allows the user to update the inventory directly from the tool. Further, we explored how we might make a future decision support system more robust. Specifically, we investigated how visual simultaneous localisation and mapping (vSLAM) might refine, or replace missing IMU position data required by our system so it might operate in GPS-denied environments.

RQ2 asks to what to degree can TAM be automated. The tool proposed in Chapter 3 is semi-automated; that is, it is designed to operate alongside the analyst rather than replace them completely. As discussed in Chapter 3 (see Section 3.5.1, Page 80), there are a number of obstacles to overcome to realise a fully automated TAM system (i.e., fully automated asset condition assessment). Nonetheless, our system (it's performance on a withheld test set of assets) demonstrates that semi-automated, decision-support systems can be readily developed and integrated into TAM processes.

In Chapter 4, we addressed research question RQ3 by considering, as a proof of concept, whether highway traffic officers (TOs), who primarily respond to traffic incidents, might also capture asset data while on patrol. Having considered the international scene, and the TO fleets available in each country, we considered a case study on the Highways England (HE) TO fleet — a dedicated incident response capability (i.e., not a police sub-unit) that patrols on the strategic road network (SRN) in England.

To determine the feasibility of our proposed TOs, a simulator was built with publically available data sets on the SRN and the HE TO fleet. Within the simulation, the TOs respond to dynamically generated incidents and patrol under one of two distinct regimes: one that tries to minimise the fleet's incident response time, and one that tries to maximise the fleet's coverage (for asset management). An additional third regime, where the TOs randomly patrol, acts as a base case. Overall, our simulations showed that the TOs (with varying fleet sizes) can successfully monitor assets across the SRN while only increasing the incident response time by a few minutes.

The results from our simulation provide the following answer for RQ3: Those agencies that deploy a dedicated TO fleet (e.g., HE and Rijkswaterstaat in the Netherlands), might use TO vehicles (equipped with a dashboard camera (dash cam)), to capture network-wide asset data. Although our work is intended as a proof-of-concept, we have shown that the HE TO fleet can capture asset data ad-hoc (i.e., the TOs collect asset data while patrolling as normal, albeit under a maximal coverage regime) alongside their primary function of incident response. Thus, our proposed schemes could potentially replace the annual survey data with near real-time data across an entire highway network.

Each method presented in this thesis is, to some degree, tuned or optimised for the case study considered in each chapter. As such, future systems, that use our methods may need to relearn some parameter values or system configurations. For example:

- agency analysts that wish to use our computer vision-based support system for inventory updating, with different survey data and inventories than those considered here, may need

to recalibrate the camera parameters or retrain the CNN (with other asset types);

- bridge operators using our spectrogram thresholding-based vehicle counting method, on a bridge other than the CSB, may need to retune the method (i.e., the threshold $\delta$, and the low and high-pass frequency bands, see page 42) so that it may capture the important frequency components of that bridge's structural response under loading; and

- highway agencies that wish to deploy a TO fleet, based on our work in Chapter 4, for asset data capture, will have to weigh up its impact on incident response times — which is likely to be different for each highway network with potentially varying topology, fleet size, and incident rate.

Nonetheless, we have presented a number of systems that demonstrate how applications of data science, computer vision, and mathematical modelling can enrich the TAM process. We have presented methodologies which both process data already available to a highway agency (i.e., from fixed sensors and remote surveys), and proposed novel ad-hoc data capture schemes. In each case, we have identified directions to further develop our ideas towards future deployable and operational capabilities.

Each chapter considered one of the three sensing modes defined in Section 1.2: fixed, remote survey, and ad-hoc. In doing so, we have examined case studies on intelligent transportation technologies deployed right across the breadth of TAM. Furthermore, all of the methods developed in this thesis exploit simple, light-weight sensor setups; namely, a single accelerometer (in Chapter 2), a single forward-facing camera and GPS-enabled IMU (in Chapter 3), and the dashboard cameras and telemetry installed in TO vehicles (in Chapter 4). As such, we leave our work open to researchers to further develop and mature our methods, without the need to procure expensive or specialised equipment.

## 5.1 Results Highlights

We now summarise some key research highlights from the thesis.

- We developed a method to estimate the number of vehicles that travel across a bridge from data captured by a single accelerometer installed onto a structural component of the bridge. Our initial curve fitting method, that uses a linear superposition assumption in conjunction with the Akiake information criterion (see Figure 2.9) achieved a 74% prediction accuracy, on a case study on the Clifton Suspension Bridge (CSB), see Table 2.3, page 32.

- We conducted a second trial on the CSB to collect more accelerometer data with accurate ground-truth data. This involved liaising with the CSB trust and developing a bespoke mobile application, see Figure 2.11, page 35.

- We proposed a novel spectrogram thresholding-based procedure that outperformed a leading TSR method (ROCKET), see Table 2.4 for detailed results, page 46.
- We showed that our regression-based methods continued to work well for longer time windows. However, the method overfitted the training set when we extended the windows to 5 minutes (thus reducing the training size), see Figure 2.19, page 48.
- We applied computer vision techniques in order to develop a decision support tool designed to assist an agency analyst in roadside asset inventory. Overall, on a test set of assets, our system automatically verified 91% of the inventory, which would otherwise need to be verified by manual inspection.
- By collecting control points and minimising their reprojection error (see Equation 3.6), we calibrated a forward-facing camera installed on a survey vehicle. The intrinsic and extrinsic camera parameters are given in Table 3.1, see page 68.
- We used transfer learning (see Figure 3.10, page 72) to retrain the ResNet50 architecture for asset classification. On a validation set of assets, the CNN achieved an impressive 98% classification accuracy. We thus demonstrated how one may achieve high classification performance with limited training data (i.e., an insufficient amount to train a CNN from scratch). However, the CNN is not a perfect classifier, and some assets in the withheld test set were incorrectly identified for further manual inspection, see Table 3.4 and Figure 3.15, pages 77 and pages 80.
- We augmented the ORB-SLAM system to overcome two problems when using traditional vSLAM systems on the highway. However, the position of the survey vehicle (computed directly from the survey imagery) is computed to an arbitrary scale and suffers from drift, see Figure 3.22, page 91.
- We considered, as a proof-of-concept, whether TOs might be used by highway agencies to collect asset data across an entire highway network, in addition to their primary function of incident response.
- Through processing the HATRIS links (see Table 4.1, page 109), we developed a graphical representation of the SRN, whose nodes and directed edges respectively represent the highway junctions and carriageway. See Figure 4.4, page 112.
- Through building a simulator we deployed the HE TO fleet on the SRN graph, who respond to dynamically generated incidents, while patrolling under one of two distinct regimes. The first regime aimed to minimise incident response time, and we investigated its emergent properties for different parameter ($\delta_n$ and $\delta_f$) values, see Figures 4.8, 4.9, and 4.10, pages 120, 121, and 122. The second regime aimed to maximise network coverage for asset data capture.

- We showed that the fleet can successfully be used to capture asset data, without significantly reducing incident response times. Detailed results are given in Table 4.2, page 128. Furthermore, we reproduced this result for a number of different fleet sizes, see Figure 4.16, page 130.

## 5.2 Future Work

We now propose some potential directions for future research and consider the impact of our work.

The work presented in Chapters 3 and 4 are complimentary in that they improve the same TAM practise; namely, updating an inventory of assets installed across a highway network — currently an expensive and time-consuming manual task, that considers data collected on an annual basis.

A potential operational capability, based on our work, might involve TOs (patrolling under our coverage regime) collecting network-wide asset imagery data, which is then analysed by our computer vision-based system — thus, maintaining an up to date inventory of all roadside assets (under an agencies jurisdiction) at all times.

It seems the next step, to realise this capability, is to liaise with highway agencies in order to develop a real-world trial, probably on a small section of highway. The trial, in the first instance, should focus on the infrastructure required by our system; that is, where and how the data is collected and analysed. There are seemingly three available options for data analysis — the data may be analysed either:

1. in the TO vehicle;
2. at the roadside; or
3. at a central facility (e.g., an agency head office).

Option 1 would require a small computing device within the TO vehicle; data would then be sent directly from the dash cam to the device for analysis. The analysis would be fully automated, so that the TOs may continue to patrol as usual. Alternatively, miniaturized computers (e.g., a Raspberry Pi [213]) can be equipped with a camera, and thus might replace the dash cam completely. Then, our computer vision-based system could be run in real-time during the TO patrol.

Option 2 would require installation of additional highway infrastructure, whereby a TO vehicle may wirelessly upload the collected asset data (via a connected device in the vehicle) to a computer installed on the roadside (i.e., edge computing technology [267]), where it is then automatically analysed. Option 3 would instead transmit the data from the roadside to a central computer in an agency facility, accessible to an agency analyst and thus the data maybe analysed

manually or automatically. The results of the automatic analysis performed by Option 2 may be provided to an analyst via transmission from the roadside infrastructure to a central computer, whereas the results from Option 1 must first be uploaded to the roadside infrastructure.

Once an option has been decided on and successfully implemented, then the work may focus on deploying TOs patrolling under our coverage regime (i.e., for maximum network coverage). This is likely to require careful consideration of the impact on incident response times (IRTs) — some agencies may not tolerate any increase in IRTs at all. However, a hybrid approach, where TO fleets patrol under a mixture of regimes, might provide a more flexible operational framework with which to consider the IRT-network coverage trade-off.

Finally, we provide a bulleted list for some alternative potential avenues for future research.

- Investigate whether methods proposed in each chapter can be combined. For example: potentially, the TO vehicles could also be equipped with an accelerometer, to capture network-wide pavement condition data. In this case, a system that might automatically detect potholes (i.e., from TO vehicle jerks) and pavement roughness directly from the accelerometer data, may well use some of the time series methods presented in Chapter 2.

- Embed our vehicle count estimation system within existing SHM software solutions (e.g., the dashboard developed by Gunner et al. see Figure 2.4, page 22). Further, future research should consider multi-sensor wireless sensor network-based SHM systems (i.e., where multiple accelerometers form nodes in the network), and investigate to what degree our methods work on larger, potentially multi-lane (more than two) bridges.

- Integrate our computer vision-based decision support system into the existing AVIS (Asset Visual Information System, see page 56) used by HE. Of course, different sets of survey imagery (collected by different survey vehicles) may require different camera parameters, which may be estimated by the camera calibration process described in Section 3.3.1, or alternatively, via access to the survey vehicle where the camera's configuration and focal lengths may be measured.

# REFERENCES


[1] M. A. ABDELWAHAB, *Accurate vehicle counting approach based on deep neural networks*, in Proceedings of 2019 International Conference on Innovative Trends in Computer Engineering (ITCE), 2019, pp. 1–5.

[2] M. ABDULKAREM, K. SAMSUDIN, F. Z. ROKHANI, AND M. F. A. RASID, *Wireless sensor network for structural health monitoring: A contemporary review of technologies, challenges, and future direction*, Structural Health Monitoring, 19 (2020), pp. 693–735.

[3] K. ABDULRAHIM AND R. A. SALAM, *Traffic surveillance: A review of vision based vehicle detection, recognition and tracking*, International Journal of Applied Engineering Research, 11 (2016), pp. 713–726.

[4] N. AGMON, D. URIELI, AND P. STONE, *Multiagent patrol generalized to complex environmental conditions*, in Proceedings of the 25th AAAI Conference on Artificial Intelligence, 2011, pp. 1090–1095.

[5] C. AI AND Y. J. TSAI, *An automated sign retroreflectivity condition evaluation methodology using mobile LIDAR and computer vision*, Transportation Research Part C: Emerging Technologies, 63 (2016), pp. 96 – 113.

[6] H. AKAIKE, *Information Theory and an Extension of the Maximum Likelihood Principle*, Springer New York, New York, NY, 1998, pp. 199–213.

[7] S. S. M. ALI, B. GEORGE, L. VANAJAKSHI, V. JAYASHANKAR, AND V. J. KUMAR, *A multiple loop vehicle detection system for heterogeneous and laneless traffic*, in Proceedings of 2011 IEEE International Instrumentation and Measurement Technology Conference, 2011, pp. 1–5.

[8] B. ALLISON, *The structures management information system (SMIS) database*, in Bridge Management 5: Inspection, Maintenance, Assessment and Repair, Thomas Telford Publishing, 2015, pp. 83–89.

[9] AMERICAN ASSOCIATION OF STATE HIGHWAY AND TRANSPORTATION OFFICIALS, *AASHTO transportation asset management guide: A focus on implementation*, AASHTO, 2011.

[10] AMERICAN ASSOCIATION OF STATE TROOPERS, *Highway patrol/state police websites*, 2020.
https://www.statetroopers.org/useful-links/highway-patrol-state-police-websites
Accessed: 01.03.2021.

[11] AN GARDA SÍOCHÁNA, *Roads policing*, 2020.
https://www.garda.ie/en/roads-policing
Accessed: 22.03.2021.

[12] APACHE KAFKA, *Kafka*, 2021.
https://kafka.apache.org/
Accessed: 19.07.2021.

[13] A. A. ARIYO, A. O. ADEWUMI, AND C. K. AYO, *Stock price prediction using the ARIMA model*, in Proceeding of 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, 2014, pp. 106–112.

[14] P. ARNOUL, M. VIALA, J. P. GUERIN, AND M. MERGY, *Traffic signs localisation for highways inventory from a video camera on board a moving collection van*, in Proceedings of Conference on Intelligent Vehicles, 1996, pp. 141–146.

[15] E. ARTIN, *The Gamma Function*, Courier Dover Publications, Mineola, NY, USA, 2015.

[16] ARUP, *WP 1-086(ppro 4/45/12) — application of remote survey data for geotechnical asset condition and performance*, 2017.
https://s3.eu-west-2.amazonaws.com/assets.highwaysengland.co.uk/specialist-information/knowledge-compendium/2016-17/Application+of+Remote+Survey+Data+for+Geotechnical+Asset+Condition+and+peformance.pdf
Accessed: 10.05.2021.

[17] AUSTROADS, *Data standard for road management and investment in Australia and New Zealand Version 2*, 2018.
https://austroads.com.au/publications/asset-management/ap-t334-18/media/AP-T334-18_Austroads_Data_Standard_V2.pdf
Accessed: 20.09.2021.

[18] ——, *Guide to traffic management part 10: Traffic control and communication devices*, 2019.
https://austroads.com.au/__data/assets/pdf_file/0026/342836/AGTM10-19-Guide-to-Traffic-Management-Part-10-Traffic-Control-and-Communication-Devices.pdf
Accessed: 20.09.2021.

[19] AUTO EXPRESS TEAM, *What do Highways England traffic officers do all day?*, 2019.
https://www.autoexpress.co.uk/107988/what-do-highways-england-traffic-officers-do-all-day-pictures
Accessed: 22.03.2021.

[20] AUTOCAR, *Highways traffic officers — meet the people behind the nicknames*, 2020.
https://www.autocar.co.uk/car-news/new-cars/highways-traffic-officers-meet-people-behind-nicknames
Accessed: 26.01.2021.

[21] R. AZZAM, T. TAHA, S. HUANG, AND Y. ZWEIRI, *Feature-based visual simultaneous localization and mapping: A survey*, SN Applied Sciences, 2 (2020), pp. 1–24.

[22] V. BALALI, A. ASHOURI RAD, AND M. GOLPARVAR-FARD, *Detection, classification, and mapping of U.S. traffic signs using Google street view images for roadway inventory management*, Visualization in Engineering, 3 (2015), p. 15.

[23] O. BARNDORFF-NIELSEN AND G. SCHOU, *On the parametrization of autoregressive models by partial autocorrelations*, Journal of Multivariate Analysis, 3 (1973), pp. 408–419.

[24] V. BATANOVIĆ, D. PETROVIĆ, AND R. PETROVIĆ, *Fuzzy logic based algorithms for maximum covering location problems*, Information Sciences, 179 (2009), pp. 120–129.

[25] H. BAY, A. ESS, T. TUYTELAARS, AND L. VAN GOOL, *Speeded-up robust features (SURF)*, Computer Vision and Image Understanding, 110 (2008), pp. 346–359.

[26] BBC, *Smart motorways are dangerous, says Yorkshire police chief*, 2021.
https://www.bbc.co.uk/news/uk-england-south-yorkshire-55782301
Accessed: 10.06.2021.

[27] B. BERBERIAN, J.-C. SARRAZIN, P. LE BLAYE, AND P. HAGGARD, *Automation technology and sense of control: A window on human agency*, PLOS ONE, 7 (2012), pp. 1–6.

[28] J. BIAN, Z. LI, N. WANG, H. ZHAN, C. SHEN, M.-M. CHENG, AND I. REID, *Unsupervised scale-consistent depth and ego-motion learning from monocular video*, in Proceedings of Advances in Neural Information Processing Systems 32, 2019, pp. 35–45.

[29] G. A. BODVARSSON AND S. T. MUENCH, *Effects of loop detector installation on the Portland cement concrete pavement lifespan: Case study on I-5*, tech. rep., Washington Department of Transportation Office of Research and Library Services, 2010.

[30] M. BOGATOV, *The motorway life in the fast lane, 3 the need for speed*, 2016.
https://www.youtube.com/watch?v=1_vu0WSqN-Y
Accessed: 22.03.2021.

[31] L. BOTTOU, *Large-scale machine learning with stochastic gradient descent*, in Proceedings of COMPSTAT'2010, 2010, pp. 177–186.

[32] G. E. BOX, G. M. JENKINS, AND G. C. REINSEL, *Time Series Analysis, Forecasting and Control*, Prentice Hall, 1994.

[33] J. BROWNJOHN, M. BOCCIOLONE, A. CURAMI, M. FALCO, AND A. ZASSO, *Humber bridge full-scale measurement campaigns 1990–1991*, Journal of Wind Engineering and Industrial Aerodynamics, 52 (1994), pp. 185–218.

[34] J. M. BROWNJOHN, *Structural health monitoring of civil infrastructure*, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 365 (2007), pp. 589–622.

[35] J. M. BROWNJOHN, T. PAN, A. MITA, AND K. CHOW, *Dynamic and static response of Republic Plaza*, Journal of the Institution of Engineers, 38 (1998), pp. 35–41.

[36] C. CALIENDO, M. GUIDA, AND A. PARISI, *A crash-prediction model for multi-lane roads*, Accident Analysis and Prevention, 39 (2007), pp. 657–670.

[37] CAMBRIDGE DICTIONARY, *Infrastructure*, 2021.
`https://dictionary.cambridge.org/dictionary/english/infrastructure`
Accessed: 13.09.2021.

[38] P. E. CARNELLI, M. SOORIYABANDARA, AND R. E. WILSON, *Large-scale VANET simulations and performance analysis using real taxi trace and city map data*, in Proceedings of 2018 IEEE Vehicular Networking Conference (VNC), 2018, pp. 1–8.

[39] F. CARRIÓN-VIRAMONTES, J. LÓPEZ-LÓPEZ, J. QUINTANA-RODRÍGUEZ, AND A. LOZANO-GUZMÁN, *Nonlinear assessment of cable vibration in a stayed bridge*, Experimental Mechanics, 48 (2008), pp. 153–161.

[40] E. CETINIC, T. LIPIC, AND S. GRGIC, *Fine-tuning convolutional neural networks for fine art classification*, Expert Systems with Applications, 114 (2018), pp. 107–118.

[41] P. CHUJAI, N. KERDPRASOP, AND K. KERDPRASOP, *Time series analysis of household electric consumption with ARIMA and ARMA models*, in Proceedings of the International Multi-conference of Engineers and Computer Scientists, vol. 1, 2013, pp. 295–300.

[42] R. CHURCH AND C. REVELLE, *The maximal covering location problem*, Papers of the Regional Science Association, 32 (1974), pp. 101–118.

[43] D. CIREŞAN, U. MEIER, J. MASCI, AND J. SCHMIDHUBER, *Multi-column deep neural network for traffic sign classification*, Neural Networks, 32 (2012), pp. 333 – 338.

[44] J. CIVERA, A. J. DAVISON, AND J. M. MONTIEL, *Inverse depth parametrization for monocular SLAM*, IEEE Transactions on Robotics, 24 (2008), pp. 932–945.

[45] G. CLARKE AND J. W. WRIGHT, *Scheduling of vehicles from a central depot to a number of delivery points*, Operations Research, 12 (1964), pp. 568–581.

[46] S. COHEN AND Z. CHRISTOFOROU, *Travel time estimation between loop detectors and FCD: A compatibility study on the Lille network, France*, Transportation Research Procedia, 10 (2015), pp. 245–255.

[47] Y. M. COSTA, L. S. OLIVEIRA, AND C. N. SILLA JR, *An evaluation of convolutional neural networks for music classification using spectrograms*, Applied Soft Computing, 52 (2017), pp. 28–38.

[48] D. COYLE, J. MOORE, P. O. KRISTENSSON, P. FLETCHER, AND A. BLACKWELL, *I did that! Measuring users' experience of agency in their own actions*, in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2012, pp. 2025–2034.

[49] G. B. DANTZIG AND J. H. RAMSER, *The truck dispatching problem*, Management Science, 6 (1959), pp. 80–91.

[50] A. J. DAVISON, *Real-time simultaneous localisation and mapping with a single camera*, in Proceedings of the IEEE International Conference on Computer Vision, vol. 2, 2003, pp. 1403–1410.

[51] A. J. DAVISON, I. D. REID, N. D. MOLTON, AND O. STASSE, *MonoSLAM: Real-time single camera SLAM*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 29 (2007), pp. 1052–1067.

[52] M. Á. Á. DE LA CONCEPCIÓN, L. M. S. MORILLO, J. A. Á. GARCÍA, AND L. GONZÁLEZ-ABRIL, *Mobile activity recognition and fall detection system for elderly people using Ameva algorithm*, Pervasive and Mobile Computing, 34 (2017), pp. 3–13.

[53] A. DE LA ESCALERA, J. ARMINGOL, AND M. MATA, *Traffic sign recognition and analysis for intelligent vehicles*, Image and Vision Computing, 21 (2003), pp. 247–258.

[54] A. DEMPSTER, F. PETITJEAN, AND G. I. WEBB, *ROCKET: Exceptionally fast and accurate time series classification using random convolutional kernels*, Data Mining and Knowledge Discovery, 34 (2020), pp. 1454–1495.

[55] J. DENG, W. DONG, R. SOCHER, L.-J. LI, K. LI, AND L. FEI-FEI, *Imagenet: A large-scale hierarchical image database*, in Proceedings of 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.

[56] DEPARTMENT FOR TRANSPORT, *Traffic management act 2004*, 2004.
https://www.legislation.gov.uk/ukpga/2004/18/contents?pt1-pb2-l1g4
Accessed: 01.03.2021.

[57] ——, *Strategic road network statistics*, 2019.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/448276/strategic-road-network-statistics.pdf
Accessed: 29.10.2021.

[58] ——, *Road safety data*, 2020.
https://data.gov.uk/dataset/cb7ae6f0-4be6-4935-9277-47e5ce24a11f/road-safety-data
Accessed: 28.04.2021.

[59] ——, *A fresh start for the strategic road network*, 2021.
https://webarchive.nationalarchives.gov.uk/20120606131143/http:/assets.dft.gov.uk/publications/strategic-roads-network/strategic-road-network.pdf
Accessed: 10.06.2021.

[60] ——, *Road condition and maintenance data*, 2021.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/836188/road-conditions-technote.pdf
Accessed: 15.09.2021.

[61] ——, *Transport statistics Great Britain 2019*, 2021.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/870647/tsgb-2019.pdf
Accessed: 14.09.2021.

[62] M. DEWINTER, C. VANDEVIVER, T. V. BEKEN, AND F. WITLOX, *Analysing the police patrol routing problem: A review*, ISPRS International Journal of Geo-Information, 9 (2020), p. 157.

[63] DIRECTOR GENERAL OF THE ORDNANCE SURVEY UK, *A brief description of the national grid and reference system*, His Majesty's Stationery Office (HMSO), 1946.

[64] EBO VAN WEEL, *Autodrip nominated for intertraffic innovation award 2016*, 2016.
https://www.ebovanweel.com/en/autodrip-nominated-for-intertraffic-innovation-award-2016
Accessed: 30.06.2021.

[65] N. A. EL-SHERBENY, *Vehicle routing with time windows: An overview of exact, heuristic and metaheuristic methods*, Journal of King Saud University — Science, 22 (2010), pp. 123–131.

[66] ELECTRONIQUE CONTRÔLE MESURE, *Inductive loop*, 2019.
https://www.ecm-france.com/en/areas-of-activity/weigh-in-motion/inductive-loop/
Accessed: 25.08.2021.

[67] J. ERIKSSON, L. GIROD, B. HULL, R. NEWTON, S. MADDEN, AND H. BALAKRISHNAN, *The pothole patrol: Using a mobile sensor network for road surface monitoring*, in Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services, 2008, pp. 29–39.

[68] ESRI, *Resources for ArcMap*, 2021.
https://www.esri.com/en-us/arcgis/products/arcgis-desktop/resources
Accessed: 20.05.2021.

[69] EUROPEAN COMMISSION, *Common framework for a European life cycle based asset management approach for transport infrastructure networks*, 2017.
https://cordis.europa.eu/project/rcn/204966/factsheet/en
Accessed: 09.06.2021.

[70] C. R. FARRAR AND K. WORDEN, *An introduction to structural health monitoring*, Philosophical Transactions of the Royal Society A, 365 (2006), pp. 303–315.

[71] D. Ö. FARUK, *A hybrid neural network and ARIMA model for water quality time series prediction*, Engineering Applications of Artificial Intelligence, 23 (2010), pp. 586–594.

[72] FEDERAL HIGHWAY ADMINISTRATION, *Traffic statistics, methodology review: Alternative data sources*, 2014.
https://www.fhwa.dot.gov/policyinformation/pubs/vdstits2007/04.cfm
Accessed: 25.08.2021.

[73] M. A. FISCHLER AND R. C. BOLLES, *Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography*, Communications of the ACM, 24 (1981), pp. 381–395.

[74] FLEET NEWS, *Fraikin strikes fleet management deal with Highways England*, 2018.
https://www.fleetnews.co.uk/news/fleet-industry-news/2018/07/05/fraikin-strikes-fleet-management-deal-with-highways-england
Accessed: 30.06.2021.

[75] M. M. FLOOD, *The traveling-salesman problem*, Operations Research, 4 (1956), pp. 61–75.

[76] H. FU, M. GONG, C. WANG, K. BATMANGHELICH, AND D. TAO, *Deep ordinal regression network for monocular depth estimation*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2002–2011.

[77] GAIST, *Highways*, 2020.
`https://www.gaist.co.uk/intelligence-for-highways`
Accessed: 08.02.2021.

[78] D. GARCÍA AND D. TCHERNIAK, *An experimental study on the data-driven structural health monitoring of large wind turbine blades using a single accelerometer and actuator*, Mechanical Systems and Signal Processing, 127 (2019), pp. 102–119.

[79] J. A. GHANI AND S. P. DESHPANDE, *Task characteristics and the experience of optimal flow in human computer interaction*, The Journal of Psychology, 128 (1994), pp. 381–391.

[80] S. K. GHOSH, M. SUMAN, R. DATTA, AND P. K. BISWAS, *Power efficient event detection scheme in wireless sensor networks for railway bridge health monitoring system*, in Proceedings of 2014 IEEE International Conference on Advanced Networks and Telecommuncations Systems (ANTS), 2014, pp. 1–6.

[81] A. GIL, Ã. REINOSO, M. BALLESTA, AND M. JULIÁ, *Multi-robot visual SLAM using a Rao-Blackwellized particle filter*, Robotics and Autonomous Systems, 58 (2010), pp. 68–80.

[82] F. GLOVER, *Tabu search part 1*, ORSA Journal on Computing, 1 (1989), pp. 190–206.

[83] H. GOLDSTEIN, *Classical Mechanics*, Addison-Wesley, Boston, MA, 3 ed., 2002.

[84] M. GOLPARVAR-FARD, V. BALALI, AND J. M. DE LA GARZA, *Segmentation and recognition of highway assets using image-based 3D point clouds and semantic texton forests*, Journal of Computing in Civil Engineering, 29 (2015), p. 04014023.

[85] R. C. GONZALEZ AND R. E. WOODS, *Digital Image Processing*, Pearson, London, UK, 4 ed., 2018.

[86] GOOGLE, *Google documents*, 2021.
`https://www.google.co.uk/sheets/about`
Accessed: 26.07.2021.

[87] ——, *Google earth*, 2021.
`https://earth.google.com/web`
Accessed: 26.07.2021.

[88] GOVERNMENT BUILDINGS, *UK government property and land database*, 2021.
https://governmentbuildings.co.uk/
Accessed: 10.06.2021.

[89] K. GRACE, J. SALVATIER, A. DAFOE, B. ZHANG, AND O. EVANS, *When will AI exceed human performance? Evidence from AI experts*, Journal of Artificial Intelligence Research, 62 (2018), pp. 729–754.

[90] GRAFANA LABS, *Grafana: The open observability platform*, 2021.
https://grafana.com/
Accessed: 19.07.2021.

[91] A. GRAVES, A.-R. MOHAMED, AND G. HINTON, *Speech recognition with deep recurrent neural networks*, in Proceedings of 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, 2013, pp. 6645–6649.

[92] J. GREENHALGH AND M. MIRMEHDI, *Real-time detection and recognition of road traffic signs*, IEEE Transactions on Intelligent Transportation Systems, 13 (2012), pp. 1498–1506.

[93] D. GRIFFIN AND J. LIM, *Signal estimation from modified short-time Fourier transform*, IEEE Transactions on Acoustics, Speech, and Signal Processing, 32 (1984), pp. 236–243.

[94] S. GUNNER, P. J. VARDANEGA, T. TRYFONAS, J. H. G. MACDONALD, AND R. E. WILSON, *Rapid deployment of a WSN on the Clifton Suspension Bridge, UK*, in Proceedings of the Institution of Civil Engineers — Smart Infrastructure and Construction, vol. 170, 2017, pp. 59–71.

[95] P. HAGGARD, S. CLARK, AND J. KALOGERAS, *Voluntary action and conscious awareness*, Nature Neuroscience, 5 (2002), pp. 382–385.

[96] R. W. HALL, *Incident dispatching, clearance and delay*, Transportation Research Part A: Policy and Practice, 36 (2002), pp. 1–16.

[97] R. HARTLEY AND A. ZISSERMAN, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, 2 ed., 2004.

[98] J. HARTMANN, J. H. KLÜSSENDORFF, AND E. MAEHLE, *A comparison of feature descriptors for visual SLAM*, in Proceedings of 2013 European Conference on Mobile Robots, 2013, pp. 56–61.

[99] K. HE, X. ZHANG, S. REN, AND J. SUN, *Deep residual learning for image recognition*, in Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[100] M. He, C. Zhu, Q. Huang, B. Ren, and J. Liu, *A review of monocular visual odometry*, The Visual Computer, 36 (2020), pp. 1053–1065.

[101] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, et al., *CNN architectures for large-scale audio classification*, in Proceedings of 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017, pp. 131–135.

[102] Highways Agency, *A1(M) Bramham to Wetherby*, 2009.
https://web.archive.org/web/20100627202057/http://www.highways.gov.uk/roads/projects/5526.aspx
Accessed: 30.10.2021.

[103] Highways England, *Strategic road network statistics*, 2015.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/448276/strategic-road-network-statistics.pdf
Accessed: 22.03.2021.

[104] ——, *Highways England network journey time and traffic flow data*, 2016.
https://data.gov.uk/dataset/9562c512-4a0b-45ee-b6ad-afc0f99b841f/highways-england-network-journey-time-and-traffic-flow-data
Accessed: 30.10.2021.

[105] ——, *Highways England strategic road network initial report*, 2017.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/666857/Strategic_Road_Network_Initial_Report_Overview.pdf
Accessed: 30.03.2021.

[106] ——, *The road to growth: Our strategic economic growth plan*, 2017.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/600275/m160503_the_road_to_growth_Our_strategic_economic_growth_plan.pdf
Accessed: 22.03.2021.

[107] ——, *Highways Agency network journey time and traffic flow data*, 2018.
https://data.gov.uk/dataset/dc18f7d5-2669-490f-b2b5-77f27ec133ad/highways-agency-network-journey-time-and-traffic-flow-data
Accessed: 29.10.2021.

[108] ——, *Traffic statistics, methodology review: Alternative data sources*, 2018.

https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/722515/traffic-statistics-methodology-review-alternative-data-sources.pdf
Accessed: 25.08.2021.

[109] ——, *Traffic statistics methodology review: Overview*, 2018.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/722511/traffic-statistics-methodology-review.pdf
Accessed: 27.08.2021.

[110] ——, *Asset data management manual, part 2 — requirements and additional information*, 2019.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/722515/traffic-statistics-methodology-review-alternative-data-sources.pdf
Accessed: 25.08.2021.

[111] ——, *Operational metrics manual*, 2019.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/775149/Operational_Metrics_Manual.pdf
Accessed: 20.09.2021.

[112] HIGHWAYS ENGLAND, *Traffic officer vehicle equipment*, 2019.
https://www.youtube.com/watch?v=7DQHq59_eLo
Accessed: 30.06.2021.

[113] HIGHWAYS ENGLAND, *Customer service and operations*, 2020.
https://careers.highwaysengland.co.uk/jobs-you-can-do/customer-service-and-operations
Accessed: 29.10.2021.

[114] ——, *Economic analysis of the second road period*, 2020.
https://highwaysengland.co.uk/media/vs3h1jx2/gfd20_0072-economic-analysis-of-rp2-brochure_v4.pdf
Accessed: 22.03.2021.

[115] ——, *Highways England asset management strategy*, 2020.
http://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/860289/Asset_Management_Strategy_Low_Res.pdf
Accessed: 22.03.2021.

[116] ——, *Highways England fleet vehicles*, 2020.
https://www.gov.uk/government/publications/highways-england-fleet-vehicles/
Accessed: 30.10.2020.

[117] ——, *Our roads*, 2020.
https://highwaysengland.co.uk/about-us/our-roads/
Accessed: 30.10.2020.

[118] ——, *Out with the traffic officers of Highways England, and what they do for us*, 2020.
https://www.linkedin.com/pulse/out-traffic-officers-highways-england-what-do-us-simon-montague
Accessed: 22.03.2021.

[119] HIGHWAYS ENGLAND, *Road investment strategy: 2015 to 2020*, 2020.
https://www.gov.uk/government/collections/road-investment-strategy
Accessed: 22.03.2021.

[120] HIGHWAYS ENGLAND, *Traffic officer fleet vehicles*, 2020.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/741217/FOI_765216_-_HE_Fleet_Vehicles_-_TO_Vehicles.csv/preview
Accessed: 30.10.2021.

[121] ——, *Traffic officer manual*, 2020.
https://www.whatdotheyknow.com/request/341964/response/841566/attach/5/TO%20Manual%20Work%20Instructions%2008.07.16docx.pdf?cookie_passthrough=1
Accessed: 22.03.2021.

[122] ——, *Where we're based*, 2020.
https://careers.highwaysengland.co.uk/how-we-work/where-we-re-based
Accessed: 06.11.2020.

[123] ——, *CS 450 inspection of highway structures*, 2021.
https://www.standardsforhighways.co.uk/prod/attachments/c5c2c3e5-f7f3-4c94-8254-184e41ccd1a0?inline=true
Accessed: 15.09.2021.

[124] ——, *Geotechnical data management system*, 2021.
https://www.hagdms.com
Accessed: 10.05.2021.

[125] ——, *Highways England pavement management system network layer HAPMS*, 2021.
https://data.gov.uk/dataset/64df0f9e-cb98-4413-a88c-31dc55dd75f0/highways-england-pavement-management-system-network-layer-hapms
Accessed: 10.05.2021.

[126] ——, *National traffic information service publish services*, 2021.
https://www.trafficengland.com/resources/cms-docs/overview.pdf
Accessed: 10.06.2021.

[127] ——, *Our latest traffic information*, 2021.
https://www.trafficengland.com
Accessed: 09.06.2021.

[128] ——, *Smart motorways stocktake: First year progress report 2021*, 2021.
https://highwaysengland.co.uk/media/bb4lpkcp/smart-motorways-stocktake-first-year-progress-report-2021.pdf
Accessed: 10.06.2021.

[129] ——, *Stopped vehicle detection upgrades — making journeys even safer with radar-based technology*, 2021.
https://highwaysengland.co.uk/our-work/smart-motorways-evidence-stocktake/stopped-vehicle-detection-upgrades/
Accessed: 24.08.2021.

[130] Highways England Asset Management Development Group, *Asset data managemnt manual*, 2020.
https://data.gov.uk/dataset/64df0f9e-cb98-4413-a88c-31dc55dd75f0/highways-england-pavement-management-system-network-layer-hapms
Accessed: 20.05.2021.

[131] Highways Magazine, *Life after SCANNER in Bristol*, 2020.
https://www.highwaysmagazine.co.uk/Life-after-SCANNER-in-Bristol/5465
Accessed: 10.05.2021.

[132] E. Huising and L. Gomes Pereira, *Errors and accuracy estimates of laser data acquired by various laser scanning systems for topographic applications*, ISPRS Journal of Photogrammetry and Remote Sensing, 53 (1998), pp. 245–261.

[133] InfluxDB, *InfluxDB: Purpose-built open*, 2021.
https://www.influxdata.com/
Accessed: 19.07.2021.

[134] INFRASTRUCTURE AND PROJECT AUTHORITY, *Transforming infrastructure performance: Roadmap to 2030*, 2021.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/1016726/IPA_TIP_Roadmap_to_2030_v6__1_.pdf
Accessed: 13.09.2021.

[135] INFRASTRUCTURE AUSTRALIA, *What we do*, 2021.
https://www.infrastructureaustralia.gov.au/what-we-do
Accessed: 15.10.2021.

[136] L. IOCCHI, L. MARCHETTI, AND D. NARDI, *Multi-robot patrolling with coordinated behaviours in realistic environments*, in Proceedings of 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2011, pp. 2796–2801.

[137] JACOBS, *Transport*.
https://www.jacobs.com/solutions/markets/infrastructure/transportation
Accessed: 10.06.2022, 2022.

[138] M. JALAYER, H. ZHOU, J. GONG, S. HU, AND M. GRINTER, *A comprehensive assessment of highway inventory data collection methods*, Journal of the Transportation Research Forum, 53 (2014), pp. 73–92.

[139] G. JIN, X. ZHONG, S. FANG, X. DENG, AND J. LI, *Keyframe-based dynamic elimination SLAM system using YOLO detection*, in Proceedings of 2019 International Conference on Intelligent Robotics and Applications (ICRA), 2019, pp. 697–705.

[140] P. P. JOVANIS AND H.-L. CHANG, *Modelling the relationship of accidents to miles traveled*, Transportation Research Record, 1068 (1986), pp. 42–51.

[141] F. KARRAY, M. ALEMZADEH, J. ABOU SALEH, AND M. N. ARAB, *Human-computer interaction: Overview on state of the art*, International Journal on Smart Sensing and Intelligent Systems, 1 (2017).

[142] T. KENT, A. RICHARDS, AND A. JOHNSON, *Single-agent policies for the multi-agent persistent surveillance problem via artificial heterogeneity*, in Proceedings of 2020 European Conference on Multi-Agent Systems, 2020, pp. 1–18.

[143] S. KIM, S. PAKZAD, D. CULLER, J. DEMMEL, G. FENVES, S. GLASER, AND M. TURON, *Wireless sensor networks for structural health monitoring*, in Proceedings of the 4th International Conference on Embedded Networked Sensor Systems, 2006, pp. 427–428.

[144] S. KIRKPATRICK, C. D. GELATT, AND M. P. VECCHI, *Optimization by simulated annealing*, Science, 220 (1983), pp. 671–680.

[145] G. KLEIN AND D. MURRAY, *Parallel tracking and mapping for small AR workspaces*, in 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, 2007, pp. 225–234.

[146] L. A. KLEIN, *ITS Sensors and Architectures for Traffic Management and Connected Vehicles*, CRC Press, Boca Raton, FL, 2017.

[147] A. KRIZHEVSKY, I. SUTSKEVER, AND G. E. HINTON, *Imagenet classification with deep convolutional neural networks*, in Proceedings of the 25th International Conference on Neural Information Processing Systems, 2012, pp. 1097–1105.

[148] N. KRYVINSKA, A. PONISZEWSKA-MARANDA, AND M. GREGUS, *An approach towards service system building for road traffic signs detection and recognition*, Procedia Computer Science, 141 (2018), pp. 64–71.

[149] L3HARRIS, *Using imagery and image analysis software to manage highway assets*, 2021.
`https://www.l3harrisgeospatial.com/Support/Maintenance-Detail/ArtMID/13350/ArticleID/18339/5102`
Accessed: 07.06.2021.

[150] LAND MOBILE: COMMUNICATIONS FOR BUSINESS, *Highways England upgrades its critical communications with Sepura*, 2020.
`https://www.landmobile.co.uk/news/highways-england-upgrades-its-critical-communications-with-sepura-1`
Accessed: 22.03.2021.

[151] G. LAPORTE, S. ROPKE, AND T. VIDAL, *Chapter 4: Heuristics for the vehicle routing problem*, in Vehicle Routing: Problems, Methods, and Applications, Second Edition, SIAM, 2014, pp. 87–116.

[152] J. D. LEES-MILLER AND R. E. WILSON, *Sampling of redistribution of empty vehicles for personal rapid transit*, Transportation Research Record, (2011), pp. 174–181.

[153] M. LEI, D. LEFLOCH, P. GOUTON, AND K. MADANI, *A video-based real-time vehicle counting system using adaptive background method*, in Proceedings of 2008 IEEE International Conference on Signal Image Technology and Internet Based Systems, 2008, pp. 523–528.

[154] G. LENZ, *Tacoma narrows bridge collapse*, 2008.
`https://www.youtube.com/watch?v=lXyG68_caV4`,
Accessed: 24.08.2022.

[155] S. LEUTENEGGER, P. FURGALE, V. RABAUD, M. CHLI, K. KONOLIGE, AND R. SIEGWART, *Keyframe-based visual-inertial SLAM using nonlinear optimization*, Proceedings of Robotics Science and Systems (RSS) 2013, (2013).

[156] Q. LI, Y. HE, K. ZHOU, X. HAN, Y. HE, AND Z. SHU, *Structural health monitoring for a 600m high skyscraper*, The Structural Design of Tall and Special Buildings, 27 (2018), p. e1490.

[157] T.-Y. LIN, M. MAIRE, S. BELONGIE, J. HAYS, P. PERONA, D. RAMANAN, P. DOLLÁR, AND C. L. ZITNICK, *Microsoft COCO: Common objects in context*, in Proceedings of 2014 European Conference on Computer Vision (ECCV), 2014, pp. 740–755.

[158] F. LIU, C. SHEN, AND G. LIN, *Deep convolutional neural fields for depth estimation from a single image*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5162–5170.

[159] R. P. LOCE, R. BALA, AND M. TRIVEDI, *Computer Vision and Imaging in Intelligent Transportation Systems*, Wiley-IEEE Press, Hoboken, NJ, 1st ed., 2017.

[160] D. LORD, S. P. WASHINGTON, AND J. N. IVAN, *Poisson, poisson-gamma and zero-inflated regression models of motor vehicle crashes: Balancing statistical fit and theory*, Accident Analysis and Prevention, 37 (2005), pp. 35–46.

[161] LORD MICROSTRAIN, *G-Link-200*, 2021.
`https://www.microstrain.com/wireless-sensors/g-link-200`
Accessed: 26.07.2021.

[162] ——, *MSCL — The microstrain communication library*, 2021.
`https://github.com/LORD-MicroStrain/MSCL`
Accessed: 19.07.2021.

[163] ——, *V-Link-200*, 2021.
`https://www.microstrain.com/wireless-sensors/v-link-200`
Accessed: 19.07.2021.

[164] D. G. LOWE, *Object recognition from local scale-invariant features*, in Proceedings of the IEEE International Conference on Computer Vision, 1999, pp. 1150–1157.

[165] W. MA, D. XING, A. MCKEE, R. BAJWA, C. FLORES, B. FULLER, AND P. VARAIYA, *A wireless accelerometer-based automatic vehicle classification prototype system*, IEEE Transactions on Intelligent Transportation Systems, 15 (2014), pp. 104–111.

[166] J. H. G. MACDONALD, *Pedestrian-induced vibrations of the Clifton Suspension Bridge, UK*, Proceedings of the Institution of Civil Engineers — Bridge Engineering, 161 (2008), pp. 69–77.

[167] A. MACHADO, G. RAMALHO, J.-D. ZUCKER, AND A. DROGOUL, *Multi-agent patrolling: An empirical analysis of alternative architectures*, in Proceedings of International Workshop on Multi-Agent Systems and Agent-Based Simulation, 2003, pp. 155–170.

[168] A. MALEKJAFARIAN, P. J. MCGETRICK, AND E. J. OBRIEN, *A review of indirect bridge monitoring using passing vehicles*, Shock and Vibration, 2015 (2015), p. 286139.

[169] M. MANCINI, G. COSTANTE, P. VALIGI, AND T. A. CIARFUGLIA, *Fast robust monocular depth estimation for obstacle detection with fully convolutional networks*, in Proceedings of 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016, pp. 4296–4303.

[170] MAPILLARY, *Mapillary*, 2021.
`https://help.mapillary.com/hc/en-us`
Accessed: 07.06.2021.

[171] MASSACHUSETTS INSTITUTE OF TECHNOLOGY, *MIT app inventor*, 2021.
`https://appinventor.mit.edu`
Accessed: 26.07.2021.

[172] S. MCLOUGHLIN, C. DEEGAN, C. MULVIHILL, C. FITZGERALD, AND C. MARKHAM, *Mobile mapping for the automated analysis of road signage and delineation*, IET Intelligent Transport Systems, 2 (2008), pp. 61–73.

[173] A. MELO, M. BELCHIOR, AND V. FURTADO, *Analyzing police patrol routes with the simulation of the physical reorganization of agents*, in Proceedings International Workshop on Multi-Agent Systems and Agent-Based Simulation, 2005, pp. 99–114.

[174] MOBILEYE, *Ordnance Survey and Mobileye create a new type of data collection*, 2020.
`https://www.mobileye.com/uk/fleets/case-study/ordnance-survey-mobileye-create-a-new-type-of-data-collection/`
Accessed: 10.05.2021.

[175] A. MOGELMOSE, M. M. TRIVEDI, AND T. B. MOESLUND, *Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey*, IEEE Transactions on Intelligent Transportation Systems, 13 (2012), pp. 1484–1497.

[176] J. W. MOORE, *What is the sense of agency and why does it matter?*, Frontiers in Psychology, 7 (2016), p. 1272.

[177] R. MOSS AND S. MATTHEWS, *In-service structural monitoring: A state of the art review*, Structural Engineer, 73 (1995), pp. 23–31.

[178] R. MUR-ARTAL, J. M. MONTIEL, AND J. D. TARDOS, *ORB-SLAM: A versatile and accurate monocular SLAM system*, IEEE Transactions on Robotics, 31 (2015), pp. 1147–1163.

[179] NATIONAL ACADEMIES OF SCIENCES, ENGINEERING, AND MEDICINE, *Use of transportation asset management principles in state highway agencies*, The National Academies Press, 2013.

[180] NATIONAL INFRASTRUCTURE COMMITTEE, *National infrastructure assessment*, 2021.
https://nic.org.uk/app/uploads/CCS001_CCS0618917350-001_NIC-NIA_Accessible-1.pdf
Accessed: 13.09.2021.

[181] NATIONALE BROEPEN GIDS, *Road inspector: Salary, duties, vacancies, education, future and skills*, 2021.
https://www.nationaleberoepengids.nl/weginspecteur#about
Accessed: 09.06.2021.

[182] P. NEGRI, X. CLADY, S. M. HANIF, AND L. PREVOST, *A cascade of boosted generative and discriminative classifiers for vehicle detection*, EURASIP Journal on Advances in Signal Processing, 2008 (2008), pp. 1–12.

[183] G. NEUHOLD, T. OLLMANN, S. ROTA BULO, AND P. KONTSCHIEDER, *The Mapillary vistas dataset for semantic understanding of street scenes*, in Proceedings of the IEEE International Conference on Computer Vision (ICCV), Oct 2017.

[184] NEW CIVIL ENGINEER, *Motorway traffic officers equipped to deal with cardiac arrest*, 2020.
https://www.newcivilengineer.com/latest/motorway-traffic-officers-equipped-to-deal-with-cardiac-arrest-16-03-2020
Accessed: 01.03.2021.

[185] ——, *Higher death rate recorded on smart motorways compared to hard shoulder roads, new figures show*, 2021.
https://www.newcivilengineer.com/latest/higher-death-rate-recorded-on-smart-motorways-compared-to-hard-shoulder-roads-new-figures-show-05-05-2021/
Accessed: 10.06.2021.

[186] NHS ENGLAND, *The new ambulance standards*, 2017.
https://www.england.nhs.uk/wp-content/uploads/2017/07/new-ambulance-standards-easy-read.pdf
Accessed: 01.03.2020.

[187] N. NIGAM, *The multiple unmanned air vehicle persistent surveillance problem: A review*, Machines, 2 (2014), pp. 13–72.

[188] OFFICE OF ROAD AND RAIL, *Annual assessment of Highways England's performance*, 2019.
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/818764/annual-assessment-of-highways-englands-performance-2019-print.pdf
Accessed: 04.03.2021.

[189] S. OH, S. G. RITCHIE, AND C. OH, *Real-time traffic measurement from single loop inductive signatures*, Transportation Research Record, 1804 (2002), pp. 98–106.

[190] K. P. O'KEEFFE, A. ANJOMSHOAA, S. H. STROGATZ, P. SANTI, AND C. RATTI, *Quantifying the sensing power of vehicle fleets*, Proceedings of the National Academy of Sciences, 116 (2019), pp. 12752–12757.

[191] OPENSTREETMAP, *Welcome to OpenStreetMap!*, 2021.
https://www.openstreetmap.org/
Accessed: 10.06.2021.

[192] A. V. OPPENHEIM AND R. W. SCHAFER, *Discrete-time Signal Processing*, Pearson Education Limited, Harlow, UK, 2 ed., 2004.

[193] ORGANISATION FOR ECONOMIC CO-OPERATION AND DEVELOPMENT, *Transport — Infrastructure investment*, 2021.
https://data.oecd.org/transport/infrastructure-investment.htm
Accessed: 15.10.2021.

[194] W. OSTACHOWICZ, R. SOMAN, AND P. MALINOWSKI, *Optimization of sensor placement for structural health monitoring: A review*, Structural Health Monitoring, 18 (2019), pp. 963–988.

[195] A. PANTELIAS, *Asset management data collection for supporting decision processes*, PhD thesis, Virginia Tech, 2005.

[196] R. PARASURAMAN AND V. RILEY, *Humans and automation: Use, misuse, disuse, abuse*, Human Factors, 39 (1997), pp. 230–253.

[197] R. PARASURAMAN, T. SHERIDAN, AND C. WICKENS, *A model for types and levels of human interaction with automation*, IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 30 (2000), pp. 286–297.

[198] J. PARKKA, M. ERMES, P. KORPIPAA, J. MANTYJARVI, J. PELTOLA, AND I. KORHONEN, *Activity classification using realistic data from wearable sensors*, IEEE Transactions on Information Technology in Biomedicine, 10 (2006), pp. 119–128.

[199] R. K. PATEL AND V. GIRI, *Feature selection and classification of mechanical fault of an induction motor using random forest classifier*, Perspectives in Science, 8 (2016), pp. 334–337.

[200] B. PICINBONO, *On instantaneous amplitude and phase of signals*, IEEE Transactions on Signal Processing, 45 (1997), pp. 552–560.

[201] V. PILLAC, M. GENDREAU, C. GUÃL'RET, AND A. L. MEDAGLIA, *A review of dynamic vehicle routing problems*, European Journal of Operational Research, 225 (2013), pp. 1–11.

[202] M. A. PIMENTEL, A. E. JOHNSON, P. H. CHARLTON, D. BIRRENKOTT, P. J. WATKINSON, L. TARASSENKO, AND D. A. CLIFTON, *Toward a robust estimation of respiratory rate from pulse oximeters*, IEEE Transactions on Biomedical Engineering, 64 (2016), pp. 1914–1923.

[203] H. PIRKUL AND D. A. SCHILLING, *The maximal covering location problem with capacities on total workload*, Management Science, 37 (1991), pp. 233–248.

[204] POLIZEI BRANDENBURG, *Autobahnpolizei*, 2021.
`https://polizei.brandenburg.de/liste/autobahnpolizei/62080`
Accessed: 09.06.2021.

[205] POLIZIA DI STATO, *Le attività della stradale*, 2021.
`https://www.poliziadistato.it/articolo/le-attivita-della-stradale`
Accessed: 09.06.2021.

[206] D. POLLITT, *Highways Agency puts traffic officers in the driving seat: Two training programs help patrols to cut motorway congestion*, Human Resource Management International Digest, 17 (2009), pp. 20–22.

[207] POTHOLE SPOTTER, *Pothole-spotter trial*, 2017.
`http://www.pothole-spotter.co.uk/`
Accessed: 15.09.2021.

[208] C. PRINS, *A simple and effective evolutionary algorithm for the vehicle routing problem*, Computers and Operations Research, 31 (2004), pp. 1985–2002.

[209] C. PRINS, *A GRASP × evolutionary local search hybrid or the vehicle routing problem*, in Bio-inspired Algorithms for the Vehicle Routing Problem, F. B. Pereira, ed., Springer, Berlin, Heidelberg, 2009, pp. 35–53.

[210] QUARTZ, *The Quartz guide to artificial intelligence: What is it, why is it important, and should we be afraid?*, 2017.

https://qz.com/1046350/the-quartz-guide-to-artificial-intelligence-what-is-it-why-is-it-important-and-should-we-be-afraid
Accessed: 09.11.2021.

[211] QUORA, *What are the black strips laid across a road measuring?*
https://www.quora.com/What-are-the-black-strips-laid-across-a-road-measuring
Accessed: 25.08.2021, 2019.

[212] S. C. RADOPOULOU AND I. BRILAKIS, *Automated detection of multiple pavement defects*, Journal of Computing in Civil Engineering, 31 (2017), p. 04016057.

[213] RASPBERRY PI FOUNDATION, *Teach, learn, and make with raspberry pi*, 2022.
https://www.raspberrypi.org/ Accessed: 21.01.2022.

[214] J. REDMON AND A. FARHADI, *YOLOv3: An incremental improvement*, arXiv preprint arXiv:1804.02767, (2018).

[215] P. REFAEILZADEH, L. TANG, AND H. LIU, *Cross-Validation*, Springer US, Boston, MA, 2009, pp. 532–538.

[216] A. REILLY, G. FRAZER, AND B. BOASHASH, *Analytic signal generation-tips and traps*, IEEE Transactions on Signal Processing, 42 (1994), pp. 3241–3245.

[217] D. REIS, A. MELO, A. L. V. COELHO, AND V. FURTADO, *Towards optimal police patrol routes with genetic algorithms*, in Intelligence and Security Informatics, S. Mehrotra, D. D. Zeng, H. Chen, B. Thuraisingham, and F.-Y. Wang, eds., Berlin, Heidelberg, 2006, Springer Berlin Heidelberg, pp. 485–491.

[218] A. REISS, I. INDLEKOFER, P. SCHMIDT, AND K. VAN LAERHOVEN, *Deep PPG: Large-scale heart rate estimation with convolutional neural networks*, Sensors, 19 (2019), p. 3079.

[219] RETAIL SENSING, *Piezoelectric sensor*, 2021.
https://www.retailsensing.com/definition/piezoelectric-sensor.html
Accessed: 25.08.2021.

[220] RIJKSWATERSTAAT, *Road inspector at Rijkswaterstaat*, 2021.
https://werkenbij.rijkswaterstaat.nl/vakgebieden/weginspecteur
Accessed: 09.06.2021.

[221] RIJKWATERSTAAT, *Night shift in the Veluwe*, 2021.
https://www.rijkswaterstaat.nl/wegen/wegbeheer/weginspecteurs/alle-blogs/nachtdienst-op-de-veluwe
Accessed: 10.06.2021.

[222] J. RIVAS, R. WUNDERLICH, AND S. J. HEINEN, *Road vibrations as a source to detect the presence and speed of vehicles*, IEEE Sensors Journal, 17 (2017), pp. 377–385.

[223] E. RUBLEE, V. RABAUD, K. KONOLIGE, AND G. BRADSKI, *ORB: An efficient alternative to SIFT or SURF*, in Proceedings of the IEEE International Conference on Computer Vision, 2011, pp. 2564–2571.

[224] D. E. RUMELHART, G. E. HINTON, AND R. J. WILLIAMS, *Learning representations by back-propagating errors*, Nature, 323 (1986), pp. 533–536.

[225] SAFETY AND HEALTH PRACTITIONER, *How safe are smart motorways? Only 48% of motorists say they know how to use them*, 2021.
`https://www.shponline.co.uk/road-safety/how-safe-are-smart-motorways`
Accessed: 10.06.2021.

[226] N. SAIRAM, S. NAGARAJAN, AND S. ORNITZ, *Development of mobile mapping system for 3D road asset inventory*, Sensors, 16 (2016), p. 367.

[227] D. SCHRANK, *The 2002 urban mobility report*, 2002.
`http://mobility.tamu.edu`
Accessed: 01.03.2021.

[228] S. C. SEN-CHING AND C. KAMATH, *Robust techniques for background subtraction in urban traffic video*, in Proceedings of Visual Communications and Image Processing 2004, 2004, pp. 881–892.

[229] P. SERMANET AND Y. LECUN, *Traffic sign recognition with multi-scale convolutional networks*, in Proceedings of 2011 International Joint Conference on Neural Networks, 2011, pp. 2809–2813.

[230] H.-C. SHIN, H. ROTH, M. GAO, L. LU, Z. XU, I. NOGUES, J. YAO, D. J. MOLLURA, AND R. M. SUMMERS, *Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning*, IEEE Transactions on Medical Imaging, 35 (2016), pp. 1285–1298.

[231] K. SIMONYAN AND A. ZISSERMAN, *Very deep convolutional networks for large-scale image recognition*, in Proceedings 3rd International Conference on Learning Representations, ICLR 2015, 2015.

[232] K. C. SINHA, S. LABI, AND B. R. D. K. AGBELIE, *Transportation infrastructure asset management in the new millennium: Continuing issues, and emerging challenges and opportunities*, Transportmetrica A: Transport Science, 13 (2017), pp. 591–606.

[233] H. SOHN, C. R. FARRAR, F. M. HEMEZ, D. D. SHUNK, D. W. STINEMATES, B. R. NADLER, AND J. J. CZARNECKI, *A review of structural health monitoring literature: 1996–2001*, Los Alamos National Laboratory, USA, 1 (2003).

[234] H. SONG, H. LIANG, H. LI, Z. DAI, AND X. YUN, *Vision-based vehicle detection and counting system using deep learning in highway scenes*, European Transport Research Review, 11 (2019), pp. 1–16.

[235] J. STALLKAMP, M. SCHLIPSING, J. SALMEN, AND C. IGEL, *The German traffic sign recognition benchmark: A multi-class classification competition*, in Proceedings of the 2011 International Joint Conference on Neural Networks, IEEE, 2011, pp. 1453–1460.

[236] STATISTA, *Volume of freight moved by trucks in the United States from 2014 to 2019*, 2021.
`https://www.statista.com/statistics/1140181/volume-freight-trucks-united-states`
Accessed: 14.09.2021.

[237] J. STEENBRUGGEN, M. KUSTERS, AND G. BROEKHUIZEN, *Best practice in European traffic incident management*, Procedia-Social and Behavioral Sciences, 48 (2012), pp. 297–310.

[238] T. STRAIN, S. GUNNER, AND R. WILSON, *Estimation of vehicle counts from the structural response of a bridge*, in Proceedings of the 2019 International Conference on Smart Infrastructure and Construction (ICSIC), ICE Publishing, 2019, pp. 751–759.

[239] T. STRAIN, R. E. WILSON, A. CALWAY, AND R. LITTLEWORTH, *Augmented visual SLAM for the localisation of a transportation asset management survey vehicle*, in Proceedings of 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), 2020, pp. 1–6.

[240] T. STRAIN, R. E. WILSON, AND R. LITTLEWORTH, *Role of traffic officers in transportation asset monitoring*.
in Proceedings of the 2022 Transportation Research Board Conference (forthcoming).

[241] ——, *Computer vision for rapid updating of the highway asset inventory*, Transportation Research Record, 2674 (2020), pp. 245–255.

[242] H. STRASDAT, J. M. MONTIEL, AND A. J. DAVISON, *Real-time monocular SLAM: Why filter?*, in Proceedings of 2010 IEEE International Conference on Robotics and Automation (ICRA), 2010, pp. 2657–2664.

[243] P. STURM, *Pinhole Camera Model*, Springer US, Boston, MA, 2014, pp. 610–613.

[244] C. SZEGEDY, WEI LIU, YANGQING JIA, P. SERMANET, S. REED, D. ANGUELOV, D. ERHAN, V. VANHOUCKE, AND A. RABINOVICH, *Going deeper with convolutions*, in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015, pp. 1–9.

[245] T. TAKETOMI, H. UCHIYAMA, AND S. IKEDA, *Visual SLAM algorithms: A survey from 2010 to 2016*, IPSJ Transactions on Computer Vision and Applications, 9 (2017), pp. 1–11.

[246] C. W. TAN, C. BERGMEIR, F. PETITJEAN, AND G. I. WEBB, *Time series extrinsic regression*, Data Mining and Knowledge Discovery, 35 (2021), pp. 1032–1060.

[247] TECH.EU, *A spotlight on Sweden's Mapillary: The street-level imagery platform mapping the world's places*, 2019.
https://tech.eu/features/28616/a-spotlight-on-swedens-mapillary-the-street-level-imagery-platform-mapping-the-worlds-places/
Accessed: 07.06.2021.

[248] F. H. C. TIVIVE, A. BOUZERDOUM, AND M. G. AMIN, *A human gait classification method based on radar doppler spectrograms*, EURASIP Journal on Advances in Signal Processing, 2010 (2010), pp. 1–12.

[249] B. TRIGGS, P. F. MCLAUCHLAN, R. I. HARTLEY, AND A. W. FITZGIBBON, *Bundle adjustment — a modern synthesis*, in Proceedings of 1999 International Workshop on Vision Algorithms, 1999, pp. 298–372.

[250] W. UDDIN, *Airborne laser terrain mapping for expediting highway projects: Evaluation of accuracy and cost*, Journal of Construction Engineering and Management, 134 (2008), pp. 411–420.

[251] U.S. DEPARTMENT OF TRANSPORTATION, *Model inventory of roadway elements – MIRE 2.0*.
https://rosap.ntl.bts.gov/view/dot/49568 (Accessed: 20.09.2021), 2017.

[252] ——, *Asset management*, 2020.
https://www.fhwa.dot.gov/asset/
Accessed: 01.03.2021.

[253] ——, *Asset management overview*, 2021.
https://www.fhwa.dot.gov/asset/if08008/amo_02.cfm
Accessed: 14.09.2021.

[254] ——, *MAP-21 – moving ahead for progress in the 21st century*, 2021.
https://www.fhwa.dot.gov/map21/
Accessed: 14.09.2021.

[255] ——, *Transportation asset management plans*, 2021.
https://www.fhwa.dot.gov/asset/plans.cfm
Accessed: 14.09.2021.

[256] B. USLU, M. GOLPARVAR-FARD, AND J. M. DE LA GARZA, *Image-based 3D reconstruction and recognition for enhanced highway condition assessment*, in Proceedings of the 2011 Congress on Computing in Civil Engineering, 2011, pp. 67–76.

[257] M. VALENTI, S. SQUARTINI, A. DIMENT, G. PARASCANDOLO, AND T. VIRTANEN, *A convolutional neural network approach for acoustic scene classification*, in Proceedings of 2017 International Joint Conference on Neural Networks (IJCNN), 2017, pp. 1547–1554.

[258] C. WANG, M. A. QUDDUS, AND S. G. ISON, *Impact of traffic congestion on road accidents: A spatial analysis of the M25 motorway in England*, Accident Analysis and Prevention, 41 (2009), pp. 798–808.

[259] K. C. WANG, Z. HOU, AND W. GONG, *Automated road sign inventory system based on stereo vision and tracking*, Computer-Aided Civil and Infrastructure Engineering, 25 (2010), pp. 468–477.

[260] L. WANG, Y. LU, H. WANG, Y. ZHENG, H. YE, AND X. XUE, *Evolving boxes for fast vehicle detection*, in 2017 IEEE International Conference on Multimedia and Expo (ICME), IEEE, 2017, pp. 1135–1140.

[261] Z. WANG, Q. ZHANG, J. LI, S. ZHANG, AND J. LIU, *A computationally efficient semantic SLAM solution for dynamic scenes*, Remote Sensing, 11 (2019), p. 1363.

[262] A. WEINTRAUB, J. ABOUD, C. FERNANDEZ, G. LAPORTE, AND E. RAMIREZ, *An emergency vehicle dispatching system for an electric utility in chile*, Journal of the Operational Research Society, 50 (1999), pp. 690–696.

[263] M. J. WHELAN AND K. D. JANOYAN, *Design of a robust, high-rate wireless sensor network for static and dynamic structural monitoring*, Journal of Intelligent Material Systems and Structures, 20 (2009), pp. 849–863.

[264] K. I. WONG AND M. G. H. BELL, *The optimal dispatching of taxis under congestion: A rolling horizon approach*, Journal of Advanced Transportation, 40 (2006), pp. 203–220.

[265] WORLD HEALTH ORGANISATION, *Physical activity*, 2021.
`https://www.who.int/news-room/fact-sheets/detail/physical-activity`
Accessed: 25.08.2021.

[266] WORLD HIGHWAYS, *IBI's Routemapper charts new territory with Highways England*.
`https://www.worldhighways.com/wh12/feature/ibis-routemapper-charts-new-territory-highways-england` (Accessed: 10.05.2021), 2016.

[267] W. YU, F. LIANG, X. HE, W. G. HATCHER, C. LU, J. LIN, AND X. YANG, *A survey on the edge computing for the internet of things*, IEEE Access, 6 (2018), pp. 6900–6919.

[268] M. D. ZEILER AND R. FERGUS, *Visualizing and understanding convolutional networks*, in Proceedings of 2014 European Conference on Computer Vision, 2014, pp. 818–833.

[269] A. ZHANG, K. C. P. WANG, B. LI, E. YANG, X. DAI, Y. PENG, Y. FEI, Y. LIU, J. Q. LI, AND C. CHEN, *Automated pixel-level pavement crack detection on 3D asphalt surfaces using a deep-learning network*, Computer-Aided Civil and Infrastructure Engineering, 32 (2017), pp. 805–819.

[270] G. ZHANG, *Time series forecasting using a hybrid ARIMA and neural network model*, Neurocomputing, 50 (2003), pp. 159–175.

[271] J. ZHANG, Y. LU, Z. LU, C. LIU, G. SUN, AND Z. LI, *A new smart traffic monitoring method using embedded cement-based piezoelectric sensors*, Smart Materials and Structures, 24 (2015), p. 025023.

[272] Z. ZHOU, J. SHIN, L. ZHANG, S. GURUDU, M. GOTWAY, AND J. LIANG, *Fine-tuning convolutional neural networks for biomedical image analysis: Actively and incrementally*, in Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.

[273] L. ZHU, Y. FU, R. CHOW, B. F. SPENCER, J. W. PARK, AND K. MECHITOV, *Development of a high-sensitivity wireless accelerometer for structural health monitoring*, Sensors, 18 (2018), p. 262.

[274] Z. ZHU, D. LIANG, S. ZHANG, X. HUANG, B. LI, AND S. HU, *Traffic-sign detection and classification in the wild*, in Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2110–2118.

[275] S. ZILBERSTEIN, *Building strong semi-autonomous systems*, in Proceedings of 29th AAAI Conference on Artificial Intelligence, 2015, pp. 4088–4092.

[276] D. ZONTA, H. WU, M. POZZI, P. ZANON, M. CERIOTTI, L. MOTTOLA, G. P. PICCO, A. L. MURPHY, S. GUNA, AND M. CORRA, *Wireless sensor networks for permanent health monitoring of historic buildings*, Smart Structures and Systems, 6 (2010), pp. 595–618.