ASSURING AUTONOMY
INTERNATIONAL PROGRAMME

DEMONSTRATOR PROJECT

Final report

ALADDIN:

Assuring Long-term Autonomy through Detection and Diagnosis of Irregularities in Normal operation

JANUARY 2022

Project ALADDIN
Technical Report

Peng Wu, Enrico Anderlini, Izzat Kamarudzaman, Catherine A. Harris, Georgios Salavasidis, Alvaro Lorenzo Lopez, Eleanor Frajka-Williams, Giles Thomas, Yuanchang Liu

09/01/2022

Executive Summary

The Assuring Autonomy International Programme (AAIP) has the ambitious goal of delivering a Body of Knowledge (BoK) for the assurance of Robotics and Autonomous Systems (RAS). The demonstrator project "Assuring Long-term Autonomy through Detection and Diagnosis of Irregularities in Normal operation (ALADDIN)" (including project partners University College London (UCL) and National Oceanography Centre (NOC) and project stakeholders Lloyd's Register (LR), Maritime and Coastguard Agency (MCA) and Blue Ocean Monitoring (BOM)) have contributed to the drafting of the sections on the definition and verification of sensing and understanding requirements for RAS and the identification of sensing (and understanding) deviations. To achieve this, new methods for the detection and identification of adverse behaviour for Marine Autonomous Systems (MAS) have been introduced. Then, they are implemented on the command-control infrastructure for over-the-horizon operations of MAS (C2) – itself a RAS – developed by the NOC for verification. Additional field tests have also been implemented to validate the developed anomaly detection and fault diagnostics models.

The aim of WP1 was to define healthy and anomalous behaviour of RAS with appropriate vocabulary, identify data availability and sources for RAS deployments, with a focus on MAS, and outline the project steering direction for the following WPs. Specific datasets of deployments of a range of MAS technologies have been selected to be used in the other WPs.

Based on the identified datasets from WP1, an improved Bidirectional Generative Adversarial Networks (BiGAN) with assistive hints has been developed in WP2 to detect adverse behaviour of MAS. The developed anomaly detection system requires only normal operational data and do not require additional labelling efforts and can alert status that deviates from the system's normal operational pattern. Such features make it suitable for RAS operating in highly dynamic and uncertain environments such as oceans.

Furthermore, in WP3, supervised learning models have been developed to detect faults within a specific MAS domain with datasets covering the potential failure modes. It was concluded that although such models can achieve high fault diagnostics performance when the test platforms operate very similarly to the training platforms, it has limited capability of generalising across different MAS domains. To address the issue, the WP proposes a novel fault diagnostics deep learning model to diagnose faults for MAS via domain adaption and transfer learning, based upon the work developed in WP1. The proposed model, i.e., Marine Autonomous System Net (MASNet), is applied to address the challenging fault diagnostics tasks for distinct types of MAS that are under-observed and remotely operated in different regions and tasks by different institutions.

Two field tests, funded by the European Commission, have been implemented to test the anomaly detection and fault diagnostics methods developed in this project. Additionally, the developed tools have been implemented in NOC's command and control system for operational usage. Strategies to improve the robustness of ML systems running in production have also been discussed in the verification WP. The project management details and deliverables are summarised in the final section.

Publication list

- E Anderlini, G Salavasidis, CA Harris, P Wu, A Lorenzo, AB Phillips, G Thomas, "A Remote Anomaly Detection System for Slocum Underwater Gliders", *Ocean Engineering*, 236, 109531, 2021. DOI: 10.1016/j.oceaneng.2021.109531
- P Wu, CA Harris, G Salavasidis, A Lorenzo, I Kamarudzaman, AB Phillips, G Thomas, and E Anderlini, "Unsupervised Anomaly Detection for Underwater Gliders Using Generative Adversarial Networks", *Engineering Applications of Artificial Intelligence*, 104, 104379, 2021. DOI: 10.1016/j.engappai.2021.104379.
- P Wu, CA Harris, G Salavasidis, I Kamarudzaman, AB Phillips, G Thomas, and E Anderlini, "Anomaly Detection and Fault Diagnostics for Underwater Gliders Using Deep Learning", *IEEE Oceans 2021*, accepted.
- Z Bedja-Johnson, P Wu, D Grande, E Anderlini, "Smart Anomaly Detection for Slocum Underwater Gliders with a Variational Autoencoder with Long Short-Term Memory Networks". *Applied Ocean Research,* 120, 103030, 2022*,* DOI: 10.1016/j.apor.2021.103030.

Contents

List of Figures

List of Tables

List of Acronyms

ALADDIN	Assuring Long-term Autonomy through Detection and Diagnosis of Irregularities in Normal operation
ALI-GAN	Adversarial Learned Inference GAN
ALR	AutoSUB Long-Range
APF	Automated Piloting Framework
ASVs	Autonomous Surface Vessels
AUVs	Autonomous Underwater Vehicles
BiGAN	Bidirectional Generative Adversarial Networks
BODC	British Oceanographic Data Centre
BoK	Body of Knowledge
BOM	Blue Ocean Monitoring
C2	Command & Control infrastructure for over-the-horizon operation of marine autonomous systems
CI/CD	Continuous Integration and Continuous Delivery
CTD	Conductivity, Temperature and Density
EGO	Everyone's Gliding Observatory
FMEA	Failure Mode and Effect Analysis
FOAM	Forecast Ocean Assimilated Model
FRONTIERS	Fault detection, isolation and Recovery fOr uNderwaTer glIdERS
GAN	Generative Adversarial Networks
GPS	Global Positioning System
IFAC	International Federation of Automatic Control
IMO	International Maritime Organization
IMOS	Integrated Marine Observing System
ISO	International Organization for Standardisation
LAUV	Light Autonomous Underwater Vehicle
LR	Lloyd's Register
MARS	Marine Autonomous and Robotic Systems
MAS	Marine Autonomous Systems
MASNet	Marine Autonomous System Net
MASS	Maritime Autonomous Surface Ships
MCA	Maritime and Coastguard Agency
ML	Machine Learning
NERC	Natural Environment Research Council
NOC	National Oceanography Centre
NRT	Near Real-Time
OMG	Ocean Microstructure Gliders
PAP	Porcupine Abyssal Plain
PLOCAN	Oceanic Platform of the Canary Islands
QC	Quality Control
RAS	Robotics and Autonomous Systems
ROVs	remotely operated vehicles
SAE	Society of Automotive Engineers
SAMS	Scottish Association of Marine Science

SOCIB	Balearic Islands Coastal Ocean Observing and Forecasting System
TRL	Technology Readiness Level
UCL	University College London
UG	Underwater Glider
UMS	Unmanned Marine Systems
UMV	Unmanned Marine Vehicles
UNCLOS	United Nations Convention of the Law at Sea
VAE	Variational Auto-Encoder
VBD	Variable Buoyancy Device
WP	Work Package
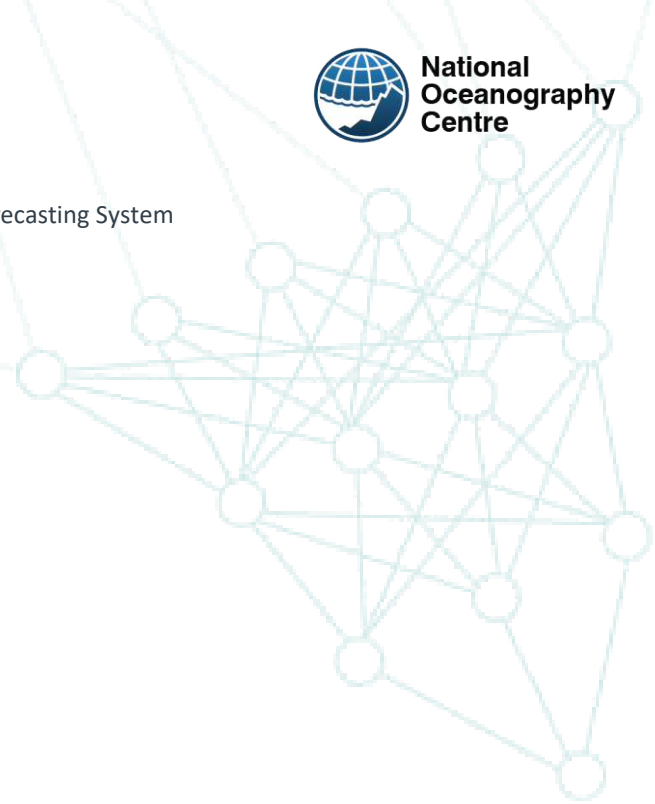
1. Requirements Definition

1.1. Introduction

The Assuring Autonomy International Programme (AAIP) has the ambitious goal of delivering a Body of Knowledge (BoK) for the assurance of Robotics and Autonomous Systems (RAS). The content of the BoK must be comprehensive, cross-domain, cross-technology and cross-application. The guidance should be as general as possible, but providing domain-specific guidance when required and use established assurance approaches.

The BoK will cover four main areas to support core principles for developing, assuring and regulating RAS:

- Defining the required behaviour - Defining what it means for the RAS to be 'safe';
- Implementing a RAS to provide the required behaviour – Demonstrating the sufficiency of the implementation;
- Understanding and controlling deviations from required behaviour – Identifying and controlling sources of deviation;
- Gaining approval for operation of RAS – Gaining approval for operation in the specified environment from the relevant regulatory authority.

The demonstrator project "Assuring Long-term Autonomy through Detection and Diagnosis of Irregularities in Normal operation (ALADDIN)" will contribute to the drafting of the following sections of the BoK:

- 2.2.1.1 – Defining sensing requirements,
- 2.2.1.2 – Defining understanding requirements,
- 2.2.4.1 – Verification of sensing requirements,
- 2.2.4.2 – Verification of understanding requirements,
- 3.1.1 – Identifying sensing deviations

As a result of the focus on sensing and understanding requirements and deviations, project ALADDIN's expected greatest contribution will be in the field of smart fault diagnostics for RAS.

Thanks to the project partners and stakeholders' expertise, the case study of Marine Autonomous Systems (MAS) will be used to contribute to the BoK, with the verification possible thanks to field tests of real Autonomous Underwater Vehicles (AUVs). In particular, the main RAS that will be used in the project is the new common Command-Control (C2) and data processing system for the over-the-horizon operation of MAS that is being developed by the National Oceanography Centre (NOC) as part of the NERC/ISCF-funded project Oceanids[1].

The first Work Package (WP) is instrumental in collating the different information required by further WPs. The aim of WP1 was to define healthy and anomalous behaviour of RAS with appropriate vocabulary, identify data availability and sources for RAS deployments, with a focus on MAS, and outline the project steering direction for the following WPs. Each of these

---

[1] https://www.noc.ac.uk/projects/oceanids

points is covered in the following sections. Section 1.2 introduces the terminology for the condition monitoring of MAS. Section 1.3. covers the sources of the remote sensing and RAS deployment datasets that can be used in the remainder of the project. Section 1.4 presents a brief overview of state-of-the-art fault diagnostics methods, including a research gap. Section 1.5 describes the selected steering direction for the project. Section 1.6 summarises the conclusions of WP1 and presents the outline of WPs 2-4.

1.2. Definition of the Conditional Status of Robotic and Autonomous Systems

Vocabulary

The formal definition of a vocabulary to describe healthy and anomalous operating conditions for RAS is critical to introduce sensing and understanding frameworks for fault diagnostics that are general, transferable and technology agnostic. Rather than creating a completely new vocabulary, suitable existing terminology has been identified from a number of sources, including regulators and quality control systems, mostly from sectors other than maritime autonomy.

Regulators – Marine Autonomous Systems

Firstly, not only does no accepted vocabulary currently exist for the condition monitoring of MAS, but even accepted terminology to refer to them is currently lacking. In the literature, MAS are also referred to as Unmanned Marine Systems (UMS) or Unmanned Marine Vehicles (UMV). Until general regulations are developed for MAS in general by the International Maritime Organization (IMO), which will then be accepted as the standard globally, the acronym "MAS", which is the standard for the Natural Environment Research Council in the UK, will be adopted in project ALADDIN.

The Maritime Safety Committee of the IMO is currently running a scoping exercise to draft new regulations for the operation of Maritime Autonomous Surface Ships (MASS) [1]. From the draft, it is clear that the command and control infrastructure that is currently being developed at the NOC for the over-the-horizon operation of their fleet of MAS (in itself a RAS with human-in-the-loop control possible) will need to be referred to as a Remote Control Centre (RCC) once the IMO vocabulary is finalised and accepted.

Because of the present lack of regulations, the fit of MAS within the legal framework is not clear. Many MAS may not be considered to be included in the definition of "ship"; hence, they may not enjoy states' rights of navigation under the United Nations Convention of the Law at Sea (UNCLOS) [2]. Veal, at al. [2] suggest that the international legal framework delegates the decision of whether a MAS is a ship or not to the flag state's national laws, with the outcome binding on other states. The MAS falling under the definition of ship will need to comply with the current regulatory framework for shipping, with compliance becoming harder with increasing level of autonomy. The MAS not falling under the definition of ship will have navigational rights in the areas beyond national jurisdiction, but there are doubts about their navigational rights in the jurisdictional zones of other states.

Classification societies, which provide rules and standards for the design and construction of complex engineering systems like ships and then certify them as safe, have started to develop

regulations specific to MAS, anticipating the IMO who have only recently started to meet to propose new rules. As AUVs have been available for the past thirty years, there are more sets of rules for their classification at present, e.g. DNV-GL's Underwater Technology, Part 5, Chapter 8 [3]. The regulations define the documentation, survey and testing requirements, the principles for the design and construction of AUVs and control and automation philosophy. The last section is particularly interesting as it requires pre-planned missions to be entered (currently by human operators) and stored and the parameters needed for the operation of the vehicle monitored and a failure report sent to the operator as well as saved on board in case the transmission is not possible. If the failures cannot be corrected, the mission will need to be aborted. The tools and sensors that should be used for the data connection and navigation are also described. However, the regulation on the remote and on-board condition monitoring of components are not specific and detailed.

Lloyd's Register, a global classification society and part of the project steering committee, developed the first design code for MAS in 2017 [4]. The code defines autonomy levels for their operation that range from manual, level 0, to fully autonomous decisions, level 6. As part of the nomenclature, the definition of "Reasonably Foreseeable Operating Conditions" is particularly interesting for project ALADDIN, as MAS' adverse behaviour due to extreme environmental disturbances, e.g. strong ocean eddies or currents for the smaller MAS, may fall outside of this definition. Since rules specific to condition-based maintenance and fault-tolerant RAS behaviour are not yet covered as part of the code, the outcomes of project ALADDIN have the potential to contribute to future versions of the document.

Regulators – Condition Monitoring

Vocabulary specific to condition monitoring are published by the International Federation of Automatic Control (IFAC) and the International Organization for Standardisation (ISO) standards.

The condition monitoring terminology for technical processes for the IFAC was first introduced by the Safety Process Technical Committee and reported in [5]. Identical vocabulary with the addition of some terms specific to machinery and road vehicles can be found in the ISO standards "13372:2012(en): Condition monitoring and diagnostics of machines" [6] and "ISO 26262-1:2018(en): Road vehicles — Functional safety" [7], respectively. According to this vocabulary, we can define the following states and signals:

- *Standard conditions* or the *baseline*, indicating the system operating as intentionally designed;
- *Anomaly* or *abnormality* as a deviation from standard conditions;
- *Malfunction* as an intermittent irregularity in the fulfilment of a system's desired function;
- *Error* as the deviation between a measured or computed value (of an output variable) and the true, specified or theoretically correct value;
- *Disturbance* as an unknown (and uncontrolled) input acting on a system;
- *Perturbation* as an input acting on a system, which results in a temporary departure from the current state, i.e. the parametric and dynamic uncertainty in the model of the system;

- *Fault* as an unpermitted deviation of at least one characteristic property or parameter of the system from the acceptable / usual / standard condition;
- *Failure* as the permanent interruption of a system's ability to perform a required function under specified operating conditions;
- *Failure mode* as the observable manifestation of a system fault;
- *Fault progression* as the characterization of the change in the observability of a fault over time;
- *Sign* as a characteristic parameter of a signal, which shows information about a state;
- *Syndrome* as a group of signs or symptoms that collectively indicate or characterize an abnormal condition.

Fault types can be found and classified during the design of a RAS through a Failure Mode and Effects Analysis (FMEA), which describes the process of reviewing as many components, assemblies, and subsystems as possible to identify potential failure modes in a system and their causes and effects.

Additionally, building on the IFAC and ISO frameworks and taking inspiration from the medical field and the Aeronautical Design Standard (ADS), Society of Automotive Engineers (SAE) standards, Lloyds Register has adopted the following vocabulary to describe functions for digital twins used in asset health management [8].

Anomaly detection differentiates between nominal and abnormal operating conditions. The portion of operational data (time series data) indicative of the anomaly will be further analysed to determine the existence and narrow the type of faults and failure condition. Hence, this requires the definition of a *baseline*, i.e. a descriptor or group of descriptors which provides a criterion of the normal behaviour of the system under various process states.

Fault detection refers to the algorithmic processing that builds on the results of anomaly detection to evaluate and identify the specific failure mode (i.e. fault is an incipient failure), the affected replaceable part(s) and its estimated severity (i.e. level of degradation/damage experienced by the affected part).

Diagnostics describes the concurrent processing and analysis of several faults, degradation in progress, including reasoning at the sub-assembly, equipment, sub-system, system level to provide an integrated health insight. Hence, fault diagnostics follows fault detection and includes *fault isolation*, i.e. the determination of the kind, location and time of detection of a fault., and *identification*, i.e. the determination of the size and time-variant behaviour of a fault.

Prognostics involves predicting the time progression of a specific failure mode from its incipience (i.e. a fault) to the estimated time of the part's failure. Given that no part exists in isolation and that there is a multitude of influences at the sub-assembly, equipment, system, asset level, including operational, maintenance i.e. real-world factors that can be technically challenging to represent, prognostics will include the abstraction and analysis of the following

1. existing failure modes and their effect on the part deterioration rate;
2. representation of different stressors (i.e. factors that drive the failure mode);
3. knowledge and initiation criteria of future failure modes;

4. interrelationship between failure modes and their effect on the part deterioration rates;
5. the effects of servicing and maintenance on part degradation;
6. some form of uncertainty representation and management – meaning the knowledge of the conditions (context) and assumptions underpinning the prognosis.

The last point is important given that prognostics deals with the ability to generate insights on damage that is yet to happen.

Target granularity. The more detailed (granular) an asset can be represented, the better clarity on the maintenance and operational decisions. Ideally the granularity should be directed at the "replaceable parts level" where inspections, servicing and maintenance (replacement) actions are performed. A higher level of abstraction i.e. parts sub-assembly, equipment level, sub-system level will be useful but will require some degree of disambiguation (investigation) from the crew and engineers in defining the maintenance/operational actions.

Failure Coverage. Different replaceable parts, structural components and elementary functions will be susceptible to different, unique failure modes that dictate their consequent corrective and maintenance actions. Knowing the exact failure coverage of a digital twin enables the estimation of its value on both failure avoidance (i.e. cost saving by avoiding a real-world failure) and maintenance/servicing extension (i.e. cost deferment of inspections and overhauls given the absence of failure condition).

It is clear that the scope of project ALADDIN will be limited to fault detection and diagnostics, i.e. describing and verifying the sensing and understanding requirements and deviations for RAS. Specific vocabulary for failure modes of underwater gliders, which will represent the main case study of the project, can be found in [9]. For other RAS, these will need to be replaced with technology-specific failure modes as obtained from the FMEA by the designer.

Quality Control Systems – Scientific Community

Quality Control (QC) systems are already being extensively used by the oceanography community to characterise possible errors or problems with the measured scientific data. A comparison of different oceanographic quality flag schemas can be found in the Ocean Data View[2]. Additionally, the Natural Environment Research Council (NERC) have developed a dedicated QC vocabulary, which includes also operational phases for underwater gliders[3]. Similar solutions have been implemented by other operators, with the QC manual for Argo floats being extremely interesting, as it includes common instrument failure types[4]. This terminology has been used as the basis of reliability studies for underwater gliders [9], where specific vocabulary for faults on AUVs is introduced.

---

[2] https://odv.awi.de/documentation/

[3] http://vocab.nerc.ac.uk/collection/S28/current/all/

[4] https://archimer.ifremer.fr/doc/00228/33951/32470.pdf, Appendix 4.1

QC flags are also used in fields other than oceanography. For instance, the satellite data provided by Copernicus is classified in 44 categories to characterise the type of land cover in the CORINE Land Cover (CLC) Inventory[5,6]. Similarly, 19 classes are used in the BigEarthData training set[7], although everything seaward of the lowest tide limit is simply classified as "Sea and ocean" class 52.

RAS Adverse Behaviour Labelling Policy

Metadata

Including the status of RAS conditions as metadata, i.e. data providing information on other data, is desirable, as it can help operators identify anomalous behaviour and improve fault diagnostics solutions, whilst informing end users of the collected data of potential problems. A particularly interesting example is cruise reports for oceanographic missions, e.g. as shown on the British Oceanographic Data Centre (BODC) portal[8], where the reports contain a very brief summary of main events with a time stamp.

Levels of Annotation

The metadata annotation of healthy or anomalous operating conditions would need to span four levels:

1. The upmost level would consist in a classification of a whole deployment as either presenting completely normal conditions or some adverse behaviour for the RAS. This would facilitate the use of the dataset for further training, validation or test of fault diagnostics methods.
2. The broad annotation can be further refined on a lower level of local segmentations of phases of the deployment, e.g. a number of dives for AUVs. This would enable a quicker recognition of which portions of the dataset to use for training and which to use for testing purposes.
3. It is then possible to annotate individual phases of the deployment, e.g. individual dives for AUVs, as presenting either normal or adverse operating conditions.
4. Finally, individual operational phases can also be annotated in detail to characterise the data on the lowest level of time steps.

1.3. Sources for the Sensing Data of Marine Autonomous Systems

Vehicle Data

Data from real RAS deployments is fundamental to train, validate and test fault diagnostics algorithms. Project ALADDIN will focus on the case study of MAS. Hence, here is a list of

---

[5] https://land.copernicus.eu/pan-european/corine-land-cover

[6] https://land.copernicus.eu/user-corner/technical-library/copy_of_Nomenclature.pdf

[7] http://bigearth.net/#

[8] https://www.bodc.ac.uk/resources/inventories/cruise_inventory/reports/jc152.pdf

publicly available repositories with data from MAS deployments, inclusive of the engineering data needed to track the RAS' state.

British Oceanographic Data Centre

The BODC contains one of the largest data repositories for deployments of scientific AUVs and small MASS[9]. In particular, the data from some projects, like AlterEco[10], ElletLine[11] and MASSMO[12], includes data from multiple MAS operating in the same area, which is particularly interesting for studies involving multiple RAS.

The dataset includes data from the following types of MAS:

- Seaglider, Deepglider and Slocum underwater gliders in a range of versions and sensory payloads,
- Autonaut small MASS,
- AutoSUB Long-Range (ALR), hybrid long-range AUVs.

Integrated Marine Observing System

Australia's Integrated Marine Observing System (IMOS) contains data of deployments of underwater gliders around Australia's waters[13]. Although the datasets that are published on the website do not include the engineering data, it is possible to contact the IMOS office for the information, which will readily share engineering data for the requested deployments.

IFREMER

France's IFREMER includes a very similar data portal to the BODC and IMOS, although the information is in French[14].

PLOCAN and SOCIB

Spain's PLOCAN and SOCIB have very similar data portals that are easily accessible[15,16]. The datasets are particularly interesting because they refer to deployments in warm, tropical waters, where biofouling is a stronger problem.

Although there are other national and regional datasets for oceanographic deployments, the BODC, IMOS, IFREMER, PLOCAN and SOCIB's are representative examples.

---

[9] https://www.bodc.ac.uk

[10] https://projects.noc.ac.uk/altereco/

[11] https://projects.noc.ac.uk/ExtendedEllettLine/frontpage

[12] https://projects.noc.ac.uk/massmo/frontpage

[13] https://imos.org.au/data

[14] http://data.ifremer.fr

[15] http://obsplatforms.plocan.eu

[16] http://thredds.socib.es/thredds/catalog/auv/glider/catalog.html

EGO-network

The Everyone's Gliding Observatory (EGO) network is a global database of publicly available deployment data from underwater gliders[17]. Although only scientific data in netCDF format is readily available, the engineering data can be obtained by contacting the principal investigator of each deployment.

Datasets of MASS

At present, only prototype MASS have been developed. At the lower Technology Readiness Level (TRL) scale, research institutions have been experimenting with increasing levels of technology on existing platforms, e.g. by the Norwegian Technical University's (NTNU) Cyber-Physical Systems Laboratory[18,19]. At the higher TRL scale, Promare and IBM have been developing a fully autonomous, solar-powered sailing boat, the Mayflower, to transverse the oceans[20] and Ocean Infinity has been developing a fleet of 15 MASS for the monitoring and mapping of the oceans[21]. However, developers of small MASS for defence applications, like L3Harris-ASV[22], have developed most vehicles to date.

Nevertheless, no dataset for MASS deployments is currently available, other than for the smallest wave-powered Autonaut and Wavegliders for oceanographic applications.

Complementary Remote-Sensing Data

Satellite Observations

Copernicus is the European Union's Earth Observation Programme, combining data from satellite, ground-based, airborne and seaborne measurement systems[23]. The information services provided are open and freely available to all users. The system has a dedicated service specific for marine data and products[24], e.g. assimilated models including Forecast Ocean Assimilated Model (FOAM) and Atlantic Margin Model (AMM) 15[25]. The satellite data provides altimetry, chlorophyll, sea surface temperature, resolving mesoscale features such as eddies and fronts.

---

[17] https://www.ego-network.org/dokuwiki/doku.php

[18] https://www.ntnu.no/blogger/cpslab/autonomous-ships/

[19] https://www.ntnu.edu/autoferry

[20] https://mas400.com

[21] https://oceaninfinity.com

[22] https://www.asvglobal.com

[23] https://www.copernicus.eu/en/access-data

[24] https://marine.copernicus.eu/

[25] https://resources.marine.copernicus.eu/?option=com_csw&view=details&product_id=NORTHWESTSHELF_ANALYSIS_FORECAST_PHYS_004_001_b

Additionally, polar data and algorithms[26] have been developed as part of the **Extreme Earth** EU project[27]. The data includes a very large training set of satellite data, labelled with ice types in both the European Arctic and Greenland[28]. The aim is to produce maps showing different ice types for maritime users.

Seafloor and Water Column Observations

In addition to underwater gliders, the water column of the oceans is observed through a global network of profiling floats, the Argo floats[29]. There are many publicly available repositories for seafloor and water column data sets:

- **EMODnet**[30], with portals that include the bathymetry and physics of the ocean column[31], with the data likely to be most useful in relation to underwater gliders.
- The European Multidisciplinary Seafloor and water column Observatory (**EMSO**)[32] includes data from regional mooring and fixed platform observatories around Europe, from North East Atlantic, through the Mediterranean, to the Black Sea. The NOC operate the Porcupine Abyssal Plain (PAP) mooring[33].
- The European **JERICO** project provides free access to European coastal data[34].
- **SeaDataNet**[35] contains data and vocabularies that are split into three sets: navigation, hydrography and metrology, and bathymetry.
- **ConnectingGEO**[36] links the observation data with the scientific and technology communities.

The **Google Dataset Search**[37] feature provides multiple results for Conductivity, Temperature and Density (CTD) data, glider data and data from different types of data acquisition systems.

---

[26] https://portal.polartep.io/ssoportal/pages/login.jsf

[27] http://earthanalytics.eu/about.html

[28] https://drive.google.com/file/d/1QYpekbD69V_9tK3YI1aOSdP0VV5gKWhQ/view

[29] https://argo.ucsd.edu/

[30] https://www.emodnet.eu/en/portals

[31] http://www.emodnet-physics.eu/Map/

[32] http://emso.eu/#

[33] http://data.emso.eu/

[34] https://www.jerico-ri.eu/

[35] https://www.seadatanet.org/

[36] http://www.connectingeo.net/

[37] https://datasetsearch.research.google.com/

1.4. Overview of Fault Diagnostics Methods

Fault diagnostics and prognostics methods can be classified as rule- or knowledge-based, model-based or data-driven [10].

Rule-Based Methods

Rule-based diagnostics are an approach that relies on bespoke heuristics, usually in the form of if-then statements, obtained from designers' observations of the system [11]. Hence, they tend to be very problem-specific (and thus with poor generalisation), require significant effort from the designers and scale poorly to large problems.

Rule-based methods have been used for years for the fault diagnostics of engineering systems. The authors successfully designed and implemented a rule-based solution for the detection and identification of wing loss on underwater gliders [12]. However, project ALADDIN will focus on other approaches.

Model-Based Methods

Model-based diagnostics use a model constructed from in-depth knowledge of the system dynamics for fault detection, isolation and identification, typically by analysing the residuals between the actual and estimated values of each sensor. Model-based solutions are standard condition monitoring tools on a large range of engineering systems, e.g. aircraft [13]. The authors successfully introduced an effective model-based system for the detection and isolation of the loss of one wing [12] and high levels of biofouling [14] and on underwater gliders.

Over the past two decades, model-based fault diagnostics methods have evolved into the development of digital twins, i.e. the software representation of a physical asset, system or process designed to detect, prevent, predict, and optimise performance through real-time analytics to deliver business value, for improved condition monitoring of complex engineering systems, such as ships [15]. The concept of digital twin can have important implications for the condition monitoring of complex RAS.

Data-Driven Methods

Building on the recent advances in machine learning, data-driven methods for fault diagnostics and prognostics have been gaining in popularity thanks to their generality and scalability to even the largest problems (in fact, accuracy improves with increasing dataset size). Amongst the many reviews of data-driven solutions for intelligent maintenance systems, we refer the reader to Ellefsen, et al. [16], who focus on the application to MAS, and three more general, but more detailed studies [17]–[19].

From these studies, it is possible to differentiate between first-generation machine learning solutions, such as support vector machines, Gaussian processes and classification and regression trees, to second-generation deep learning methods, based on neural networks with many deep layers. Since approximately 2015, deep learning has outperformed other classical machine learning solutions for complex classification and regression tasks, as such voice and pattern recognition, thanks to its ability to extract useful features from and to scale

efficiently to extremely large datasets. Hence, here is a list of the deep learning methods that have been used for fault diagnostics and prognostics tasks to date.

- **Restricted Boltzmann Machine**, an undirected bipartite graphical model used as generative models which learn a reconstructed version of the input data through stochastic processing units. Two typical deep learning examples are
  - **Deep belief networks**,
  - **Deep Boltzmann machines**.
- **Autoencoder**, unsupervised networks that are trained to reconstruct the input on the output layer through two stages: an encoder learns a hidden representation of the data via feature extraction and then a decoder maps the hidden representation back to the input space to obtain a reconstruction of the data. A problem with standard autoencoders is the tendency to learn identity functions without extracting meaningful information about the data. Hence, the following alternative variants have been proposed to improve performance and solve the issue:
  - **Spare autoencoder**, which include sparsity constraints to improve the classification task performance by increasing the likelihood that different categories will be easily separable.
  - **Denoising autoencoder**, which uses Binary or Gaussian noise to corrupt the input for regularisation to prevent the network from learning a trivial solution.
  - **Contractive autoencoder**, which encourage the robustness of the representation by penalizing the sensitivity of the features rather than regularizing the reconstruction.
  - **Variational autoencoder**, which are directed generative models that use variational inference framework to approximate the input data distribution. Variational autoencoders enable the design of generative models for large complex datasets, bridging the gap between deep learning and probability models.
- **Convolutional neural network**, deep discriminative networks with convolutional or parameter-sharing layers that have shown excellent results in processing data with grid-like topology, e.g. images but also time series data.
- **Recurrent neural network**, which have feedback loops to remember the information of former units and are the most suitable for sequential data such as time-series applications.
- **Generative Adversarial Networks** (GAN), which are a powerful generative model consisting of two neural networks: a discriminator and a generator. The generator learns the distribution of the inputs and creates the fake data, whilst the discriminator analyses both fake and real data and evaluates them for authenticity. Performance improvements are achieved during training as the generator and discriminator learn to outsmart each other.

New algorithms are constantly being developed that are combinations of these strategies.

Research Gap

Fink, et al. [10] identify five main promising directions for the application of deep learning to fault diagnostics, prognostics and health management:

- **Transfer learning**, which comprises methods to transfer knowledge from one domain to another to improve accuracy and performance, e.g. from the training to the test data, from simulations and experiments to the full-scale prototype of the RAS, from one well-understood machine to another in the fleet. In particular, interesting research questions include:
  - "How do we deal with the case where input and output space of the source and target domain are not identical?"
  - "How can we efficiently encode the input data to make them more transferable?"
  - "How do we deal with the quantity imbalance between healthy and fault data?"
- **Deep learning for fleet approaches**, which incorporate techniques to learn and extract relevant informative patterns across an entire fleet of RAS from the fault patterns of the individual devices, considering the high variability of the system configurations and the dissimilarity of the operating conditions. The aim of these solutions is to maximise the available data and information for improved performance, as anomalies are rare events. The approaches that have been proposed for the monitoring and fleets of systems, e.g. RAS, include:
  - Clustering the fleet in sub-fleets based on characteristic parameters, e.g. average operating regimes. The downside is that the aggregated parameters may not be representative of the unit's specificities or capture all relevant conditions.
  - Using the entire time series of condition monitoring signals to perform time series cluster analysis of units. However, this is affected by the curse of dimensionality and time series cluster analysis becomes even more challenging when operating conditions evolve over time.
  - Developing models for the functional behaviour of the individual units and identifying similar devices following this learned functional behaviour. A requirement is for units to experience sufficient similarity in their operating regimes; hence, large fleets may be required to identify systems with sufficient similarity.
  - Performing domain alignment, or transfer learning, in the feature space of the different units to compensate for the distribution shift between the individual devices of the fleet. However, there are no guarantees that the system will behave in a similar way in the future.
- **Generative algorithms**, which learn a probabilistic distribution without any assumption of the induced family distribution. Hence, they can be used to enhance the training set to solve bias problems caused by the rarity of faults. As the technology is still new, improvements can be made by changing the structure of the algorithms.
- **Reinforcement learning**, which is a decision-making framework and seems a promising method to improve the planning policy for condition-based maintenance.

Physics-induced machine learning, a promising technique to induce interpretability in the machine learning models. Prior knowledge is integrated in the models, e.g. through dynamic models to generate virtual sensors [20] or generative methods [21], delivering improvements in terms of performance as well as interpretability. As a result, the amount of required data samples for training is reduced and the performance of the learning algorithms improved.

1.5. Project Steering Direction

Based on the literature review described in the previous sections and the meetings with the steering committee, the selected project direction is described in the following sections.

Selected Terminology and Procedure for the Annotation of RAS Adverse Behaviour

In project ALADDIN, the ISO terminology is adopted to describe the condition monitoring processes, whilst the vocabulary from the draft IMO regulations is used to define MAS. In particular, the scope of project ALADDIN will be limited to the introduction of novel methods for the fault detection and diagnostics of RAS verifying the procedures with the case study of MAS, leading to the formulation of protocols for the assurance of the operation of RAS.

Although RAS failure mode and fault progression types can be described through a FMEA, the information may be commercially sensitive and unavailable to operators and tends to be very system specific. Hence, in project ALADDIN, the challenging problem of *open-set fault diagnostics* is addressed, i.e. the problem where the knowledge of the faulty system is incomplete during training and the number and extent of the faults, of different types, can evolve during operation [20], [21]. This will be addressed through the formulation of fault diagnostics methods based on unsupervised learning, which will be trained with data points of standard operating conditions, the *baseline*, and unlabelled data points, which may also contain data representative of faulty conditions, although the number of faults is not known. The system will then be used to autonomously label all available data for MAS deployments as either standard or anomalous and, if anomalous, which class it belongs to. Afterwards, the designer can manually set a representative name for the class based on the actual physical fault type from the general FMEA of the RAS type, so that the data can be used in further supervised learning approaches, which may be more accurate and efficient.

Therefore, three levels are specified for the data labelling, corresponding to three additional columns in the data set:

1. The Boolean labels "standard" (i.e. like the baseline) or "anomalous.
2. The categorical labels "standard", one of the existing classes that are specific to the analysed RAS type, e.g. "leak", "power/battery failure", "buoyancy pump failure", etc. for underwater gliders. If a new class is created automatically by the smart fault diagnostics system, e.g. if the human designer still needs to analyse the category, the classes are labelled sequentially as "class 1", "class 2", etc.
3. The Boolean labels "checked" or "not checked" to define if the data point has been checked as actually belonging to the correct category. This is a fundamental step to ensure the quality of the data used for the training of fault diagnostics systems, as the user may overwrite the values in the other columns.

The last step is fundamental to provide feedback to the system and remove any errors in detection and identification of anomalies. In the future, the system should be complemented with a mapping of the fault types as returned by the categorical labels to the exact subsystems and failure modes to aid the maintenance or replacement of components in a condition-monitoring intelligent maintenance framework.

Additionally, metadata using the timestamp will be used to capture the four levels of annotation described in Section 1.2. The metadata formatting will be based on the ISO 13374

standards on the "Condition monitoring and diagnostics of machines — Data processing, communication and presentation" [22].

For underwater gliders, the general failure mode identified in [9] are adopted in project ALADDIN and are reported in Table 1.1 for clarity. These will be complemented with the individual equipment manufacturer manuals, e.g. [23], [24]. In general, the manifestation of the failure will be functional before showing explicit physical damage.

Note that ideally a rigorous data labelling procedure is put in place before the operations of the RAS. However, as this was not done in the past with the operation of MAS, project ALADDIN will make up for it with the proposed labelling strategy, which is based on the philosophy of simple but meaningful labels.

Table 1.1: Failure modes of underwater gliders – modified from [9]*.*

Failure Mode	Subsystem
Leak	Pressure hull
Power/Battery failure	Battery or power electronics
Buoyancy pump failure	Variable buoyancy device
Air bladder leak	Variable buoyancy device
Oil bladder leak	Variable buoyancy device
Collision with a vessel	Pressure hull, appendages
Collision with the seabed	Pressure hull, appendages
Glider recovered by a fishing boat	GPS
Science sensor failure	Sensor bay
Roll tilt sensor failure	Navigation sensors
Pitch tilt sensor failure	Navigation sensors
Compass failure	Navigation sensors
Iridium communications failure	Communication sensors
Argos failure	Communication sensors
GPS sensor failure	Communication sensors
Data logging failure	On-board software
Command/Control software failure	Shore-based software
On-board software failure	On-board software
Attitude (pitch) control failure	Pitch control
Roll motor failure	Heading control
Rudder broken	Heading control
Fin locked at fixed position	Heading control
Digfin not working properly	Heading control
Unknown	Unknown

For the ALR, the failure modes will be complemented by the FMEA performed by the NOC.

Selected Data Sources

Due to the ready availability of data, the deployments of underwear gliders will be used as a case study for the definition and verification of sensing and understanding procedures for RAS in project ALADDIN. In particular, for the development, training, validation and initial testing of the fault diagnostics methods, available deployment data from MAS will be employed.

Underwater Gliders

Firstly, the data from ten deployments of G2 Slocum underwater gliders [25], [26] operated by the NOC will be analysed (Table 1.2), as anomalous behaviour has already been studied for this data set.

Table 1.2: Deployments of Slocum G2 underwater gliders operated by the NOC, inclusive of glider identification number, date and project of the deployment, duration and registered RAS status. The deployments were made as part of project AtlantOS (EC/633211), CaNDyFloSS (NE/K001701/1) and ALTERECO (NE/P013902/2). OMG refers to Ocean Microstructure Glider, i.e. a glider with a very large sensory package that causes a significant increase in drag.

No.	Glider ID	Date	Project	Duration [days]	RAS status
1	345	2014	AtlantOS, CaNDyFloSS	123.9	Healthy
2	397	2015	AtlantOS, CaNDyFloSS	45.9	Healthy
3	399	2015	AtlantOS, CaNDyFloSS	84.6	Possible biofouling
4	419	2015	AtlantOS, CaNDyFloSS	11.0	Healthy
5	423	2015	AtlantOS, CaNDyFloSS	6.8	OMG
6	424	2015	AtlantOS, CaNDyFloSS	20.8	OMG
7	194	2017	ALTERECO	83.9	Angle of list
8	304	2019	ALTERECO	76.9	Loss of right wing
9	345	2019	ALTERECO	76.8	Strong disturbances
10	436	2019	ALTERECO	89.8	Loss of left wing

Additionally, through UCL's project "Identification of the Dynamics of Underwater Gliders (IDUG)" as part of the umbrella project EUMarineRobots (EC/731103), the dataset for Slocum G2 gliders is complemented by an additional two deployments operated by PLOCAN with simulated and observed high levels of marine growth or biofouling in Table 1.3.

Table 1.3: Deployments of Slocum G2 underwater gliders operated by PLOCAN, inclusive of glider identification number, date and project of the deployment, duration and registered RAS status.

No.	Glider ID	Date	Project	Duration [days]	RAS status
11	492	2014	IDUG	9.5	Simulated high levels of biofouling
12	492	2015	IDUG	20.5	Growing natural levels of biofouling

To investigate the ability of the procedures to generalise to different MAS technologies, the dataset is augmented with deployments of Seaglider underwater gliders, a different type of vehicle [27], as studied in [28]. In particular, six deployments operated by the NOC jointly with the Scottish Association of Marine Science (SAMS) and IMOS are considered, as shown in Table 1.4.

Table 1.4: Deployments of Seaglider underwater gliders operated by NOC/SAMS and IMOS, inclusive of glider identification number, date, duration and registered RAS status.

No.	Glider ID	Operator	Duration [days]	RAS status
13	sg545	NOC/SAMS	16.8	Healthy
14	sg532	NOC/SAMS	176.9	Healthy
15	sg550	NOC/SAMS	44.1	Healthy
16	sg616	NOC/SAMS	165.9	Healthy
17	sg603	NOC/SAMS	175.6	Healthy
18	sg602	NOC/SAMS	143.8	Healthy
19	sg153	IMOS	33.7	Healthy
20	sg516	IMOS	91.3	Healthy
21	sg514	IMOS	103.4	Possible biofouling
22	sg516	IMOS	66.9	Healthy
23	sg540	IMOS	36.4	Healthy
24	sg514	IMOS	107.9	Possible biofouling

Additionally, these datasets can be complemented by the data collected by other deployments from project AlterEco 1-7, and by project MASSMO 5, consisting of Seaglider

and Slocum underwater gliders, which are publicly available on the BODC website[38]. These missions are particularly interesting, as they comprise data from multiple vehicles operating in the same area so that fleet management solutions can be investigated.

Furthermore, if there is sufficient time, the deployments of the ALR [29], e.g. in Loch Ness, can also be analysed to verify the transferability of the methods to more complex RAS platforms than underwater gliders.

For all datasets of recent projects operated by the NOC, both the recovery and decimated datasets are available. The data on-board the vehicles, which can be recovered in full at the end of a deployment, is sent ashore to the over-the-horizon command and control infrastructure in a decimated format to reduce the transmission time, costs and power expenditure. The recovery data can be freely downloaded from the BODC portal, whilst the recovery data is available through the NOC's C2.

Selected Areas of Research

From the identified research gaps, two main areas of research are selected as most promising for project ALADDIN to make the greatest contribution to the state-of-the-art fault diagnostics research and industrial practices:

- Multi-platform sensing: sensors may sometimes present drifts or be mis-calibrated. Understanding which sensors are affected can be challenging on complex RAS. A new research direction involves using the data from multiple RAS in the same operational area in addition to historical environmental disturbance data to correctly pinpoint the damaged sensor.
- Fault diagnosis: deep learning methods for fleet management. As faults are rare, the idea is to exploit information from similar systems. However, different operational conditions, system configuration and even types make the problem extremely challenging. Domain adaption solutions seem most appropriate, but require large datasets of operational data.

In project ALADDIN, we can exploit the large amount of data coming from more than thirty MAS, including different asset types, to address both challenges and develop novel solutions that can be exploited by other sectors that will rely on fleets of RAS, e.g. the automotive sector.

Additionally, these areas of research will be complemented by the introduction of knowledge-induced hybrid deep learning solutions that incorporate virtual sensors based on dynamic models of the RAS. These solutions are critical to reduce the number of datapoints required for learning, thus reducing learning time. Furthermore, the virtual sensors are instrumental in correctly identifying faults at lower subsystem level so that targeted maintenance can be planned within a future intelligent maintenance framework, which is particularly important for under-observed systems like AUVs or satellites.

The project will not be limited to the introduction of new data-driven anomaly detection and fault detection and diagnostics algorithms, but rather verification and validation strategies for these solutions will be developed and documented.

---

[38] https://www.bodc.ac.uk/data/bodc_database/gliders/

By performing advanced research in the two selected main areas of research, we are confident that project ALADDIN will contribute to the BoK on the assurance of RAS as follows:

- Multi-platform sensing
  - **BoK: 2.2.1, 2.2.4.1 & 3.1.1 – Defining & verifying sensing requirements; identifying sensing deviations;**
- Fault diagnosis: Deep learning methods for fleet management
  - **BoK: 2.2.2, 2.2.4.1 & 3.1.2 – Defining & verifying understanding requirements; identifying understanding deviations.**

Although no contribution to the BoK is directly made in WP1, WP2 will start to address the definition of sensing requirements for RAS and procedures for the identification of sensing deviations. WP3 will define understanding requirements for RAS. In addition to the original plan, the project may also contribute to the identification of understanding deviations in WP3. WP4 will contribute to the verification of sensing and understanding requirements.

Selected Fault Diagnostics Models

From an initial literature review, GAN have been selected as particularly promising for the detection of faults on RAS [30], [31]. Deployment dive cycles from a fleet of vehicles tagged as "standard" in the first level of data labelling process will be applied to train the GAN-based models, so that the models can learn a representation of the data distribution pattern through a generative and adversarial process between generator and discriminator, whilst capturing the normal variances between individual vehicles. Deployment cycles with unusual patterns, which deviate from the learned pattern, will be marked as "anomalous" by the models if they present a high value of the anomaly score. The GAN-based models can be directly applied for fault diagnostics over larger datasets of deployment to label operational cycles automatically. It should be noted that improvements to existing GAN structures are ongoing to achieve higher accuracy with reduced computational costs.

Although GAN appear promising for the detection of faults on RAS, they are relatively difficult to train and are computationally expensive. Therefore, other deep learning approaches, e.g. semi-supervised domain adaption, will be explored as the data labelling progresses in WP2 to achieve higher accuracy and better genericity and transferability. In addition, knowledge-induced learning and hybrid methods will be developed in WP2 and WP3 to improve fault isolation and identification. In particular, combining GAN with variational autoencoders and model-based virtual sensors can provide improve fault isolation and identification accuracy.

Initial Planning for the Verification Stage

Although proper planning for the verification stage of the project will be carried out at the end of WP2 and WP3, an initial planning has been started. In particular, we have begun to plan for field tests, which need to be planned a long time ahead to reduce risks and secure the funding, and for the implementation of the implementation of the designed methods onto C2.

Possible Field Tests

An initial field test is preliminary planned for December 2020 (dependent on Covid) as part of the EUMarineRobots Transnational Access scheme (EC/731103) project "Fault dEtection, isolation and recovery for AuTonomous UnderwateR vEhicles (FEATURE)" run by UCL in collaboration with the University of Porto, Portugal. The project will verify the accuracy of novel fault diagnostics methods in detecting, isolating and identifying a broken pressure sensor on a profiling AUV, so that the on-board controller will switch to the pressure sensor in the CTD unit, and a broken inertial measurement unit on a mapping AUV, so that the on-board computer will switch to the pitch tilt sensor.

Two additional field tests with direct relevance to project ALADDIN are also being discussed as possible:

- Field tests of multiple AUVs off the Island of Mallorca in early 2021 by the NOC as part of commissioning trials in collaboration with SOCIB,
- Field tests of one glider simulating wing loss and ballasting problems in February 2021 by UCL in collaboration with SOCIB as part of the transnational access scheme JERICO (EC/871153) - dependent on funding.

Additionally, depending on the evolution of the Covid outbreak, deployments by the NOC in late 2021 can be used for verification of the fault detection and diagnostics system through C2, although the system will not make any changes to the missions, i.e. will be used in read-mode only.

Interfacing of a Model-Based Fault Diagnostics System with C2

Anticipating the WP4 verification stage, we have started to integrate a model-based fault diagnostics system within C2 to ensure that we have the interfaces required for the implementation of ALADDIN's fault detection and diagnostics system on actual deployment of AUVs at sea. The C2 interfacing assumes that the diagnostics system is to be applied as an external algorithm. This assures that new algorithms for MAS to be implemented in C2 are done in safe and secure manner, while enabling near-real-time execution and monitoring.

The implementation of the model-based solution also allows us to assess and improve the robustness of algorithms developed for real-world operations. Furthermore, maintenance is another aspect that we consider to ensure continuous reliability of the system. As the requirements of the algorithms grow, we have taken this opportunity to plan for enhancements to the C2 infrastructure to include more features such as metadata access and events labelling.

1.6. Conclusions

In the absence of accepted international terminology for MAS at present, the vocabulary for condition monitoring of MAS that will be used in the remainder of the project has been introduced. The terminology is designed to be general and transferable to any RAS technology, although failure modes are specific to each design and can be obtained from the FMEA. A procedure for the robust labelling of RAS timeseries data has been presented. The process is based on the principle of simplicity and includes a step of human-in-the-loop feedback. The

use of metadata based on time stamps will enable the labelled data points on the different layers of timeseries: RAS deployment, segments of the deployment and individual cycles.

Furthermore, specific datasets of deployments of a range of AUV technologies have been selected to be used in the remainder of the project. The datasets will be complemented by satellite data to capture the environmental disturbances during the deployments.

After identifying an interesting research gap, project ALADDIN will focus on multi-platform sensing and deep learning for fleet management fault diagnostics to maximise its contribution to advancing the state of the art of intelligent maintenance systems.

At present, a GAN-based method is being introduced that will be used for the definition of sensing requirements and deviations of RAS in WP2. The method will be improved with a hybrid model-based and data-driven architecture in WP3 to define understanding requirements and deviations. Additionally, code is being implemented on C2 for the verification of the methods in WP4, with some field tests of AUVs being planned.

2. Sensing

This section focuses on sensing. The contents presented in Section 2 has been published in [32]. Another study [33] using a Variational Autoencoder with Long Short-Term Memory Networks to detect anomalies for UGs has been accepted for publication.

2.1. Introduction

Whilst autonomous systems are predicted to become pervasive in the maritime industry [34], this growth is currently heavily constrained by the challenges of fully independent remote operation in hazardous and dynamic marine environments. Marine Autonomous Systems (MAS), such as Underwater Gliders (UG), can be at sea for up to months at a time, during which they periodically surface and communicate via satellite with remote expert operators known as pilots. The transmission of data to and from the MAS is severely constrained by low-bandwidth satellite, making it challenging for pilots to monitor MAS time series data and behaviour during operation manually. If the underlying cause of observed adverse behaviour cannot be correctly diagnosed and the situation remedied, e.g. via the remote adjustment of piloting parameters or mission scope, the MAS and its data cargo can be lost or present a hazard to shipping [35]. As a result, to reduce operational costs, increase reliability and scale-up the use of MAS within the maritime industry, strategies must be developed for automated anomaly detection and fault diagnosis.

The code of practice for maritime autonomous surface systems developed by Maritime UK [36] recognises the need for MAS to support on-board signal processing with remote condition monitoring to interpret the impact of faults and adverse conditions on the vehicle's safety and performance. On-board systems are limited by power and computational constraints, whilst current manual detection and diagnosis approaches are limited by the experience of the individual pilot and are subject to human error, especially when MAS require pilot attention around the clock. In the absence of general on-board anomaly detection and diagnosis systems, the ability to transmit sensor data in a timely manner to an off-board system and to receive appropriate commands in response becomes of critical importance for MAS safety and performance.

The operation of MAS platforms beyond the visual line of sight requires a suitable command and control system. For example, the UK's National Oceanography Centre (NOC) Oceanids C2 system is a platform to support the over-the-horizon operation of MAS within the National Marine Equipment Pool for efficient fleet management [37], [38]. Another example is the LSTS Neptus and Dune over-the-horizon command-and-control environment [39]–[41]. This work aims to develop a holistic automated anomaly detection system[39], well-suited to the limited availability of multivariate time series data during MAS operations.

The contributions of this study are twofold. First, this work proposes an improved Bidirectional Generative Adversarial Networks (BiGAN) anomaly detection system guided by periodic assistive hints to achieve effective and stable training of generative adversarial models. Second, this work introduces a novel holistic anomaly detection system for MAS to be integrated within remote control centres to monitor operations over the horizon. The

---

[39] https://github.com/pwu01/ALADDIN-BiGAN-anomaly-detection

system is based on BiGAN to detect faults by tracking the anomaly score. As compared with state-of-the-art anomaly detection systems for MAS that exploit steady-state conditions, deep neural networks are used to capture dynamic effects from the time series data. Two healthy deployment datasets in time series are used to train the system via unsupervised learning. The developed system is tested using actual datasets from nine deployments collected by a selection of vehicles operating in a range of locations and environmental conditions with varying mission length. A sensitivity analysis on the data decimation settings for satellite communication suggests the proposed approach is insensitive to these settings, making it suitable for Near Real-Time (NRT) anomaly detection for UG that are known to be under-observed systems. Such an unsupervised approach requires minimum training data preparation efforts and successfully detects anomalies for the nine test MAS deployments.

2.2. Related work

Methods for condition monitoring can be subdivided into model-based and data-driven diagnostics [10], [42]. The former relies on dynamic models of the physical systems, whereas the latter on the analysis of actual sensor data. Whilst model-based solutions are better for condition monitoring of new systems where available data is limited, data-driven methods show significant improvements in accuracy in the cases where significant prior data exists [43].

A review of nonlinear model-based methods for condition monitoring can be found in [44]. These approaches are useful for systems that present strong nonlinearities or coupling. A summary of fault detection methods for aircraft based on signal-processing and dynamic models can be found in [13]. These techniques are robust, simple and computationally relatively inexpensive. Data-driven methods can be generalised to different fault detection and diagnosis problems and scaled to a large number of sensors. These approaches would require the data to be appropriately collected and processed.

[45] integrated fault tolerance into the design of a robot real-time control architecture showing the benefits of including condition monitoring considerations from the initial stages of design of a new prototype. Specifically for Autonomous Underwater Vehicles (AUVs), many fault detection studies involve thrusters, inclusive of model-based solutions [46], radial basis function networks [47], Gaussian particle filter [48] and artificial immune system [49]. Clustering solutions are also investigated by [50] to determine faults in an unsupervised way. [51], [52] have developed an automatic fault detection system for long-range AUVs based on Bayesian nonparametric topic modelling techniques. Although the dataset focuses on the identification of bottoming events, the behaviour of the analysed long-range AUV is similar to that of UGs. The nearest neighbour classifier presents particularly high accuracy over two different test sets. A system to develop safety indicators for the operation of MAS is described in [53], with a case study on an AUV. [11] propose an integrated fault detection and diagnosis architecture for AUVs, although the focus is on on-board systems. [12] have designed rule- and model-based methods for the detection of the loss of wings on UGs and the onset of high levels of marine growth on UGs [14]. Further work by [54] has developed and tested an anomaly detection system that blends model- and data-based solutions to detect both simulated and naturally accumulated biofouling.

[55] provide a comprehensive review of deep learning for anomaly detection, and propose a taxonomy by classifying the state-of-the-art deep anomaly detection techniques into three categories, i.e. feature extraction, learning feature representations of normality and end-to-end anomaly score learning. In the category of learning feature representations of normality, models based on Auto-Encoder (AE) are proposed to detect anomalies by learning low-dimensional feature representations to reconstruct given data instances [56]–[58]. For the anomaly detection of UGs, [59] have developed different data-driven solutions, including feedforward neural networks and Auto-Encoders (AEs) that can detect anomalies such as wing loss and marine growth. However, the features learned by AE-based models can be biased by infrequent normalities in the training dataset. With data instances encoded by a prior distribution over the latent space, Variational Auto-Encoder (VAE) enables a better reconstruction of input data instances; hence improved anomaly detection performance can be achieved. For anomaly detections of multivariate sequence data, variants of VAE have haven been developed [60], [61]. The Generative Adversarial Networks (GAN) proposed by [62] can capture the data distribution via generative and adversarial processes. The improved capability of capturing data distribution is particularly useful for anomaly detection applications [63]. The superior feature representation learning capability makes GAN particularly promising for remotely operated MAS (e.g. UGs) that can be highly under-observed due to low data transmission bandwidth and limited sensing ability to reduce the on-board space and power requirements. However, GAN are constrained by issues such as training instability [55].

Based on work of [62], [64] and [65] have developed variants of the original GAN, i.e. Adversarial Learned Inference GAN (ALI-GAN) and Bidirectional-GAN (BiGAN), respectively, to additionally learn a latent representation of the data, which have become the basis of several GAN-based anomaly detection systems. [66] developed a BiGAN-based anomaly detection system for high-dimensional real-world data such as images. [67], [68] have developed a series of Bi-GAN based anomaly detection models for medical image anomaly detection. An earlier study by [69] applied GAN to detect cyber attacks, using multivariate time series with the need of the inference process to map the test data back to latent space. Although these GAN-based anomaly detection systems appear successful in the applied domains, the GAN-based anomaly detection system is still relatively difficult to train for reasons including its unsupervised nature and the generative and adversarial process between multiple deep neural networks [70]. Despite GAN-based anomaly detection systems' success in other domains, they have not been applied to MAS, which are subjected to limited accessibility to system data and require a high level of generality to detect unpredicted anomalies in highly dynamic ocean environments.

2.3. Underwater gliders and data description

Slocum underwater gliders

All data used in this study are from deployments of Slocum G2 UG [71], manufactured by Teledyne Webb Research [25], [26]. As shown in Figure 2.1, a Slocum UG is actuated by a Variable Buoyancy Device (VBD), which enables the vehicle's displacement and thus its buoyancy to be varied. Pitch is controlled by shifting the position of a movable battery pack, and the yaw angle is controlled using a rudder. Using fixed wings to provide lift, gliders can

perform a sawtooth-like profile through the water-column. A Slocum UG starts a "yo", or cycle, by reducing its buoyancy and shifting the battery forward to initiate the descent, then extends the VBD and shifts the battery afterwards to climb to a designated depth at the apogee, completing the glider's "yo". A single dive can comprise multiple yos as shown in Figure 2.1.
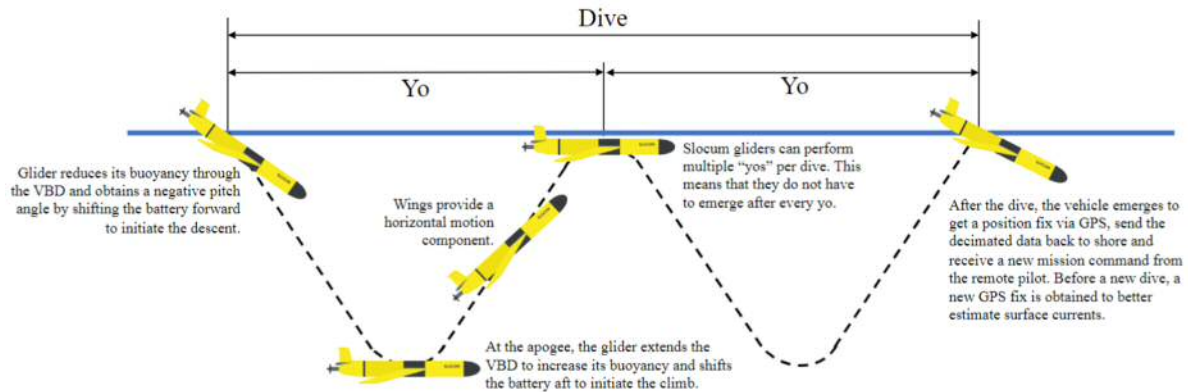
Figure 2.1: Diagram showing the concept of operation of a Slocum UG. The drawing is not to scale: the analysed vehicles reach their apogee at a maximum depth of either 200 m or 1,000 m and have glide path angles with a magnitude in the range of 15° to 30°.

Once deployed, UGs operate fully autonomously, with pilots relying on a limited snapshot of multivariate time series data sent via satellite whilst the vehicle is on the surface, which includes system health variables, current instructions, last GPS position and decimated data from past dives. During normal operation, the remote pilots will first manually check this dataset for subsystem errors, warning and oddities reported by the glider itself (e.g. glider stalls, behaviour errors, and communication interruptions [26]) along with the sawtooth dive profile to ensure each yo is symmetrical and the glider is reaching the target depth. Progress towards the target waypoint is also considered, along with a check of the battery health and consumption. This check is usually performed once per day, with the pilot making smaller observations more regularly after each dive. Therefore, pilots are only likely to look into the flight parameters in detail if the glider: is reporting errors in these subsystems, is failing to dive correctly or is clearly not making expected progress. Hence, issues that gradually emerge or that affect parameters outside those routinely monitored, such as roll, can go unnoticed, resulting in significant impacts on vehicle endurance and safety. Consequently, a smart anomaly detection system is crucial.

Datasets

Table 2.1 lists the multivariate time series datasets used in this study. The datasets are measured by Slocum G2 gliders over ten deployments operated by the NOC (1-10) [72] and one deployment (11) operated by the Oceanic Platform of the Canary Islands (PLOCAN) giving a total of 11 deployments. The labels correspond to healthy or standard baseline conditions, natural and simulated biofouling, angle of list, loss of one wing, strong environmental disturbances, e.g. due to ocean currents, and bulky sensory packs, e.g. the turbulence probes for the Ocean Microstructure Gliders (OMG). Biofouling caused by marine growth in shallow, warm and tropical waters can lead to an increase of a UG's weight, a significant drop in speed, even possible premature retrieval at sea [14]. OMG would lead to a higher drag coefficient and a higher negative buoyancy offset. The first two deployments (healthy) are used to train

the anomaly detection system and the remaining nine datasets, which include several anomalies as well as healthy behaviour, are used for testing purposes to assess the generality of the approach.

Table 2.1: The data applied in this study is measured by a number of Slocum G2 gliders over eleven developments.

No.	ID	Date	Project	Organisation	Location	Duration [days]	Status	Purpose
1	unit 345	2014	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	123.9	Healthy	Training
2	unit 397	2015	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	45.9	Healthy	Training
3	unit 419	2015	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	11.0	Healthy	Testing
4	unit 399	2015	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	84.6	Possible biofouling	Testing
5	unit 423	2015	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	6.8	OMG	Testing
6	unit 424	2015	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	20.8	OMG	Testing
7	unit 194	2017	ALTERECO	NOC	the North Sea	83.9	Angle of list	Testing
8	unit 304	2019	ALTERECO	NOC	the North Sea	76.9	Loss of right wing	Testing
9	unit 345	2019	ALTERECO	NOC	the North Sea	76.8	Strong disturbances	Testing
10	unit 436	2019	ALTERECO	NOC	the North Sea	89.8	Loss of left wing	Testing
11	unit 492	2020	IDUG	PLOCAN	Gran Canaria	9.5	Simulated biofouling	Testing

The variables are detailed in Table 2.2. Mission specific geographical positions, control signals (heading control, pitch control, rudder angle control and VBD control) are not included in the datasets to ensure generality. The vertical velocity is calculated from the depth signal, which is in turn obtained from the pressure signal. The scientific pressure sensor measures the sci_pressure signal.

Table 2.2: Sensor list.

No.	Sensor	No.	Sensor	No.	Sensor
1	battery position	6	pressure	11	heading
2	battery voltage	7	roll	12	temperature
3	state of charge	8	rudder angle	13	vacuum
4	leak detection voltage	9	pitch	14	VBD
5	vertical velocity	10	conductivity	15	sci_pressure[a]

[a] sci_pressure is the pressure as measured by the scientific payload pressure sensor.

2.4. Anomaly detection using BiGAN

The presented anomaly detection method is based upon BiGAN [64], with additional training hints guiding more effective generator (G) and discriminator (D) training. In each training iteration, the discriminator, generator and encoder (E) are trained concurrently. The assistive hint loss function is applied periodically to guide the encoder and generator using the errors terms of data patch reconstruction and discriminator feature. This approach is inspired by [68] and [70] but has been improved. In [68], the encoder E is trained only after D and G have been trained, whereas in this study the discriminator, generator and encoder are trained concurrently. In [70], the hint loss is added directly to the BiGAN loss function; our approach instead applies a periodic update step to the parameters of the generator and encoder.

Problem statement

[62] proposed Generative Adversarial Networks for estimating generative models via an adversarial process training a generative model *G* to capture the data distribution, and a

discriminative model D that estimates the probability that a data sample comes from the training data or is generated by G. This framework trains D and G concurrently such that D maximises the probability of assigning the correct label to both training samples from data x and generated samples from G. D and G play a minimax game with the value function $V(G,D)$:

$$\min_{G} \max_{D} V(D,G) = \mathbb{E}_{x \sim p_{data}(x)} \left[\log D(x) \right] + \mathbb{E}_{z \sim p_z(z)} \left[\log \left(1 - D(G(z)) \right) \right] \quad 2.1$$

where   is the data distribution,   is a prior on input noise variables. Although the ability of the original GAN framework to learn generative models mapping from simple latent distributions to arbitrarily complex data distributions has been demonstrated, it cannot project data back into the latent space. The BiGAN [64] and ALI-GAN [65] adopt a similar approach using an encoder with a generator to learn this inverse mapping.

Figure 2.2 shows the structure of the BiGAN, which includes an additional encoder E that maps data x to its latent representations z. A trained BiGAN encoder can serve as a useful feature representation for related semantic tasks, i.e. the latent representation z can be regarded as a representation of data x. Unlike the standard GAN [62], the discriminator D of the BiGAN discriminates $(x, E(x))$ and $(G(z), z)$. The training objective of the BiGAN is:

$$\min_{G,E} \max_{D} V(D,E,G) = \mathbb{E}_{x \sim p_{data}(x)} \left[\log D(x, E(x)) \right] + \mathbb{E}_{z \sim p_z(z)} \left[\log \left(1 - D(G(z), z) \right) \right] \quad 2.2$$
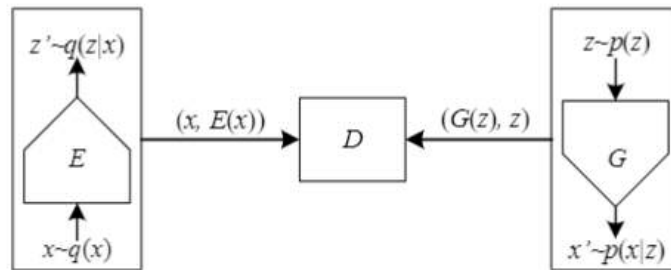
Figure 2.2: Structure of the BiGAN [64].

Algorithm 1 The training procedure for underwater glider anomaly detection using GAN.

1: Prepare the training dataset
2: Initialise D, G and E parametrised by θ_D, θ_G and θ_E, respectively
3: **for** $i = 1$ to t **do** $\triangleright$ t is the number of total training iterations
4:      **procedure** TRAIN D, G AND E
5:          **for** $j = 1$ to k **do** $\triangleright$ Assistive hint is applied every k training steps
6:              Sample a mini-batch with M data samples x from the training dataset
7:              Sample z with a size of M from a prior Gaussian distribution $z \sim p_z(z)$
8:              $\mathcal{L}_{BiGAN} \leftarrow \frac{1}{M}\sum_{s=1}^{M} \log\left(D\left(x^{(s)}, E(x^{(s)})\right)\right) + \frac{1}{M}\sum_{s=1}^{M}\log\left(1 - D(G(z^{(s)}), z^{(s)})\right)$
9:              $\theta_D \leftarrow \theta_D + \nabla_{\theta_D}\mathcal{L}_{BiGAN},\ \theta_G \leftarrow \theta_G + \nabla_{\theta_G}\mathcal{L}_{BiGAN},\ \theta_E \leftarrow \theta_E + \nabla_{\theta_E}\mathcal{L}_{BiGAN}$
10:          **end for**
11:      **end procedure**
12:      **procedure** TRAIN THE G AND E WITH HINT
13:          Sample M data patches x from the training dataset
14:          $\mathcal{L}_{hint} \leftarrow \frac{\kappa}{n_x}\|x - G(E(x))\|_2 + \frac{1}{n_f}\|f(x, E(x)) - f(G(E(x)), E(x))\|_2$ $\triangleright$ Hint loss
15:          $\theta_E \leftarrow \theta_E + \nabla_{\theta_E}\mathcal{L}_{hint},\ \theta_G \leftarrow \theta_G + \nabla_{\theta_G}\mathcal{L}_{hint}$
16:      **end procedure**
17:      **if** $i \bmod n = 0$ **then** $\triangleright$ Test anomaly detection performance using synthetic sensor faults every n training iterations
18:          Anomaly score $\leftarrow \mathcal{L}_{hint}$
19:      **end if**
20: **end for**

Algorithm 1 details the training process of the BiGAN-based anomaly detection system. [64] have proven that the encoder *E* and generator *G* must learn to invert one another in order to fool the discriminator *D*. The encoder and generator of the BiGAN structure behave similarly to the encoder and decoder of an independent auto-encoder which learns a representation for a set of input data and reconstructs the data samples as closely as possible to their original inputs. Inspired by this feature of auto-encoders, we use the reconstruction difference between the input sample and reconstructed data to assist the BiGAN training, i.e. using the norm between the input data *x* and its reconstruction *G(E(x))* via *E* and *G*:

$$\mathcal{L}_{re} = \frac{1}{n_x}\|x - G(E(x))\|_2 \qquad\qquad 2.3$$

where   is the number of input data elements.

UNIVERSITY of York

ASSURING AUTONOMY INTERNATIONAL PROGRAMME

It is worth noting that [70] have proposed training generative adversarial models using several assistive hints. However, such hints are added to the BiGAN loss function directly in their approach. This work proposes applying such hints periodically to achieve higher training efficiency. In addition, in the discriminator network, the neural network layer right before the final output layer is defined as a feature layer, which outputs a feature f. With this feature f provided by the discriminator, an additional hint loss is defined as:

$$\mathcal{L}_{fe} = \frac{1}{n_f} \|f(x, E(x)) - f(G(E(x)), E(x))\|_2 \qquad 2.4$$

where   is the feature layer's number of neurons.

Combining   and   , the assistive hint loss function is thus:

$$\mathcal{L}_{hint} = \frac{\kappa}{n_x} \|x - G(E(x))\|_2 + \frac{1}{n_f} \|f(x, E(x)) - f(G(E(x)), E(x))\|_2 \qquad 2.5$$

where   is a hyperparameter which can be adjusted. Note that in the validation and test phases, the residual of   is defined as the anomaly score that represents the degree of anomalies. Ideally, the residual should be near zero if the query data patch is normal. A high anomaly score indicates the input data patch deviates severely from healthy deployment data pattern.

GAN for underwater glider anomaly detection

Figure 2.3 shows the proposed anomaly detection framework using BiGAN. In the training phase, the pre-processed healthy deployment datasets are applied to train the generator G, encoder E and discriminator D concurrently. Assistive hints are applied to guide the generator G and encoder E training periodically. In the test phase, the reconstruction error and discriminator feature hint error jointly represent the degree of an anomaly.
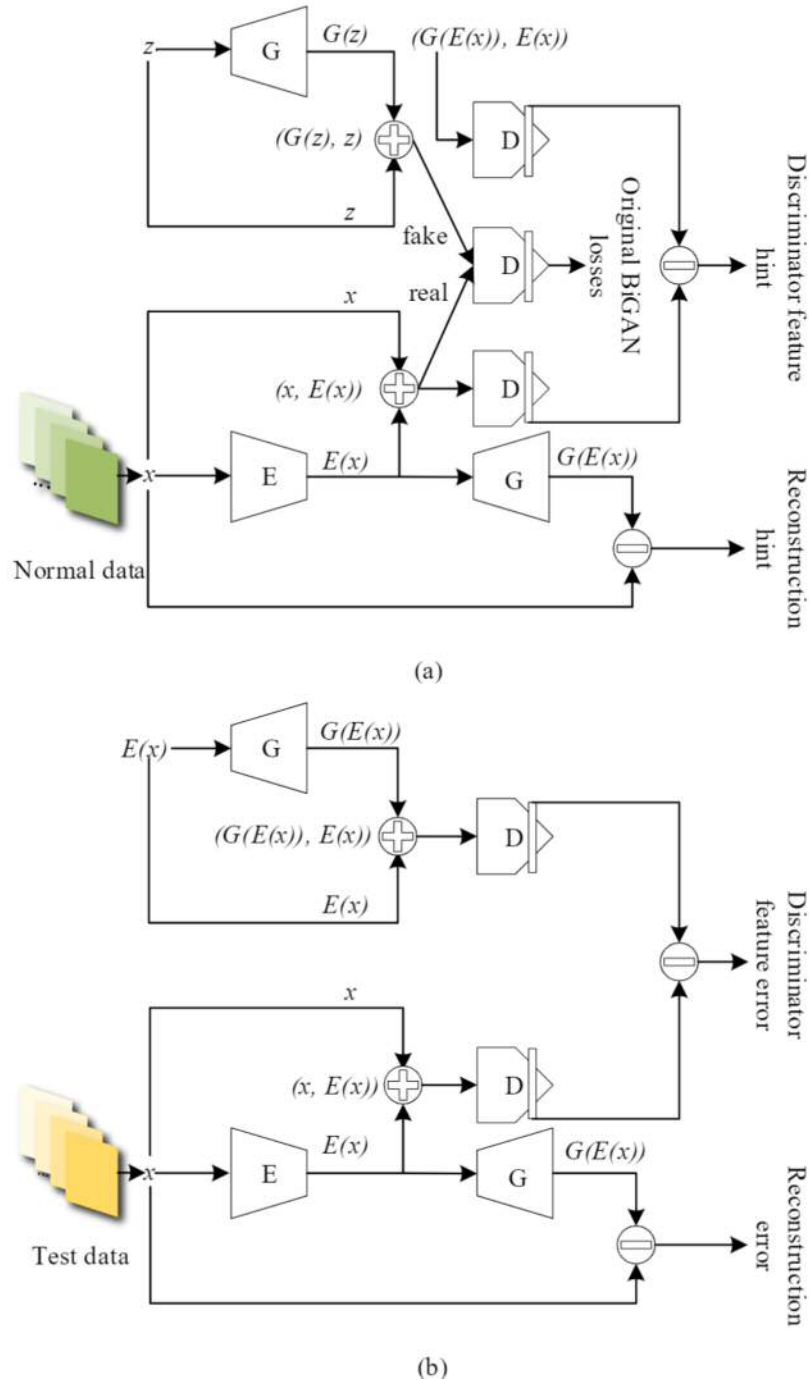
Figure 2.3: Anomaly detection using BiGAN for underwater gliders: (a) training using normal data and (b) testing using unseen deployment data.

Figure 2.4 shows the workflow of the proposed anomaly detection system underwater gliders. The model is trained using normal deployment data, i.e. no anomalies are included in the training dataset. It should be noted that this framework is unsupervised, as no labelling is required on the normal deployment data [67], [73]. During training, the model is tested periodically with synthetic sensor faults by manually setting one or more sensor readings to their lower bounds to check whether the model can detect synthetic faults. Once the model has been checked, it will be applied to detect anomalies for vehicles of the same type within

a fleet. If the model has learned the distribution   of the training data, it should be able to output a high anomaly score that represents the degree of an anomaly. The anomaly score should be close to zero if the input data is normal. When anomalies happen, the system is expected to output a high anomaly scores that represent the degree of anomalies.
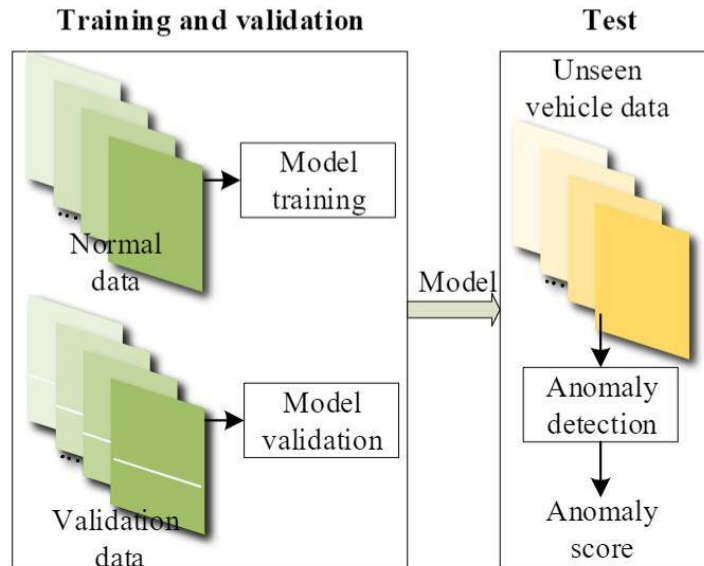
Figure 2.4: Workflow of unsupervised anomaly detection using GAN for underwater gliders.

2.5.   Training and validation

Data processing

Figure 2.5 illustrates the data processing process preparing the training and validation datasets, using the two healthy deployments of glider units 345 and 397 in 2014 and 2015. The dive cycles of the multivariate time series datasets are filtered to remove cycles with insufficient data points (less than ten data points for each sensor) to maintain cycles carrying sufficient features. Unified timelines with a time step of 5 s are subsequently applied to the filtered cycles by linear interpolation of all the remaining sensor measurements. The interpolated data are then normalised to the range of   . Random data patches with 64 time steps are sampled evenly from the valid dive cycles in each dataset. The data patches are augmented as   matrices, where *a* is the number of sensors, *b* is the number of time steps (64) for each data patch, so that the training dataset is ready for the training of the anomaly detection system. To monitor and check the performance of the anomaly detection system performance during training, synthetic sensor anomalies are injected into the data patches by setting a number of sensor measurements to their minimum values. Note that the sensors with anomalies are randomly chosen for each validation data patch. For the test datasets, a similar data processing flow has been followed. However, it should be noted that ten random data patches are sampled from each dive cycle in the test datasets.
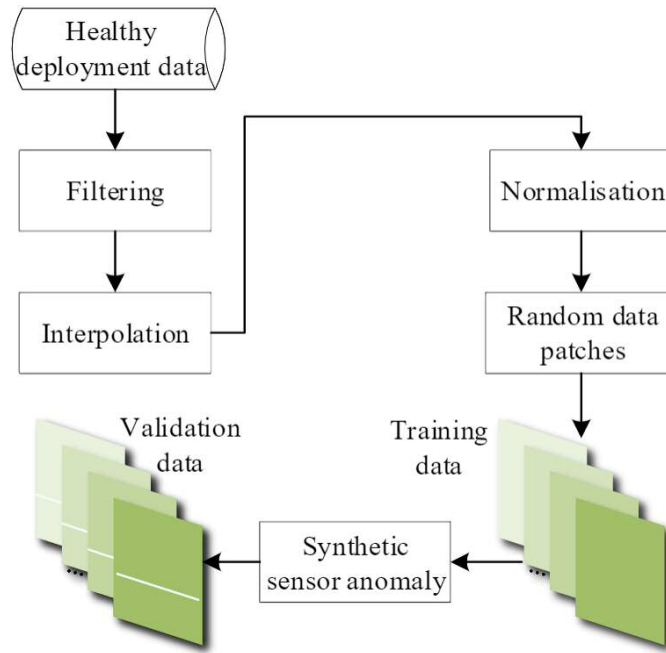
Figure 2.5: Data processing procedure applied to prepare the training and validation datasets using the two deployments of units 345 and 397.

Training

The proposed anomaly detection system is implemented in Python 3.8 and TensorFlow v2.4.1. The encoder network consists of: an input layer that receives the flattened data patches; four sequentially connected hidden dense layers that are followed by their own batch normalisation, leaky ReLU activation and dropout (0.1 dropout rate) layers; and an output layer with the size of 256 activated by sigmoid. The encoder and generator network structures are inversely similar to each other. The discriminator processes (*x, z*) to output a feature *f* from the feature layer and its prediction (a scalar) on whether (*x, z*) is from the training dataset. The hidden layers of the discriminator are configured the same as that of the encoder. The Adam optimiser is applied to update   and   is with a learning rate of   . Note that   is updated every   training iterations of   . The coefficient   that adjusts the weights of the reconstruction and feature losses is set as   . The mini-batch size is   . The training dataset includes   data patches extracted from two healthy deployments of UGs. The validation dataset includes   data patches extracted from the nine test deployments (see Table 2.1). The training is terminated after   training iterations of   (51 min on a Nvidia V100 GPU).

Validation using synthetic anomalies

Figure 2.6 shows the validation process using synthetic sensor faults (every   training iterations of *D*, *G* and *E*). For the validation without faults, the anomaly score stabilises to a value slightly less than 0.002. For the validation data samples with 1, 2 and 3 abnormal sensors, the anomaly scores stabilise to the values around 0.016, 0.022 and 0.028, respectively. The stabilised anomaly scores suggest that the training of the anomaly detection system has converged. It should be noted that the converged anomaly score (0.002) for normal data is

not exactly zero, which suggests the reconstruction and discriminator feature residuals still exist in low magnitudes. Nevertheless, this value is an order of magnitude lower compared to the ones with synthetic sensor anomalies.
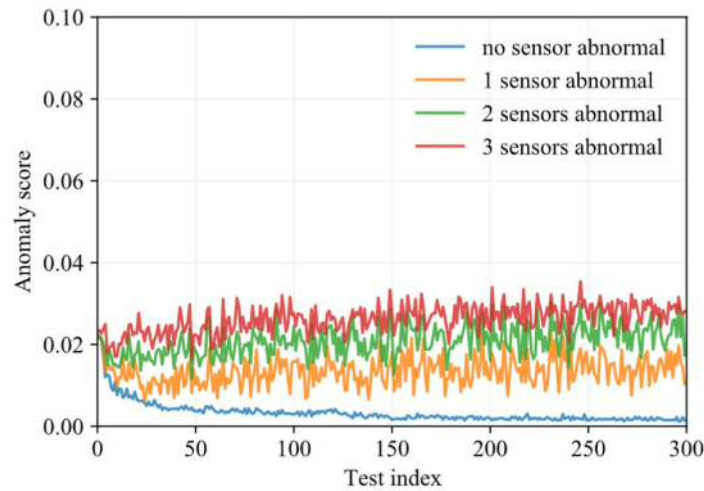
Figure 2.6: Periodic algorithm validation in training using synthetic sensor faults with 0, 1, 2 and 3 abnormal sensors randomly chosen. The model is tested every 100 training iterations of   . The records of the randomly chosen sensors are manually set to their minimum values.

Ablation study

An ablation study has been conducted to confirm the effectiveness of the assistive hints added to guide the BiGAN-based anomaly detection system training. Removing the assistive hint   (Eq. 2.1) leads to the BiGAN unable to reconstruct normal enquiry data accurately. As shown in Figure 2.7, when   is removed, the model has converged to a state of not being able to differentiate the four synthetic anomalies, suggesting the added hint has effectively guided the training of the BiGAN model (also see Figure 2.6).
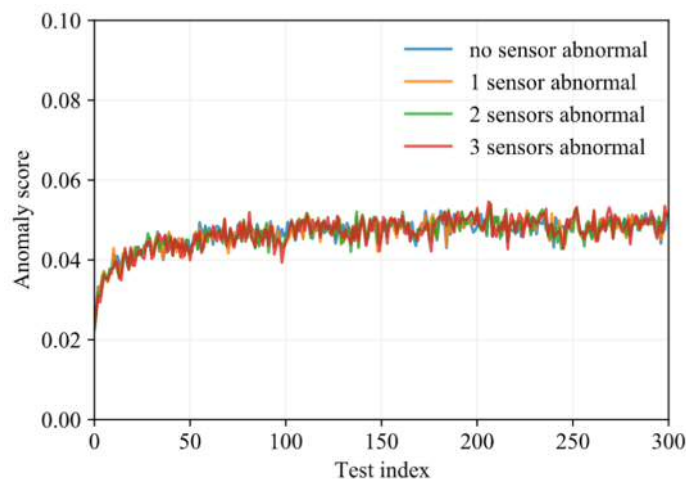
Figure 2.7: Periodic algorithm validation in training using synthetic sensor faults when no hints are added to guide the training.

2.6. Field test results and discussion

The anomaly detection system is tested using the deployment datasets detailed in Table 2.1. Ten data patches are randomly sampled from each dive cycle, which are then applied to get an averaged anomaly score for a dive cycle. Hence, the anomaly detection system applies to any dive cycles with a sufficient number of data points and can output an anomaly score for NRT monitoring when the dive cycle data is received. Table 2.3 details the mean and standard deviation values of the test deployments (calculated using dive cycle values in the deployments). The test deployments include one healthy deployment of unit 419. The average anomaly score of unit 419 will be applied as the baseline to assess the relative levels of the anomalies.

Table 2.3: Anomaly scores of the test deployments.

UG	Mean	Standard deviation
unit 419	0.00188	0.00026
unit 399	0.00235	0.00053
unit 492	0.00470	0.00113
unit 423	0.00353	0.00081
unit 424	0.00434	0.00024
unit 194	0.00303	0.00026
unit 304	0.00296	0.00067
unit 436	0.00319	0.00080
unit 345	0.00530	0.00097

The reconstructed sensor data is compared against the original input query data. The sensors with high anomaly score contributions are annotated by the system to alert the pilot. For the sensor's reading at time step , its reconstruction error   is defined as the absolute value of the difference between the enquiry data   and the reconstructed data via BiGAN:

$$\delta_{i,j} = |x_{i,j} - G(E(x))_{i,j}| \qquad \text{2.6}$$

which will be highlighted by its magnitude to visualise anomalies on sensor readings.

Healthy glider deployment

As shown in Figure 2.8, for a healthy deployment of unit 419 in 2015, the anomaly scores distribute evenly around their average value in general (0.00188 ± 0.00026). The initial anomaly score starts from the value of 0.0030 and decreases to the average anomaly score after ten dive cycles. The initial high anomaly score is likely caused by the shallow trial dives with significant dynamic effects at the start of a glider deployment. Note that the average anomaly score of 0.00188 very close to the converged anomaly score in the validation test without anomaly. The slight variance in the anomaly score throughout the cycles for this healthy glider deployment suggests that the proposed anomaly detection system can

accurately reconstruct the data samples similar to those it encountered in the training phase without giving false indications of anomalies.
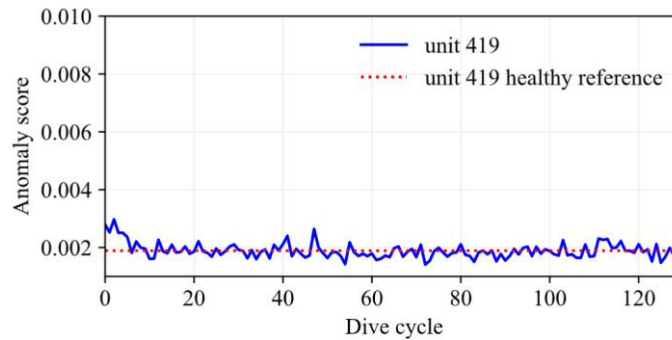
Figure 2.8: Test using a dataset of a healthy glider deployment dataset collected by unit 419 in 2015. The healthy reference is the average anomaly score of unit 419 over this deployment and will be applied as the baseline to assess the anomaly levels of other deployments.

Figure 2.9 shows the reconstruction errors of a typical data patch from unit 419 in a healthy deployment. Errors with small magnitudes distribute among most sensors, suggesting that the anomaly detection system has reconstructed the data patch with high accuracy. Ideally, the anomaly detection system should learn the healthy deployment datasets' data distributions applied for training. The low magnitudes of reconstruction errors for unit 419 suggest that the anomaly detection system has learned the patterns of the training datasets.
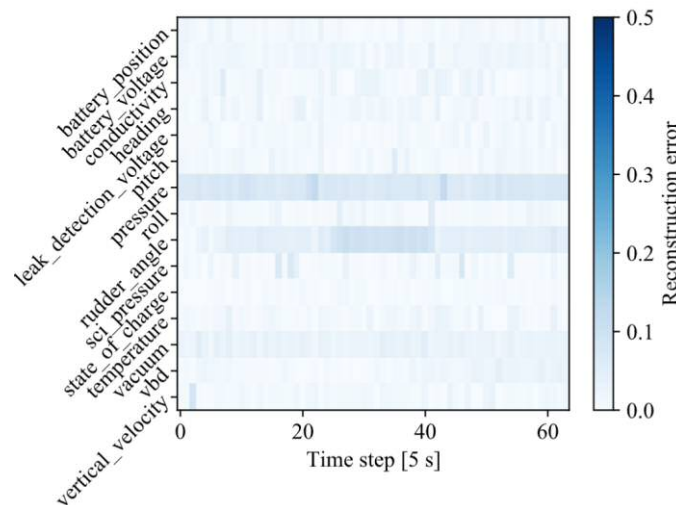
Figure 2.9: Reconstruction errors of a typical data patch of unit 419 in a healthy deployment. Note that the maximum reconstruction error could be up to 1.0; the upper limit of the colour bar is set as 0.5 to better visualise the errors.

Deployments with biofouling

Figure 2.10 details the anomaly scores of two deployments with biofouling, i.e. deployments of unit 399 (0.000235 ± 0.00053) with naturally accumulated biofouling in 2015 (Figure 2.10a) and unit 492 (0.00470 ± 0.00113) with simulated biofouling in 2020 (Figure 2.10b). For unit 399, shallow trial dives with dynamic effects at the beginning of the deployment lead to high initial anomaly scores. The anomaly score increases gradually from dive cycle 200, which is likely associated with marine growth [74]. The growing anomaly score is in line with the

increasing drag coefficient deduced through model-based and other data-driven approaches in [14]. The similar growth of the anomaly score and the drag coefficient suggest that the proposed BiGAN-based anomaly detection system can capture slowly growing anomalies, even though it is trained with deployment datasets collected by other gliders in different missions. As shown in Figure 2.10b, for the deployment of unit 492 with drag simulators added to the UG to simulate extreme levels of marine growth [14], the anomaly scores of the dive cycles are distributed around their average value of 0.00470 which is high above the baseline deduced from the healthy deployment of unit 419. The average anomaly score of unit 492 is close to the anomaly values of the final dive cycles of unit 399.
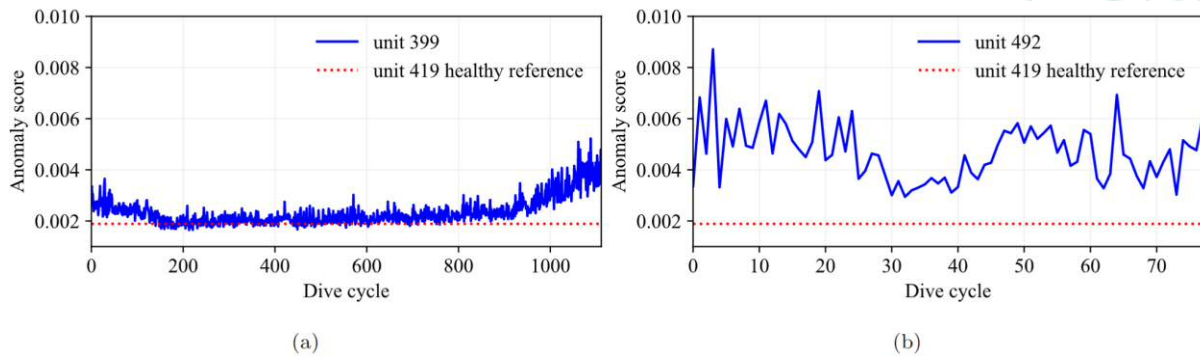
Figure 2.10: Test using datasets of two deployments with biofouling: (a) unit 399 in its final stage of deployment with a high anomaly score caused by naturally accumulated biofouling and (b) unit 492 with simulated high levels of biofouling.

As shown in Figure 2.11a, the system has reconstructed a data patch from the final stage of this deployment with high reconstruction errors due to possible biofouling. The relatively high reconstruction errors can be observed from the sensors including VBD, state_of_charge, depth and pressure. Even stronger highlights of reconstruction errors can be observed in Figure 2.11b from a data patch of unit 492 with simulated biofouling.
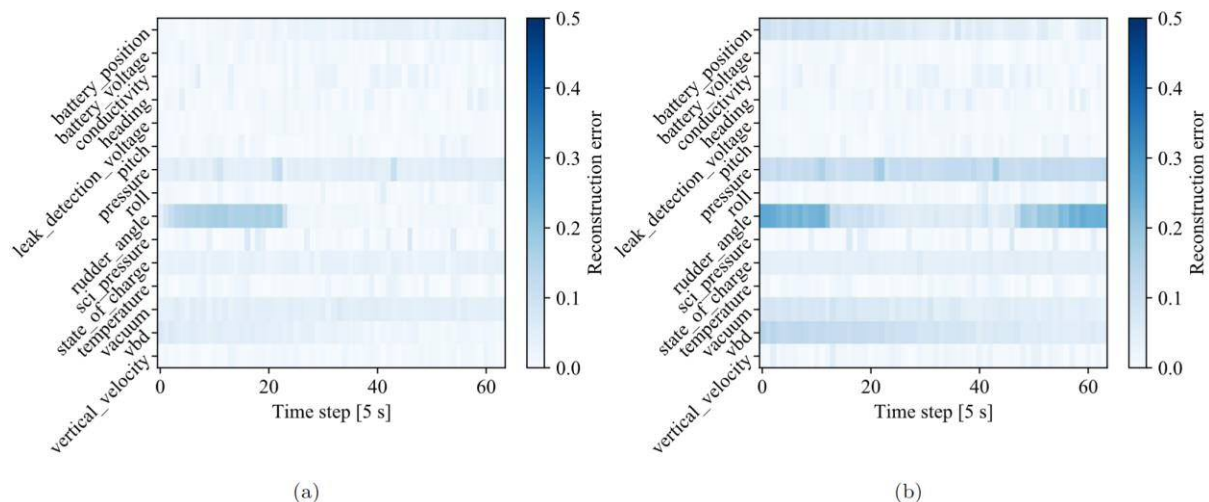
Figure 2.11: Reconstruction errors of typical data patches with biofouling: (a) unit 399 in its final stage of deployment with a high anomaly score caused by naturally accumulated biofouling and (b) unit 492 with simulated high levels of biofouling.

Deployments with OMG

Figure 2.12 shows the anomaly scores of two deployments (unit 423 and unit 424) with OMG. The two deployments' anomaly scores are similar to each other, i.e. 0.00353 ± 0.00081 for unit 423 and 0.00434 ± 0.00024 for unit 424, respectively. The additional turbulence probes have affected the gliders' hydrodynamic characteristics, leading to increased drag coefficients and high negative buoyancy offsets and, consequently, to higher anomaly scores. It is worth noting that no dramatic change in the anomaly score magnitudes can be observed from the two deployments with OMG, suggesting that it is unlikely other anomalies occurred during the two deployments. The proposed anomaly detection system appears to be effective in detecting anomalies caused by increased drag coefficient consistently throughout the deployments of the glider units.
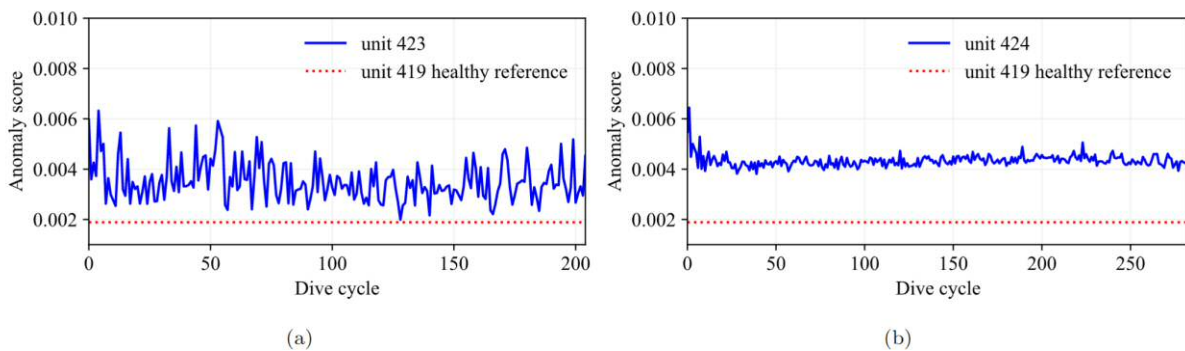
Figure 2.12: Test using datasets of two deployments with OMG: (a) deployment of unit 423 in 2015 and (b) deployment of unit 424 in 2015.

Figure 2.13 shows the reconstruction errors of a typical data patch of unit 423 and unit 424. Obviously, the rudder_angle, battery_position, battery state_of_charge and VBD sensors that are the most highlighted ones, as the vehicle becomes less manoeuvrable due to the higher drag. Although errors in lower magnitudes can also be observed for other sensors, the most striking difference is the vacuum sensor which indicates a different pressure inside the pressure hull was used for the OMG as compared with the standard gliders.
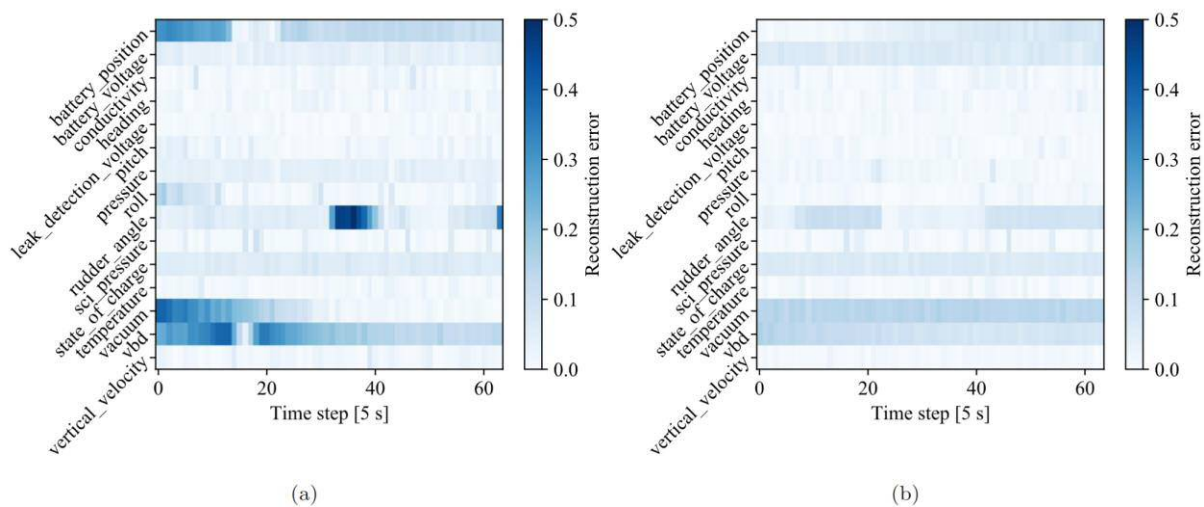
Figure 2.13: Reconstruction errors of typical data patches with OMG: (a) unit 423 and (b) unit 424.

Deployment with angle of list

Figure 2.14 details the anomaly scores of the unit 194 deployment (0.00303 ± 0.00026) in 2017 which experienced an angle of approximately 9° list due to a pre-deployment error. Fluctuations of the anomaly scores can be observed, which suggests that the operational status of this glider were relatively unstable compared to the deployments discussed in the section of Healthy glider deployment. In this deployment, the angle of list caused asymmetric drag and lift forces. Consequently, the average anomaly scores presented in Figure 2.14 are 61.2% higher than that of the unit 419 baseline deployment. In addition, an increasing trend of the anomaly score can be observed in the final stage of the deployment (after 680 dive cycles), which is possibly due to the pilot's control decisions. It appears that the anomaly detection system can detect the transient effects in the abnormal glider status.
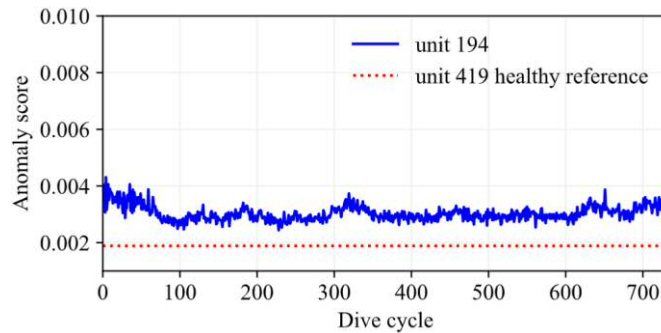
Figure 2.14: Test using dataset of a deployment dataset collected by unit 194 with angle of list in 2017.

Figure 2.15 presents the reconstruction errors of a data patch extracted from the deployment of unit 194, which encountered a significant angle of list of approximately 9°. The underwater glider attempted to compensate the list angle by applying control actions to the rudder. However, no apparent errors can be observed from the roll and pitch signals, which could be due to the errors are low in magnitudes compared to the maximum readings of these signals.
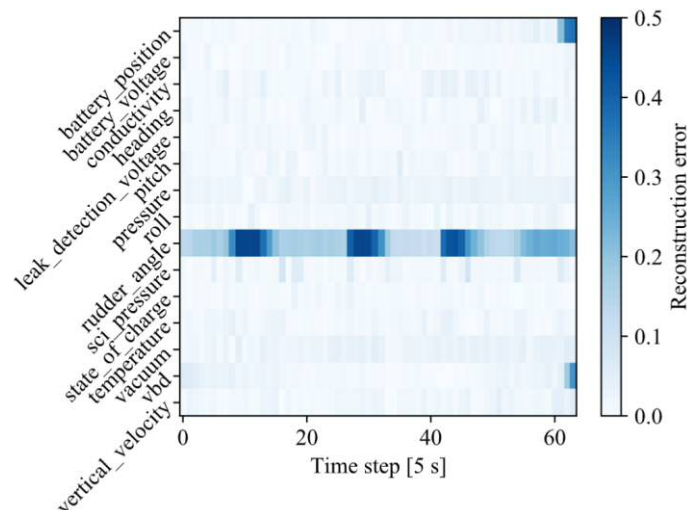
Figure 2.15: Reconstruction errors of typical data patch of unit 194 with angle of list.

Deployments with loss of wing

Figure 2.16 shows the anomaly scores of two deployments with the loss of one wing, i.e. unit 304 (0.00296 ± 0.00067) with the loss of the right wing (Figure 2.16a) and unit 436 (0.00319 ± 0.00080) with the loss of the left wing (Figure 2.16b) in 2019. As shown in Figure 2.16a, fluctuating high anomaly scores are present for the early dive cycles (before 500 dive cycles) of unit 304, which is mainly due to the very shallow dive depth, where dynamic effects of the vehicle and the variation in the oceanic sensor data are more significant [12]. The anomaly scores of unit 304 start to increase abruptly from dive cycles 510 to 560, suggesting that the anomaly detection system has detected an unusual pattern. This anomaly has also been detected in [12], which corresponds to unit 304's loss of right wing in this deployment. As shown in Figure 2.16b, the anomaly score of unit 436 jumps to 0.0060 within one dive cycle after 230 dives, suggesting something very unusual happened within that dive cycle. In addition, the anomaly scores of dive cycles from 120 to 220 suggest that unit 436 may have encountered an unusual event, which subsequently caused a delayed but instant loss of its left wing.
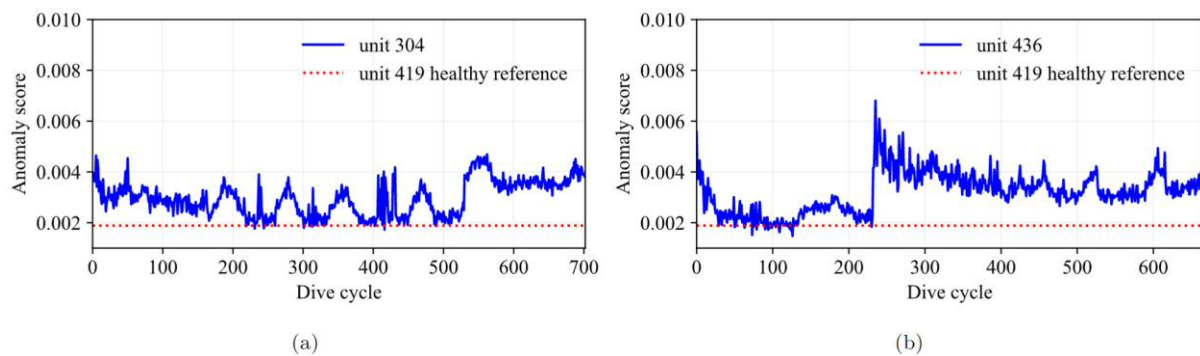
Figure 2.16: Test using datasets of two deployments with wing loss: (a) deployment of unit 304 with the loss of the right wing in 2019 and (b) deployment of unit 436 with the loss of the left wing in 2019.

Figure 2.17 highlights the reconstruction errors of data patches from units 304 and 436, which lost the right wing and the left wing in their deployments, respectively. The rudder_angle signal is the most highlighted for both units 304 and 436, suggesting the gliders had frequently attempted to use their rudders to compensate for the imbalance caused by the wing loss. In addition, the battery state_of_charge unit 304 signal also shows a high level of reconstruction errors due to the compensating control actions that consumed excessive energy from the battery. Note that the reconstruction error shown in Figure 2.17a is a late-stage data patch of the deployment; the accumulated excessive energy consumption from the battery becomes apparent in the case of unit 304.
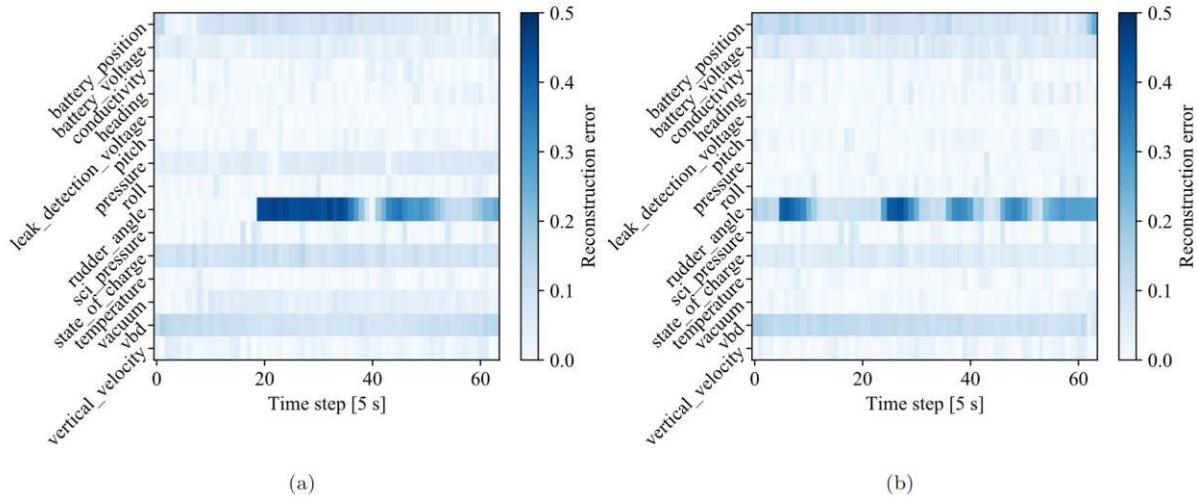
Figure 2.17: Reconstruction errors of typical data patches of losses of wings: (a) unit 304--loss of the right wing and (b) unit 436--loss of the left wing.

Deployment with strong environmental disturbances

Figure 2.18 details the anomaly scores of the dive cycles within the deployment of unit 345 in 2019. This deployment was subject to strong disturbances (e.g. transverse ocean currents). Consequently, the average of anomaly scores (0.00530 ± 0.00097) is 181.9% higher than the baseline of unit 419. It should be noted that another dataset collected by unit 345 in 2014 has been included in the training dataset. The system has identified the anomalies of unit 345 in 2019.
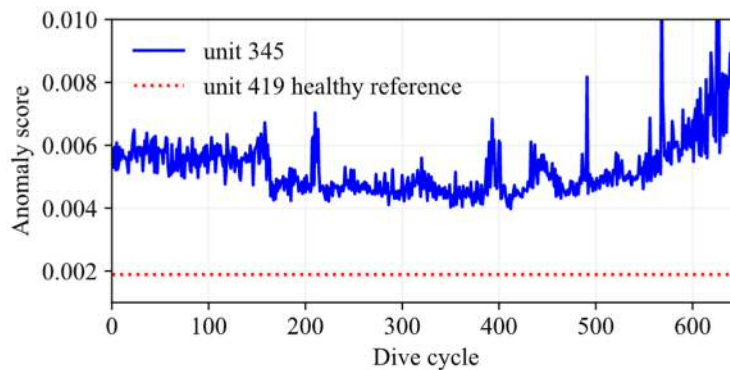
Figure 2.18: Test using the dataset of a deployment collected by unit 345 with strong environmental disturbances in 2019.

Figure 2.19 shows the reconstruction errors of a data patch of the deployment of unit 345 that experienced strong ocean disturbances such as transverse ocean currents. Apparent errors can be observed from the sensors, including battery_voltage, rudder_angle, VBD and state_of_charge. Unlike other scenarios that have been discussed, battery voltage signal deviates from the pattern the anomaly detection system has learned, which is probably due to the glider had to make frequent adjustments to its rudder to overcome the strong disturbances causing excessive power consumption from the battery, which led to a lower battery voltage.
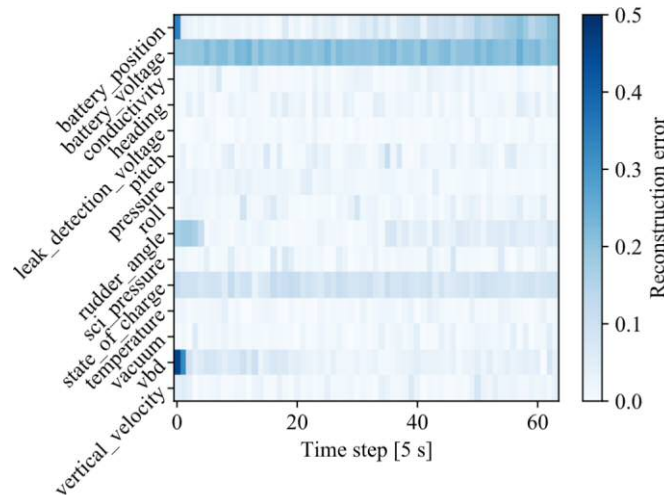
Figure 2.19: Reconstruction errors of a typical data patch of unit 345 encountered strong environmental disturbances.

Summary

The proposed anomaly detection system successfully identifies anomalies for a fleet of gliders operating in different deployments over different times. In addition, the highest observed anomaly score among all tests is around 0.010, which is significantly lower compared to the anomaly score obtained for simulated sensor faults. This indicates that it is significantly more challenging for humans to detect actual anomalies with abnormal patterns distributed over multiple sensor readings in low magnitudes.

2.7.   Sensitivity study of NRT data decimation settings

In this section, the influence of NRT data decimation settings over anomaly detection accuracy is investigated. The sensitivity study is implemented by varying the data sampling intervals to simulate the gliders' decimation process. The sampling intervals of *dt* include 5 s, 10 s, 30 s, 60 s, 120 s and 240 s. The gaps between two adjacent timestamps are filled by linear interpolation to match the enquiry data structure with a fixed   of 5 s. It should be noted that the results acquired with the decimation interval of 5 s are deemed as ground truth as they are completely based on the most detailed recovery mode data. For each decimation setting, the anomaly detection result of a dive cycle (either positive or negative with anomaly) is compared with its corresponding ground truth. A result is deemed as accurate if it matches its ground truth, otherwise it is inaccurate. The anomaly detection accuracy is calculated by dividing the number of accurate detections with total detection times.

Individual sensors

In Group A, for each of the 15 sensors under investigation, at one time, one of the sensor readings are re-sampled with different decimation settings whilst other sensor data remain as originally recorded. As a result, the influences of individual sensors can be explored. The anomaly detection accuracy is calculated using all the nine test deployments, i.e. it is an overall accuracy over all the test deployments. As shown in Table 2.4, only 3 of the sensors, i.e. battery_position, rudder_angle and VBD have minor impacts on the anomaly detection accuracy, whist the decimation of all the other sensors do not present observable impacts on

the anomaly detection accuracy. It should be noted that battery_position, rudder_angle and VBD signals correspond to the actuators directly controlling the UG. Hence, this suggests that the anomalies included in the test datasets are highly likely to have caused the actuators to operate differently than in normal scenarios.

All sensors

Table 2.5 presents the sensitivity study results on decimations of all sensors. Data from all the 15 sensors are decimated simultaneously for the sampling intervals of 5 s, 10 s, 30 s, 60 s, 120 s and 240 s, respectively. Note that   s is still deemed as ground truth. The minimum anomaly detection accuracy achieved is 90.2% for the sampling interval of 240 s, which suggests that the proposed anomaly detection system is insensitive to the data decimation settings.

Table 2.4: Sensitivity study results on individual sensor decimations.

Sensor \ dt [s]	5	10	30	60	120	240
battery_position	1.000	0.998	0.992	0.986	0.981	0.972
battery_voltage	1.000	1.000	1.000	1.000	1.000	1.000
conductivity	1.000	1.000	1.000	1.000	1.000	1.000
heading	1.000	1.000	1.000	1.000	1.000	1.000
leak_detection_voltage	1.000	1.000	1.000	1.000	1.000	0.998
pitch	1.000	1.000	0.999	0.999	0.999	0.998
pressure	1.000	1.000	1.000	1.000	0.999	0.998
roll	1.000	1.000	1.000	1.000	1.000	1.000
rudder_angle	1.000	0.996	0.987	0.975	0.968	0.968
sci_pressure	1.000	1.000	1.000	1.000	1.000	1.000
state_of_charge	1.000	1.000	1.000	1.000	1.000	1.000
temperature	1.000	1.000	1.000	1.000	1.000	1.000
vacuum	1.000	1.000	1.000	0.999	0.996	0.991
VBD	1.000	0.999	0.992	0.984	0.976	0.972
vertical_velocity	1.000	0.998	0.997	0.995	0.993	0.993

Table 2.5: Sensitivity study results on decimations of all sensors.

	5	10	30	60	120	240
accuracy	1.000	0.995	0.976	0.951	0.916	0.902

2.8. Conclusions

This work presents an unsupervised anomaly detection system with an improved training procedure of existing BiGAN. Data reconstruction and discriminator feature losses are adopted as assistive hints to periodically guide effective training of the BiGAN-based anomaly detection system. A novel data augmentation strategy of multi-sensor time series data is proposed to capture the transient features within data. The anomaly detection system can provide a thorough evaluation of a dive profile using multiple data patches extracted from the profile and is flexible with dive lengths. Compared to the authors' previous work, this study has endeavoured to apply more signals with improved signal reconstruction capability provided by the enhanced BiGAN structure. Although the method is proposed for the anomaly detection of UGs, the workflow developed does not require application-specific features. Therefore, it can be adapted for other application scenarios with multivariate time series data, e.g., high-performance computing system anomaly detection and aircraft turbulence detection.

The proposed anomaly detection system is trained using two healthy Slocum G2 glider deployments. Synthetic sensor faults are injected to the training dataset to check the anomaly detection system performance. Real-world collected datasets are applied to test the proposed anomaly detection system. The test results suggest that the BiGAN-based anomaly detection has successfully detected anomalies caused by biofouling, bulky sensors, angle of list, losses of wings and strong disturbances, whilst without giving false detections for healthy deployment. The unsupervised anomaly detection system has achieved satisfactory anomaly detection performance over a fleet of underwater gliders with minimal training data preparation. A sensitivity analysis of the decimation settings has shown that the anomaly detection system is insensitive to the data decimation settings. The outcome of the work will support the over-the-horizon operation of marine autonomous systems within the National Marine Equipment Pool at the National Oceanography Centre.

Although the system can highlight the anomalies on sensor readings, it can still be challenging for humans to accurately determine the types of anomalies, including known and unknown ones. An intelligent anomaly classification system will be developed to classify anomalies automatically using deep learning in further work. The unsupervised learning anomaly detection method proposed in this study requires only healthy deployment datasets, making it generic to different types of anomalies, even unknown ones. Such a unique feature also makes it ideal for annotating dive cycles with only deployment-level anomaly information. The annotated dive cycles can be further applied to train supervised or semi-supervised fault diagnostics models.

3. Understanding

3.1. Introduction

This section focuses on the diagnosis for RAS. Part of the work, i.e. supervised learning with assistive labelling from BiGAN (Section 3.2) has been published in [75]. The work presented in Section 3.3 using transfer learning and domain adaption is being extended to journal articles with part of the results presented in Sections 4.2 and 4.3.

3.2. Supervised learning with assistive labelling from BiGAN

Introduction

Underwater Gliders (UGs) (Figure 3.1) are a type of Autonomous Underwater Vehicle (AUV) that are being used extensively for long-term observation of key physical oceanographic parameters [76]. They operate remotely at a low surge speed of approximately 0.3 m/s with deployments of several months [77]. However, developing Near Real-Time (NRT) anomaly detection and fault diagnostics systems for such vehicles remains challenging as decimated sensor data can only be transmitted off-board periodically during operations when the UG is on the surface.

Figure 3.1: Slocum G2 underwater glider with Ocean Microstructure.

As part of an ongoing collaboration, the authors have previously developed anomaly detection systems for UGs via different approaches. In [12], a simple but effective system was developed to detect the wing loss using the roll angle. In [14], system identification techniques were employed to detect changes in model parameters which further successfully deduced simulated and natural marine growth. Anderlini, et al. [78] further conducted a field test to validate a marine growth detection system for UGs using ensembles of regression trees. In [59], the use of a range of deep learning techniques was investigated to achieve over-the-horizon anomaly detection for UGs. In [32], an anomaly detection system based on an improved Bi-directional Generative Adversarial Network (BiGAN) was prototyped to enable generic anomaly detection for different types of anomalies.

For UGs operated over the horizon, some faults can only be revealed when the faulty UGs are recovered. Also, it is not clear when the faults developed. Some undetected faults can lead to critical failures and the loss of vehicle and/or data cargo. Therefore, it is essential to understand the actual causes of high anomaly scores during remote monitoring to allow operators to take appropriate mitigations to minimise subsequent risks and maximise the successful delivery of the remainder of the deployment. This work further compares the results acquired in [32] with other baseline approaches. In addition, a new supervised fault diagnostics method for UGs is proposed. The BiGAN-based anomaly detection system is

applied to estimate when the faults are developed, such that the training dataset for the supervised fault diagnostics model can be accurately annotated. The results suggest that the BiGAN-based anomaly detection system has successfully detected different types of anomalies, in good agreement with model-based and rule-based approaches. The supervised fault diagnostics system has achieved high fault diagnostics accuracy on the available test dataset.

Method

Figure 3.2 shows the workflow of the anomaly detection and fault diagnostics for underwater gliders using deep learning. This workflow comprises two parts, i.e. unsupervised anomaly detection and supervised fault diagnostics. The unsupervised anomaly detection system is developed to alert the operators about the occurrence of an abnormal vehicle status that deviates from the normal baseline operating pattern. The developed unsupervised anomaly detection system is applied to assist in annotating the datasets with anomalies, as the exact times when the anomalies developed are unknown. With this approach, the training dataset for the supervised fault diagnostics can be accurately annotated.
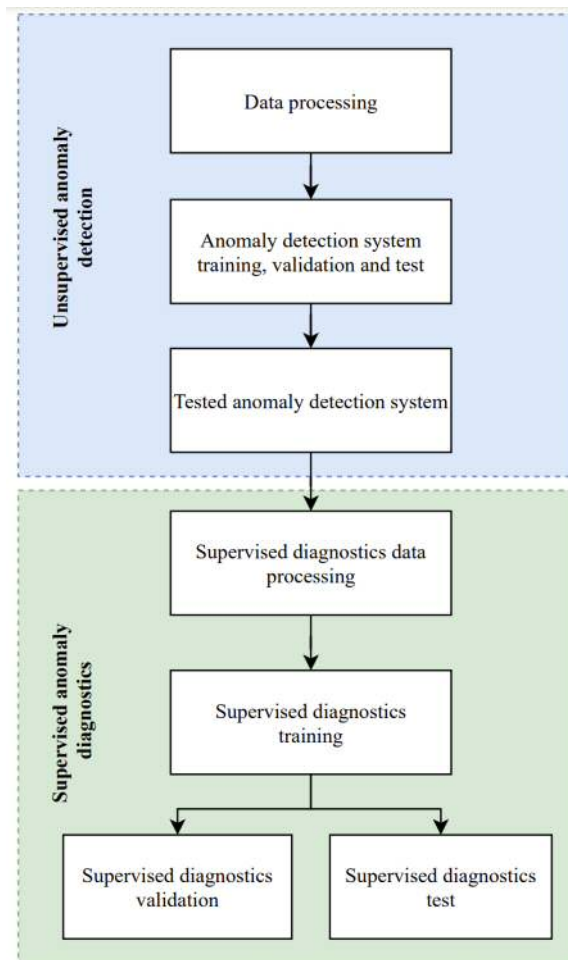
Figure 3.2: Workflow of the anomaly detection and fault diagnostics for underwater gliders using deep learning.

The supervised fault diagnostics system aims to indicate the UGs' operating status when NRT data is transmitted via satellite connection, i.e. whether the vehicles present normal

conditions or have developed a certain type of fault. Given a training dataset with training samples such that is the data patch (matrix, where is the number of signals, is the number of time steps in one data patch) and is its corresponding label (i.e. fault class), the supervised learning algorithm seeks a function that predicts the unknown class of an observation , where and are the input and output space of , respectively. For training sample , a loss function is minimised to find the function , where is the predicted label of . The function is modelled with a neural network, as shown in Figure 3.3. The original input (matrix) is flattened as the input of the neural network. Three fully connected layers (followed by their own batch normalization, LeakyReLU activation and dropout layers) forward propagate the features to a fully connected layer activated by SoftMax to output the predicted class. A cross-entropy loss function is applied to the output of the last fully connected layer.
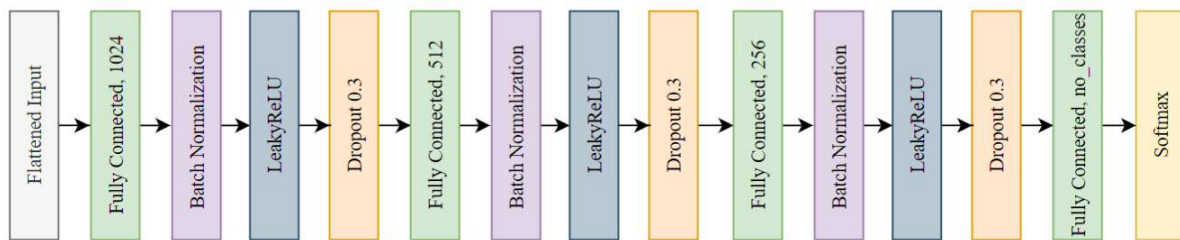
Figure 3.3: Neural network configuration for the supervised learning.

Datasets

Table 3.1 details the 11 Slocum G2 deployment datasets used in this study [32], [72]. The first two datasets collected by units 345 and 397 are applied to train the BiGAN-based anomaly detection system as those are normal deployments without any anomalies. The other datasets used for testing include one healthy deployment of unit 419, and eight deployments with anomalies including biofouling, Ocean Microstructure Gliders (OMG, which have a large sensor appendage mounted on the exterior of the hull), angle of list, loss of wings and strong disturbances. The detailed data processing procedures can be found in [32].

Table 3.1: The datasets applied for anomaly detection system training and testing.

Glider ID	Glider status	Anomaly detection
unit 345	Healthy	Training
unit 397	Healthy	Training
unit 419	Healthy	Testing
unit 399	Biofouling	Testing
unit 423	OMG	Testing
unit 424	OMG	Testing
unit 194	Angle of list	Testing
unit 304	Loss of right wing	Testing
unit 345	Strong disturbances	Testing
unit 436	Loss of left wing	Testing
unit 492	Simulated biofouling	Testing

Results

Figure 3.4 shows a sample of the validation process where only the rudder angle signal is artificially manipulated to its minimum value (-0.52 rad) and the anomaly detection has successfully annotated this anomaly. After learning the distribution of the training dataset, the model can output a high anomaly score that describes the degree of an anomaly.
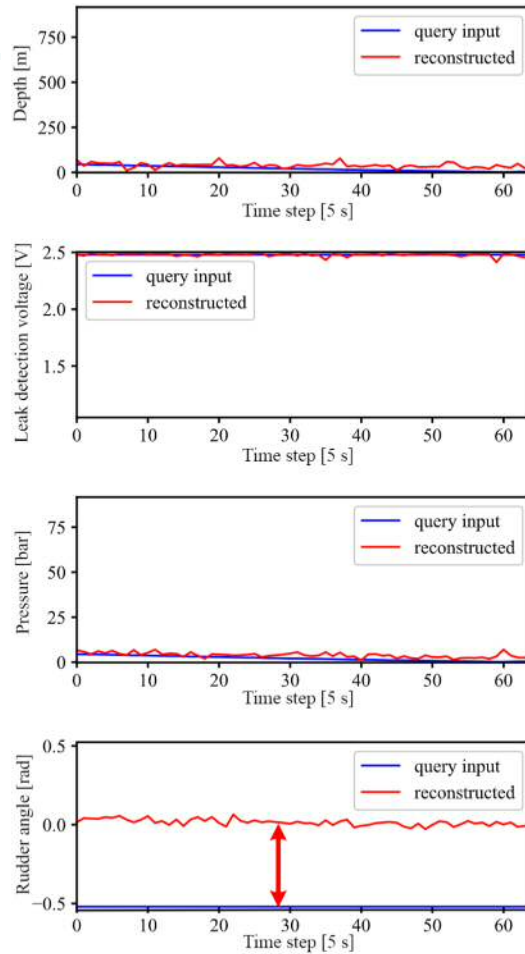
Figure 3.4: A verification sample with the rudder angle signal manually set to its minimum value while the other signals are unchanged. For the rudder angle sensor, the signal reconstructed by the BiGAN is distributed around 0 and matches the actual sensor reading, suggesting that the model has learned the distribution of the training data.

Figure 3.5 shows the anomaly detection results against the model-based and rule-based ones. It is worth noting that only seven deployments are compared here due to the availability of baseline results for the remaining deployments. The BiGAN-based anomaly detection system has correctly shown anomaly score trends in good agreement with the model-based and rule-based methods for the seven deployments. Note that for some of the deployments (e.g. unit 436: wing loss), only one baseline is presented, since the missing model-based method did not correctly show the detection metric trend.
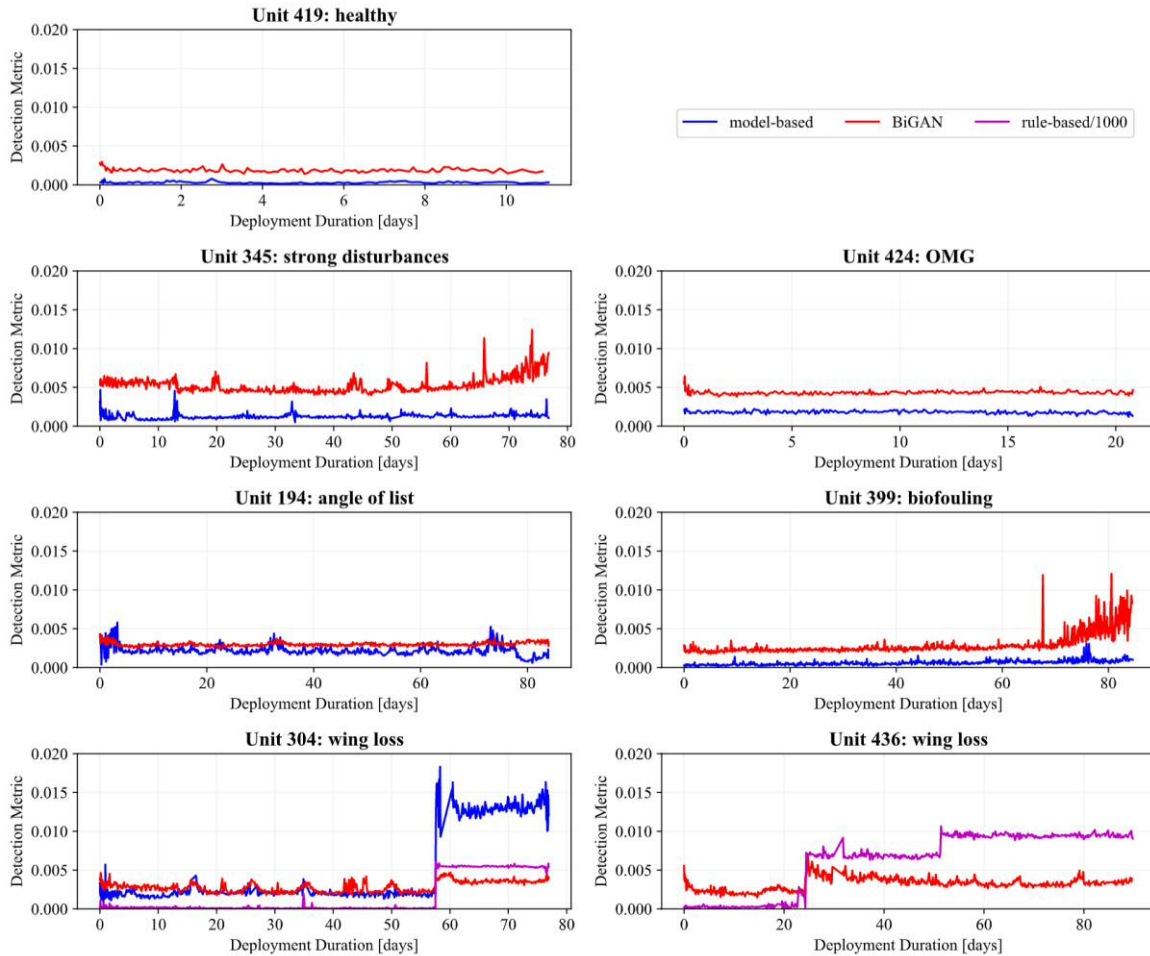
Figure 3.5: Anomaly detection results using BiGAN, compared with model-based and rule-based approaches.

Table 3.2 details the representative cycles selected from the datasets with the aid of the anomaly detection results. The chosen dive cycles are applied to the training, validation and test datasets for the supervised learning method. For each deployment, 50% of the selected cycles are randomly sampled to generate the training datasets, 25% of the cycles are applied to generate the validation dataset, and the remaining 25% of the cycles are applied to generate the test dataset. For each dive cycle, 100 data patches with 64 time steps are randomly sampled for both the training and test datasets. Note that there are 15 signals included in the original datasets. Therefore, the training, validation and test datasets include 179,200, 89,600 and 89,600 matrices with corresponding labels, respectively.

Table 3.2: Dataset annotation using unsupervised anomaly detection results.

Glider ID	Glider status	Class label	Selected cycles
unit 345	Healthy	0	[0, 400]
unit 397	Healthy	0	[0, 400]
unit 419	Healthy	0	[0, 120]
unit 399	Biofouling	1	[800, 1050]
unit 423	OMG	2	[0, 200]
unit 424	OMG	2	[0, 270]
unit 194	Angle of list	3	[0, 700]
unit 304	Loss of right wing	4	[550, 700]
unit 345	Strong disturbances	5	[0, 620]
unit 436	Loss of left wing	6	[250, 650]
unit 492	Biofouling (simulated)	1	[0, 75]

The supervised fault diagnostics model is trained on the training dataset for 10 epochs, with the Adam optimiser and a learning rate of   . The fault diagnostics accuracy on the validation dataset is 99.76%. Subsequently, the trained neural network is applied to the test dataset to detect and classify the anomalies. The overall accuracy of the model is 99.67% on the test dataset. Figure 3.6 shows the confusion matrix of the supervised fault diagnostics results on the test dataset, suggesting the model has achieved high fault diagnostics accuracy for the 6 types of faults considered, as well as healthy operating status.
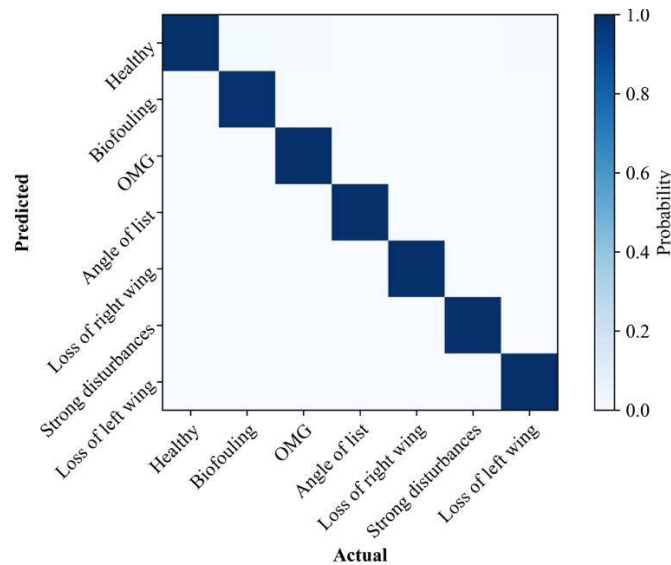
Figure 3.6: Confusion matrix of the supervised fault diagnostics results on the test dataset.

Although high fault diagnostics accuracy has been achieved over the test dataset, the fault diagnostics model is trained with datasets collected from only a few deployments. Training the model with such a small dataset can lead to an overfitted model. An overfitted model could memorise features specific to an individual deployment or a particular vehicle, which could lead to degraded fault diagnostics performance in real applications. Due to limited data available, further investigations were not conducted. In future work, the training dataset will be enriched. In addition, automatic segmentation methods will be developed to annotate the deployment datasets.

Conclusions

This work has further extended the BiGAN-based anomaly detection system developed in [32] to assist with the annotation of UG deployment datasets. The performance of the BiGAN-based anomaly detection system has been compared with rule-based and model-based methods. The annotated deployment datasets were applied to train a supervised learning fault diagnostics model. With the limited availability of training samples of the different anomalies, the supervised learning model has achieved an accuracy of 99.67% in fault diagnostics. In further work, supervised and semi-supervised models will be developed based on larger datasets with better diversity.

3.3. Transfer learning and domain adaption

This work proposes a novel fault diagnostics deep learning model to diagnose faults for Marine Autonomous System via domain adaption and transfer learning. The proposed model, i.e. Marine Autonomous System Net (MASNet), is applied to address the challenging fault diagnostics tasks for distinct types of underwater gliders that are under-observed and remotely operated in different regions and tasks by different institutions. Based on the improved Bidirectional Generative Adversarial Networks (BiGAN) developed in our previous study [32], we further extract invariant features from both the source and target domains, such that that model can detect unseen faults in the target domains with only a limited number of categories of data for training. The fault diagnostics results are evaluated against rule-based results. The MASNet show effectiveness in generalising invariant features present in the source and target domain datasets collected by distinct underwater gliders operated by different institutions in different regions hence achieving high fault diagnostics performance in the field test.

Problem statement
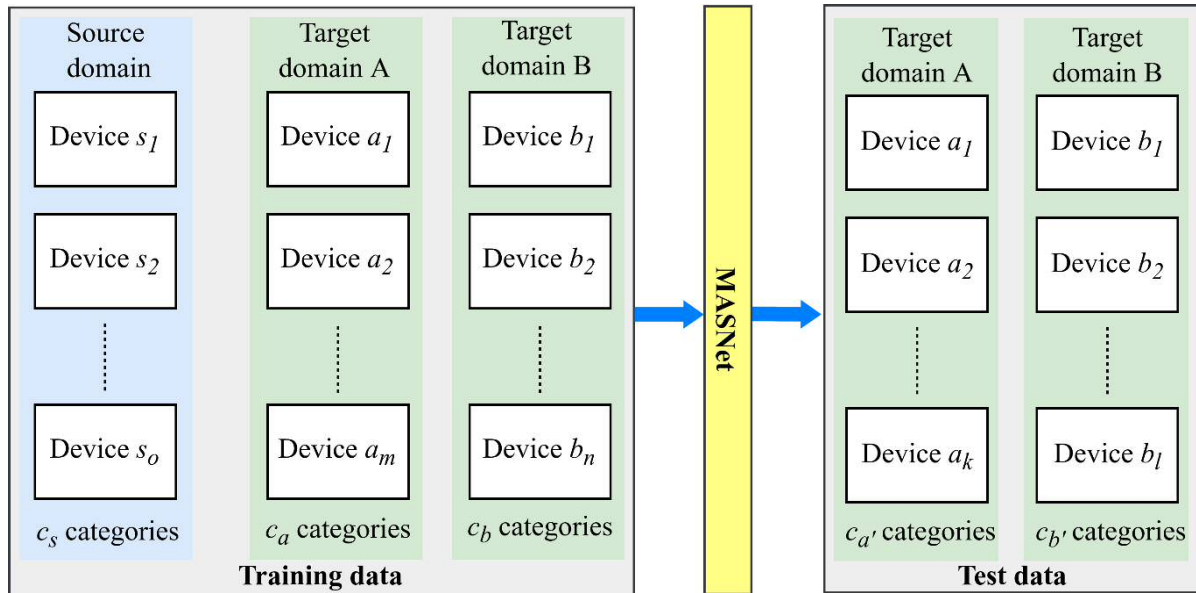
Figure 3.7: The source domain data is collected by devices labelled as   in different missions and has   different categories. The target domain data is collected by devices of distinct types in different deployments and can be operated by different institutes with distinct settings. The proposed MASNet learns from the source domain and target domain data to classify the categories of the test data which may not be present in the training data.

As shown in Figure 3.7, both the source and target domain training data are sparsely labelled. In the training data, the source domain includes   categories denoted by a set   , the target domain A includes   categories denoted by a set   , and the target domain B includes   categories denoted by a set   In the test data, the target domain A includes   categories denoted by a set   , and the target domain B includes   categories denoted by a set   . Note that:

$$(Z_a \cup Z_{a'}) \subseteq Z_s \qquad\qquad 3.1$$

$$(Z_b \cup Z_{b'}) \subseteq Z_s$$

Namely, the source domain includes more categories, while the target domains only have a limited number of classes that can be applied to extract domain-invariant and category-invariant features to classify both seen and unseen categories in the target domain. Note that the number of target domains can be expanded.

Method

Figure 3.8 details the schematic of the proposed MASNet, based on the improved BiGAN based anomaly detection framework developed in our previous work [32]. In addition to the BiGAN loss, discriminator feature loss and reconstruction loss, two additional losses are added to the framework. The feature clustering loss function aligns the features from the source and target domains encoded by   . The classification loss is a typical cross-entropy function for classification tasks.
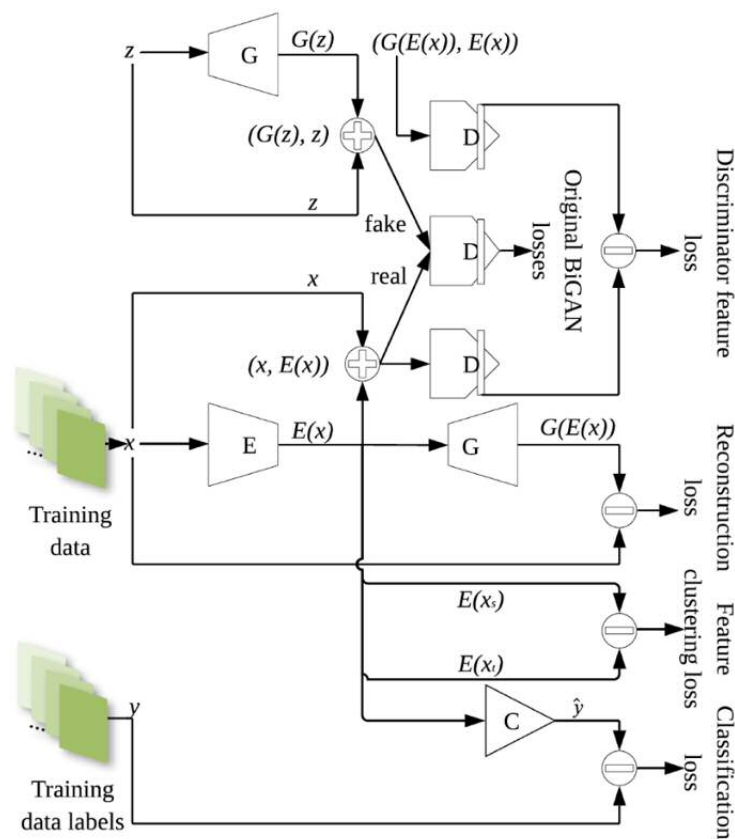
Figure 3.8: Based on an improved BiGAN-based anomaly detection model proposed in our previous study ([32], also see Section 2 , we add an additional classifier to the model. A feature clustering loss term is added to align the categorical features of the source and target domains, using the encoded latent information from the encoder   .

Datasets

Table 3.3 details the datasets of source and target domains. The source domain data comprises 10 datasets collected by Slocum G2 underwater gliders operated by NOC and one operated by PLOCAN. Seven types of operating status have been labelled manually with the anomaly detection results presented in [32]. The target domain data contains four different operating statuses, i.e. healthy, loss of right wing, loss of left wing and angle of list, which is collected in the project FRONTIERS in Mallorca, in collaboration with SOCIB. Only healthy operating data collected by SOCIB glider is included in the training to represent the target domain. The other three simulated faults are applied to the fault diagnostics model performance. The data patches, i.e. the elements of both the training and test datasets, are augmented as described in [32]. Each data patch is a   matrix, where   and   are the numbers sensors and timesteps. The selected values are   and   , considering the available common signals from both domains the results presented in [32]. Both the target and source domain datasets are augmented by randomly sampling 100 samples from each of the valid driving cycles.

Table 3.3: The source domain data applied in this study is measured by a number of Slocum G2 gliders over eleven developments. The source domain data is the most thorough data with six different types of faults and has been applied in [32]. The target domain data are collected by a Slocum G2 variant model operated by SOCIB in Mallorca. Note that the unit operated by SOCIB has an underline internal condition throughout tests done in Mallorca, which has likely shifted the data pattern collected by it further away from the ones operated by NOC.

No.	ID	Date	Project	Organisation	Location	Status	Purpose	Domain
s_1	unit 345	2014	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	Healthy	training, validation, test	source
s_2	unit 397	2015	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	Healthy	training, validation, test	source
s_3	unit 419	2015	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	Healthy	training, validation, test	source
s_4	unit 399	2015	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	Biofouling[1]	training, validation, test	source
s_5	unit 423	2015	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	OMG	training, validation, test	source
s_6	unit 424	2015	AtlantOS, CaNDyFloSS	NOC	the Celtic Sea	OMG	training, validation, test	source
s_7	unit 194	2017	ALTERECO	NOC	the North Sea	Angle of list	training, validation, test	source
s_8	unit 304	2019	ALTERECO	NOC	the North Sea	Loss of right wing	training, validation, test	source
s_9	unit 345	2019	ALTERECO	NOC	the North Sea	Strong disturbances	training, validation, test	source
s_{10}	unit 436	2019	ALTERECO	NOC	the North Sea	Loss of left wing	training, validation, test	source
s_{11}	unit 492	2020	IDUG	PLOCAN	Gran Canaria	Biofouling[2]	training, validation, test	source
t_1	unit SOCIB	2021	FRONTIERS	SOCIB	Mallorca	Healthy	training	target
t_2	unit SOCIB	2021	FRONTIERS	SOCIB	Mallorca	Loss of right wing	test	target
t_3	unit SOCIB	2021	FRONTIERS	SOCIB	Mallorca	Loss of left wing	test	target
t_4	unit SOCIB	2021	FRONTIERS	SOCIB	Mallorca	Angle of list	test	target

Table 3.4 details the signals applied to augment the datasets of this study. Note that the signals names are unified to the notations included in the SOCIB datasets. The signals of roll rate, pitch rate, rudder rate and vertical velocity are derived from the corresponding signals. The sample interval is 5 s.

Table 3.4: Signals applied to augment the fault diagnostics datasets. The signals of roll rate, pitch rate, rudder rate and vertical velocity are derived from the corresponding raw signals.

Signal	SOCIB symbol	Unit
Roll	m_roll	rad
Pitch	m_pitch	rad
Rudder angle	m_fin	rad
VBD	m_de_oil_vol	cm^3
Battery voltage	m_battery	V
Vertical velocity	vertical_velocity	m/s
Roll rate	roll_rate	rad/s
Pitch rate	pitch_rate	rad/s
Rudder rate	fin_rate	rad/s

Results

Figure 3.9a details the fault diagnostics results of the proposed MASNet applied to the target domain data collected in the FRONTIERS project in collaboration with SOCIB (also see Section 4.2 for the test settings and anomaly detection results using BiGAN). It is apparent that the MASNet has correctly detected the wing loss scenarios (in comparison with the rule-based method of using the mean roll angle difference between the ascents and descents shown in Figure 3.9b. In the early stage of the test deployment (until mid of 2021-07-07), the MASNet has correctly labelled the glider's status as healthy. Note this data collected over this period has been applied to align the invariant features between the target and source domains. The three scattered points (angle of list) are the times when the pitch angle was manually changed. In the later stage of the test, the MASNet has correctly labelled the angle of list status (simulated by adjusting balancing pills on the starboard and port sides). The final stage is mostly labelled as either healthy or angle of list, which is mainly due to multiple incorrect trim and ballast settings were applied over this period, making the vehicle's real status has never presented in normal scientific missions.

It is worth noting that without applying domain adaption procedures, using purely the data collected by NOC, extensive efforts that were made to detect the faults simulated in the FRONTIERS project did not achieve satisfactory fault diagnostics performance for a distinct glider operated by SOCIB with a minor internal oil leak. Extensive experiments suggest that aligning the invariant features between the source and target domains has significantly improved the fault diagnostics performance of the MASNet.

Nonetheless, as presented in Section 4.3, when the variance of data pattern between the source and target domains are insignificant, the classifier without domain adaption steps but with the encoder, generators and discriminator pre-trained by BiGAN shows reasonable fault diagnostics performance in two types of missions (mapping and profiling).
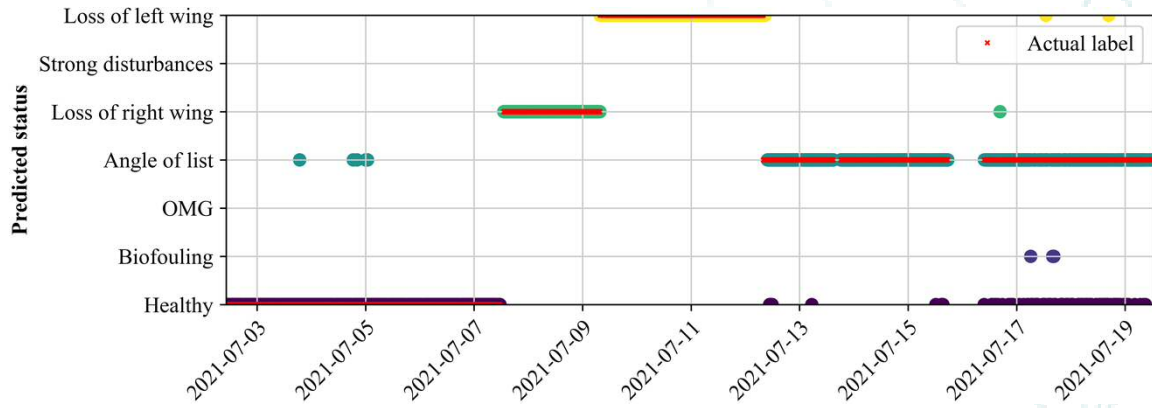
(a)
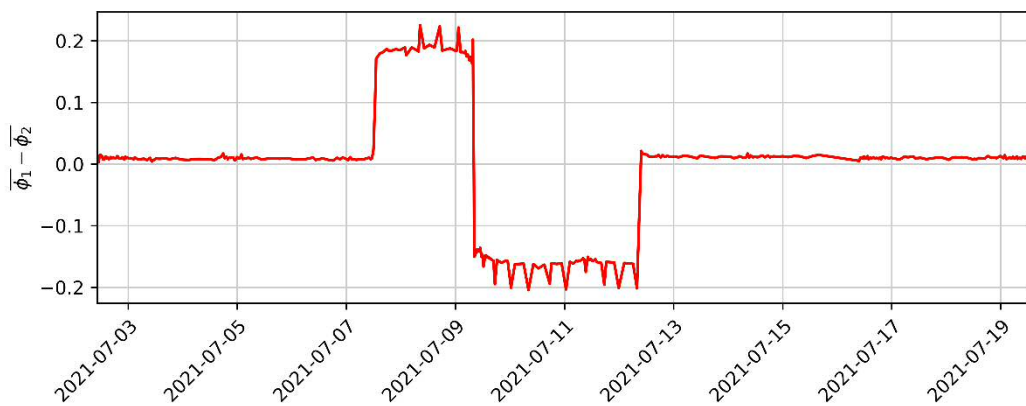
(b)

Figure 3.9: Fault diagnostics results: (a) predicted operating status of the Slocum G2 UG operated by SOCIB in the FRONTIERS project test (also see Section 4.2), (b) rule-based fault detection using the difference of the mean roll angle during the steady-state portion of ascents (   ) and descents (   ) [12], which is particular sensitive in detecting faults caused by loss of wings but cannot indicate other faults correctly. The rule-based detection of wing losses results are used to assess the MASNet model performance in detecting wing losses. The fault diagnostics results are in line with anomaly detection results presented in Section 4.2.

Conclusions and future works

The fault diagnostics results presented in this Section suggest that proposed MASNet is effective in generalising invariant features present in the source and target domain datasets collected by distinct underwater gliders operated by different institutes in different regions hence achieving high fault diagnostics performance in the field test.

The method and results presented in this Section (3.3) are being extended to journal articles with more comprehensive details and comparative studies against existing studies. Additionally, semi-supervised elements (e.g. using MixMatch [79]) will be included in the journal articles to further address the impacts of limited and sparsely labelled training datasets.

4.   Verification and Operational Implementation

4.1.   Introduction

This section presents the results that were obtained from two projects, namely FRONTIERS project in Section 4.2 and FEATURE project in Section 4.3, where we verify the BiGAN algorithm that we had developed on actual field test deployments. In particular, the tests on the FRONTIERS project were carried out by artificially replicating faults on the gliders, whereas for the FEATURE project, we showed that the algorithm also work on an AUV, a different type of vehicle via transfer learning.

In Section 4.4, we present the development being done for implementing a condition monitoring system and deploying the algorithms for operational usage. We also discuss about strategies to improve the robustness of ML systems running in production.

4.2.   FRONTIERS test

Project information

Proposal reference number	21/1001599
Project Acronym (ID)	FRONTIERS
Title of the project	Fault detection, isolation and Recovery fOr uNderwaTer glIdERS
Host Research Infrastructure	SOCIB, ES
Starting date - End date	02/07/2021 - 19/07/2021
Name of Principal Investigator Home Laboratory Address E-mail address	Enrico Anderlini, Marine Research Group, Department of Mechanical Engineering, University College London Roberts Engineering Building, London, WC1E 7JE, UK E.Anderlini@ucl.ac.uk

Test objectives

The updated aim of the project is to validate methods for the smart anomaly detection and fault diagnostics for underwater gliders. The project outcomes will help increase the reliability of these platforms and help over-the-horizon pilots to monitor the conditions of these systems.

The project aim will be achieved through the following updated objectives:

O1 Introduction of data-driven methods for the anomaly detection and fault diagnostics of MAS (as part of project ALADDIN funded by the Assuring Autonomy International Programme, a partnership of Lloyds' Register Foundation and the University of York);

O2 Validation of the tools with the actual field test of an underwater glider for the following case studies:

- sudden wing loss;
- incorrect ballasting and trimming.

Main achievements and difficulties encountered

The project has successfully validated the introduced anomaly detection and fault diagnostics methods. The glider has been deployed, recovered and redeployed multiple times to simulate the loss of either wing, incorrect ballasting (through the addition or removal of weight pills) and incorrect trimming (through the addition or removal of the weight pills along the length of the vehicle as well as different settings of the internal battery position). Furthermore, the glider had additional intrinsic anomalies: slow leak in the thermal valve of the variable buoyancy device, a small offset in the CTD sensor readings and high energy consumption levels.

The simulated faults were correctly detected and identified, whilst the intrinsic smaller faults will provide additional training data to expand the system in the future. Validation of the diagnostics of the smaller intrinsic anomalies are not done as these anomalies were not present in the training data, obtained from many past glider deployments, used to train the model. Further verifications can be done by characterising the goals of a robotic autonomous system (RAS) along with the group of sensors and external data sources in use to the sensing capabilities that they could provide. This will help in defining the verification steps in terms of understanding how available features from training datasets would map to the sensing and understanding requirements of the RAS. This is especially useful when applying deep neural network (DNN) methods to cross-domain sensing and understanding capability.

The main difficulties encountered concerned the global pandemic, which prevented the UCL team to travel to Mallorca due to the constantly changing travel rules. However, this problem was solved thanks to the professionalism of the SOCIB team, their user-friendly data exchange portal and regular email exchanges or calls. Bad weather before the project start meant that the project actually began a few days later than expected.

Dissemination of the results

The project has been advertised on LinkedIn with two posts with 1,749 total views on 23/07/2021 and to the AAIP.

Further planned dissemination activities involve:

- Open-access publication of the collected data on the SOCIB data portal https://www.socib.eu/?seccion=observingFacilities&facility=glider, https://thredds.socib.es/thredds/dodsC/auv/glider/sdeep01-scb_sldeep001/L0/2021/dep0036_sdeep01_scb-sldeep001_L0_2021-07-02_data_dt.nc.html,
- Publication of one collaborative journal article in the Journal of Field Robotics or IEEE Journal of Oceanic Engineering,
- Use of the results in up to three additional journal article publications as part of project ALADDAIN, https://www.york.ac.uk/assuring-autonomy/projects/unmanned-marine-systems-safety/,
- Inclusion of the project outcomes within the AAIP's Body of Knowledge entries 2.2.4.1 – Verification of sensing requirements, 2.2.4.2 – Verification of understanding requirements, https://www.york.ac.uk/assuring-autonomy/body-of-knowledge/,
- Advertisement on SOCIB's twitter account, https://twitter.com/socib_icts/status/1417784285264760833,

Further advertisement on the principal investigator's LinkedIn account once the results are postprocessed.

Technical and Scientific preliminary Outcomes

The project mission is summarised in Figure 4.1, which shows the GPS coordinates of the glider during the deployment, the number of days spent at sea, the number of profiles undertaken, the distance covered, the maximum depth reached and the average surge speed.

Figure 4.1: Summary of the glider deployment for the FRONTIERS project.

As shown in Figure 4.2A, the test started at t_0 (Figure 4.2B-a) and ended at t_8 (Figure 4.2B-b). The anomaly detection system based upon Bidirectional Generative Adversarial Networks (BiGAN) has successfully output anomaly scores over the test. The pitch angles for t_0-t_1, t_1-t_2, and t_2-t_3 were set as 30°, 18°, and 26°, respectively. The glider's starboard wing was removed at t_3 (see Figure 4.2B-c). At t_4, the starboard wing was restored while the port wing was removed (see Figure 4.2B-d). At t_5, the port wing was restored while the balancing weight setting in the wing rails was adjusted from left-2 & right-5 to left-5 & right-2 (each pill is 15.5 g) (see the vehicle status in Figure 4.2B-e). At t_6, the wrong battery position was applied. At t_7, the battery position servo mode was set, and the balancing weight setting was changed to left-0 & right-3 (2 extra pills removed along the length of the vehicle in each wing rail, see Figure 4.2B-f). The glider was recovered at t_8.

A data-driven anomaly detection system based on a BiGAN architecture with added hints was trained with data from deployments from the British Oceanographic Data Centre and the SOCIB portal. The system uses the decimated semi-real-time data signals from each dive of the glider sent ashore to calculate an anomaly score that can be used to determine whether anomalies are present on board the vehicle. Once trained, the system was validated using the data stream from the JERICO deployment. As can be seen in Figure 4.2A, as the 30° and 18° pitch settings were not included in the training dataset, high anomaly scores have been incorrectly returned at the start of the deployment for normal behaviour. However, the system was able to clearly detect the loss of wing, as removing the starboard and port wings resulted in high anomaly scores of similar magnitudes. Additionally, relatively high anomaly scores can be observed from t_5 to t_8 for the incorrect ballasting and trimming. In conclusion, the simulated faults were correctly detected, validating the proposed anomaly detection solution, although further work is needed to address the false positive at the start of the deployment. This will be tackled through data augmentation.
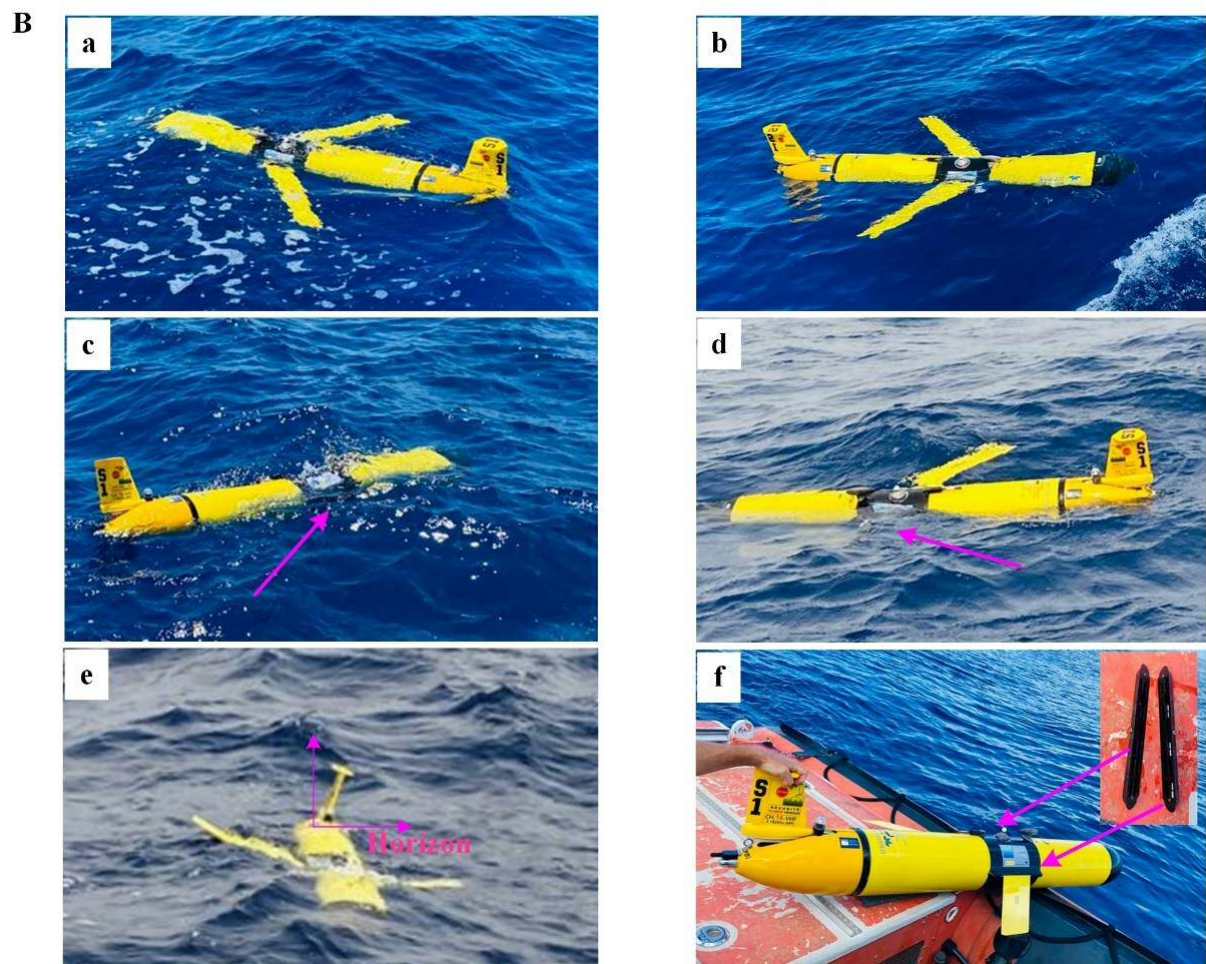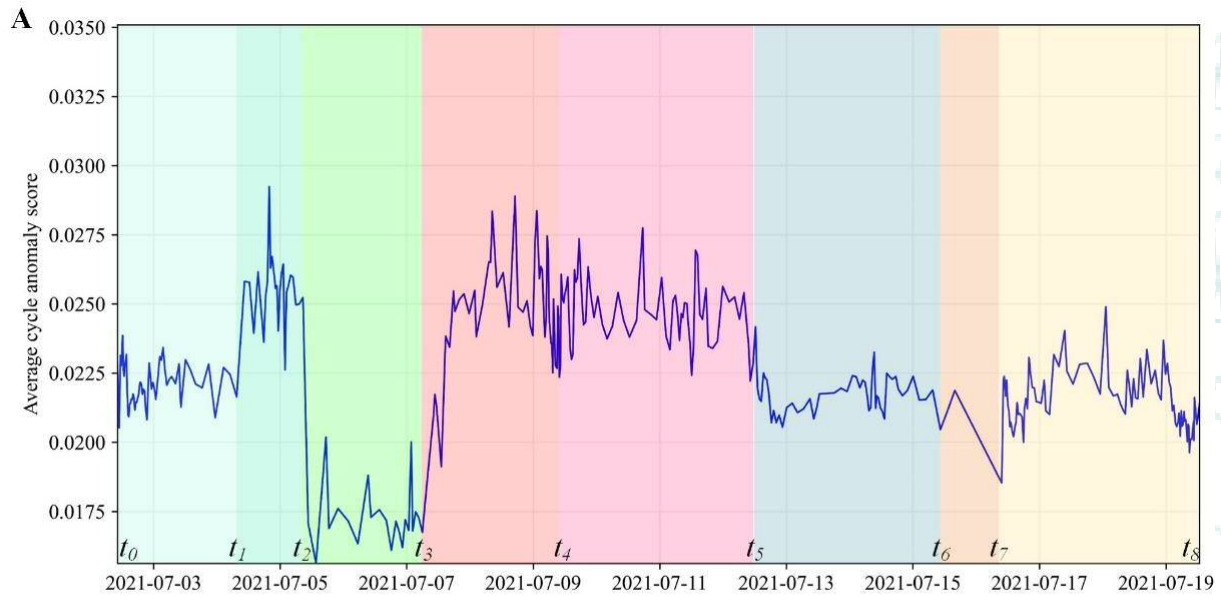
Figure 4.2: (A) anomaly scores over the test. (B) a: the glider at the beginning of the test, b: the glider before recovery at the end of the test, c: the glider with its starboard wing removed, d: the glider with its port wing removed, e: incorrectly ballasted glider, f: the balancing weight setting for the simulated trimming fault.

4.3. FEATURE test

Project information

Project Details	
AGREEMENT No [1]	EUMR-TNA-UPORTO-Local-03
Title	Fault dEtection, isolation and recovery for AuTonomous UnderwateR vEhicles
Acronym	FEATURE
Starting Date - End Date	16-08-2021 **to** 03-11-2021
Infrastructure Access / Use Case Details	
Access / Use Case code	UPORTO-Local
Applicant Institution	
Name	University College London
Country	United Kingdom
Domain	Academia
Principal Investigator	
Name	Enrico Anderlini
Email Address	E.Anderlini@ucl.ac.uk
Team Members	
Name	Davide Grande
Email Address	davide.grande.19@ucl.ac.uk
Name	Peng Wu
Email Address	peng.wu.14@ucl.ac.uk

Project objectives

The project aims to develop a new method that detects, diagnoses and notifies the occurrence of faults on autonomous underwater vehicles (AUVs) via domain adaption and transfer learning (updated due to the global pandemic of COVID-19). The project outcomes

will help increase the reliability of these platforms, thus contributing to the assurance of the operations of unmanned marine systems (UMS).

The project aims to achieve the following objectives:

O1 Development of the method for the detection of faults in the target domain using source domain data collected by similar but different AUVs and/or simulations

O2 Verification of the developed method using historical data

O3 Validation of the developed methods for two case studies with actual deployments of individual AUVs undertaking different tasks with emulated faults and normal operating status:

- a mapping AUV with stuck top and bottom vertical servomotors
- a profiling AUV with stuck top and bottom vertical servomotors

Main achievements and difficulties encountered

A novel data-driven anomaly detection and fault diagnostics method has been developed and validated through field tests. The method is based upon an improved Bidirectional Generative Adversarial Network (BiGAN) but has been further extended to deal with smart fault diagnostics for UMS with limited training data via transfer learning, achieving high fault diagnostics accuracy in the target domain. More specifically, using historical data collected distinctive AUVs in different missions, the anomaly detection model has successfully notified the occurrence of anomalies; the fault diagnostics model built upon the anomaly detection model has achieved good fault diagnostics performance.

Further verifications can be done by characterising the goals of the robotic autonomous system (RAS) along with the group of sensors and external data sources in use to the sensing capabilities that they could provide. This will help in defining the verification steps in terms of understanding how available features from training datasets would map to the sensing and understanding requirements of the RAS. Additionally, this information can be added to the metadata to enable future improvement of the system when updating the model as new datasets and data sources become available. These steps are particularly useful when applying deep neural network (DNN) methods to cross-domain sensing and understanding capability.

Two main challenges were encountered:

- Due to the global pandemic of COVID-19, the team could not travel to Porto to deploy and test the originally proposed algorithm run in real-time on the on-board CPUs of the LAUVs. Therefore, the aim & objectives have been updated to develop and validate the novel data-driven deep learning method (see Section 2.1).
- The initial deployment of lauv-xplore-1 experienced a failure of the vehicle's CPU. Consequently, the mission was postponed to November 2021. The subsequent test was completed using a similar but with a different LAUV, i.e. lauv-xplore-2, which was beneficial to test the developed model performance in the target domain.

Dissemination of the results

The methods and outcomes of the project will be published in a manuscript in a high impact journal, namely the Journal of Field Robotics. Additionally, the results will be presented in a seminar internal to UCL open to undergraduate and postgraduate engineering students. Furthermore, the outcomes of the project will be used to influence the future regulations of UMS by contributing to the body of knowledge on assuring autonomy as part of project Assuring Long-term Autonomy through Detection and Diagnosis of Irregularities in Normal operation (ALADDIN) funded by the Lloyd's Register Foundation.

Technical and Scientific preliminary Outcomes

The project aims to test the anomaly detection and fault diagnostics models developed for UMS using transfer learning and domain adaption. The models are trained to work in the target domain, i.e. previously unseen AUVs undertaking unseen tasks, using the source domain data collected by similar but distinctive AUVs and/or simulations. Figure 4.3 shows the Neptus mission plans for the 10 test runs (5 for mapping, the other 5 for profiling) with normal behaviour, and emulated faults (stuck top vertical servomotor and stuck bottom servomotor), implemented on 3 November 2021, using lauv-xplore-2. The mission data was applied to test the anomaly detection and fault diagnostics models that have been trained primarily using the source domain data collected on 16 August 2021, using lauv-xplore-1. Note that the data collected in 115641_mappingv2 (a mapping mission) appeared incomplete and was not included in this report.

(a)   (b)

Figure 4.3: Neptus mission plans for the mapping (a) and profiling (b) runs.

Figure 4.4 details the anomaly detection results of the mapping (a) and profiling (b) test runs on 3 November 2021. Missions *114025_mappingv2* and *121217_mappingv2* did not include any anomaly; therefore, low anomaly scores can be observed for the two missions. Missions *132840_mappingv2-faultS3* and *134517_mappingv2-faultS0* were completed with emulated faults, i.e. stuck top vertical servomotor and stuck bottom vertical servomotor, respectively. Apparent high anomaly scores can be observed for these two missions. For the profiling runs, stuck vertical servomotors (either top or bottom) faults have been emulated in the missions of *152318_profiling_1200rpmv3-faultS3*, *161057_profiling_1200rpmv3-faultS3v2* and *155611_profiling_1200rpmv3-faultS0*. Note that the simulated faults are not always observable as the servo motors are not always required to be running, i.e. only required for

some manoeuvring operations. Consequently, sudden anomaly score spikes can be observed at the times when a servo motor is supposed be operating freely. The BiGAN-based anomaly detection model had successfully notified the anomalies, matching the test logs.

(a)

(b)
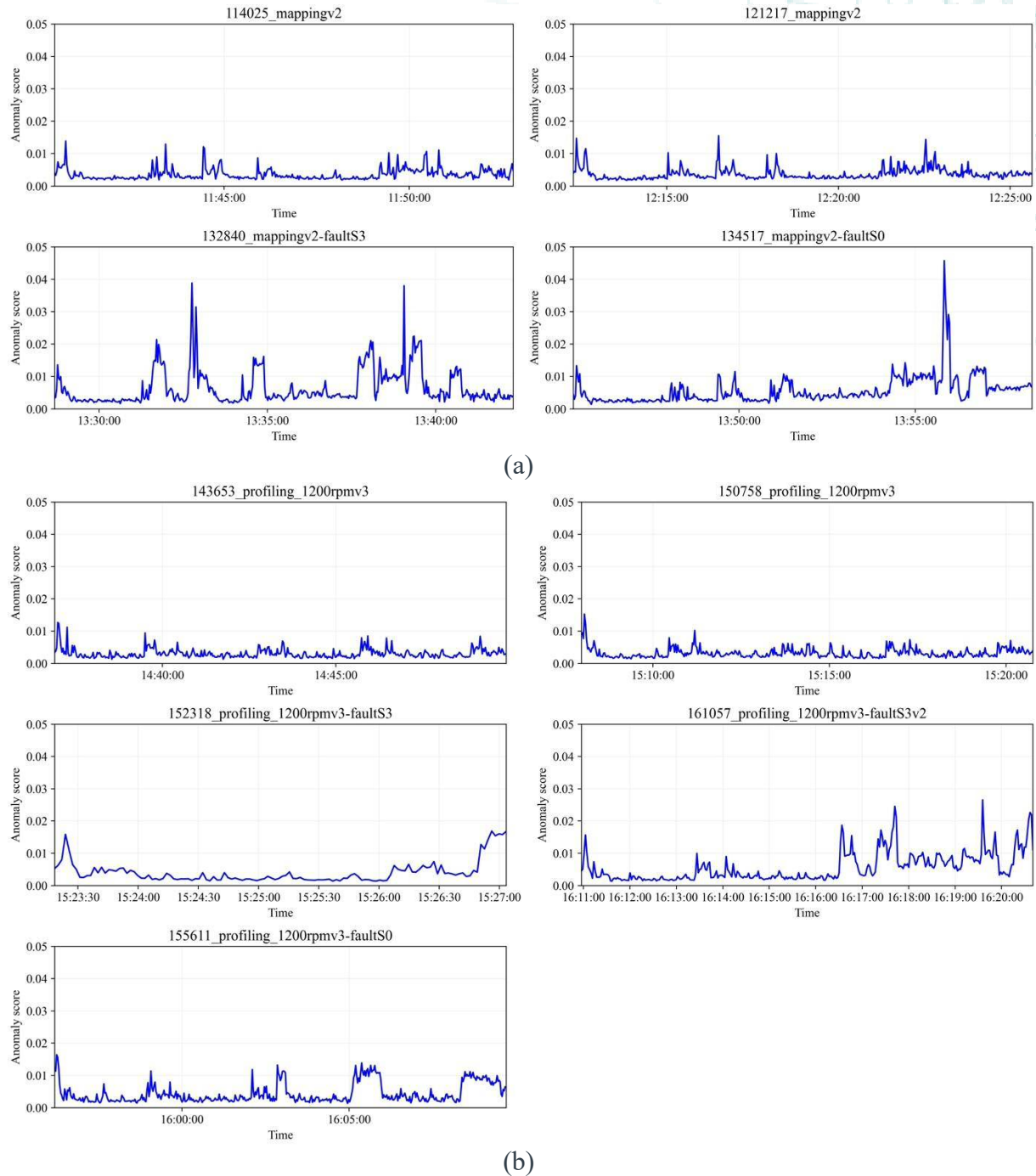
Figure 4.4: Anomaly detection results of the mapping tests (a) and profiling tests (b).

Based upon the tested anomaly detection model, classification layers were added to form the fault diagnostics model. The test data collected on 16 August 2021 were labelled and subsequently applied to train the fault diagnostics model. Note that during fault diagnostics model training, the original BiGAN component parameters were frozen, i.e., only the classification layers were updated. The BiGAN components extract essential features from the

input data and improve the fault diagnostics performance. Figure 4.5 presents the preliminary fault diagnostics results for the mapping and test runs. Without hyperparameter tuning, the model has achieved reasonable performance. In the planned manuscript, the results will be updated with performance tuning, as well as comparative studies with existing methods.

(a)   (b)
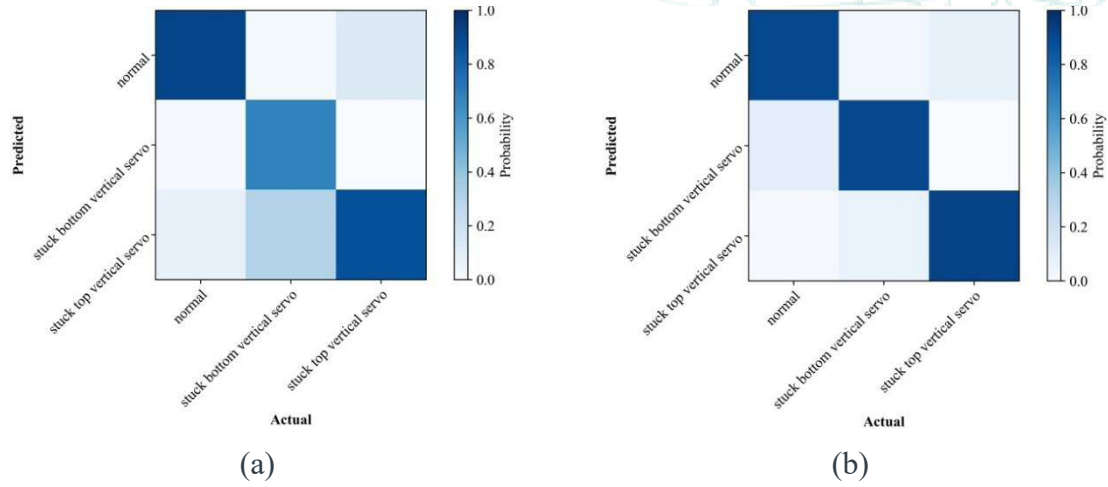
Figure 4.5: Fault diagnostics confusion matrices for the mapping (a) and profiling (b) test runs.

4.4.   Operational Condition Monitoring System

Introduction

The Marine Autonomous and Robotic Systems group (MARS) at the National Oceanography Centre (NOC) in Southampton, UK, has one of the world's largest fleets of marine autonomous systems (MAS) used for scientific and industrial purposes. The fleet, as seen in Figure 4.6, comprises both short-range autonomous underwater vehicles (AUVs) and remotely operated vehicles (ROVs), as well as long-range autonomous surface vessels (ASVs) and underwater gliders (UGs), piloted over-the-horizon by remote expert operators.

Figure 4.6: MARS fleet with the RRS Discovery by the National Oceanography Centre.

This large and diverse range of vehicle fleets increases operational complexity and creates a demand for scalable and autonomous approaches for fleet monitoring and management during operations. To address this need, several approaches were studied and developed for UGs to detect adverse behaviours and monitor the condition of the remote vehicles [12], [14], [28], [32], [33], [54], [59], [75], and [78], e.g., to detect biofouling, sensor anomalies, and wing loss by analysing past vehicle data. These algorithms include heuristics, model-based, and data-driven methods such as machine learning (ML) and deep learning

The advancements of anomaly detection techniques in the marine autonomous systems (MAS) domain enable greater autonomy of the increasingly long-range and long-endurance operations of UGs and improved data delivery. They allow glider operations to be more efficient and cost-effective, enabling one pilot to supervise the operation of many vehicles, detecting issues before they result in loss of data or the vehicle itself, and enabling early mitigating actions to be implemented. Whilst existing research has proven very promising when analysing past data downloaded from the vehicles upon recovery, to fully utilise data-driven techniques within real-world real-time operations, key assumptions made in the original works, such as volume and resolution of data, need to be revisited.

The integration and adoption of these technologies into piloting systems and standard operating procedures also require thorough testing and analysis to ensure the continuous robustness and reliability of the system. In this section, we discuss the key challenges encountered when translating data-driven approaches to real-world UG operations and present a general ML pipeline, as a framework designed to ease the integration of such approaches into MAS piloting tools, creating future scalability.

Underwater Gliders Operation

UGs at NOC are piloted through a web-based command and control interface (named 'C2') [37], [38]. The UGs communicate to the C2 via Iridium satellite, which requires the UGs to resurface and stay afloat throughout the low-bandwidth data transmission process. Communicating with the satellite is costly in terms of both data transmission fees and battery power. Importantly, extended periods spent on the surface also put the vehicle at increased risk, e.g., through increased exposure to shipping routes. Together, these constraints heavily restrict the amount of information that the UGs can send to the base station. Therefore, raw data from the UGs are decimated or down-sampled before transmission as near-real-time (NRT) data to the C2. The full datasets (known as recovery-mode data) are often only available upon physical recovery of the vehicle. The NRT data enables scientists and pilots to observe a small sample of scientific and engineering data collected by the vehicle for monitoring and to inform any interventions.

At present, pilots monitor the decimated NRT data manually via the C2 user interface. Correctly interpreting the behaviour of the UGs requires significant pilot expertise. As a result, the training requirements are significant, especially for a pilot to operate multiple vehicle types and models within the fleet.

Challenges

Utilising the decimated NRT data for condition monitoring with machine learning proves to be challenging. To date, most ML algorithms designed for MAS are reliant on high-resolution recovery-mode data and additionally, larger quantities of samples for deep learning algorithms. Consequently, the decimation process can significantly affect the accuracy of the algorithms, reducing their suitability for online usage. Running these algorithms onboard the UG is not currently considered to be an option, due to both power requirements significantly reducing endurance and limitations on computational resources. Thus, it is necessary to either achieve an acceptable accuracy when running these algorithms on decimated data or to develop energy-efficient algorithms that can run onboard the vehicle.

To build operator confidence and thus aid the adoption of data-driven approaches, we also need to maintain a consistent and comparable performance of the condition monitoring algorithms between different deployments of the UGs. The performance of the algorithms may vary depending on mission specifications such as dive profiles and data decimation requirements. The accuracy can also be affected by numerous other factors in a dynamic environment, along with anomalies novel to the condition monitoring system. The objective of optimising the algorithms' performance may even be dependent on the stakeholder preferences or application-specific requirements, such as energy efficiency, low latency, interpretability, or scientific data collection.

UNIVERSITY
of York

ALADDIN Technical Report

Copyright © 2022 University of York

Page 74

ASSURING
AUTONOMY
INTERNATIONAL PROGRAMME

Therefore, as a requirement, the ML systems or infrastructure design will need to have the ability to support frequent and continuous updates and deployment of ML algorithms for production use. However, these challenges are not unique to the MAS domain. Deployed ML algorithms in a production environment will encounter different distributions of data in the real-world, unseen during training. Developers will need to ensure that the algorithms can generalise well in a production environment and, in the long run, tackle the issue of drift in the performance of the algorithms. Moreover, the software engineering field also provides best practices that could be incorporated into an ML systems design such as the DevOps process and the continuous integration and continuous delivery (CI/CD) development philosophy.

ML Pipeline

Considering the identified challenges and requirements, we have developed an ML pipeline, designed for the MAS domain, to facilitate ML adoption for condition monitoring. The ML pipeline manages the data flow and deployments of ML algorithms for both research and production environment. Each component or task (e.g., data ingestion, data validation, data pre-processing, model training, model validation, and model deployment) in a pipeline is designed to be modular with clear and general interfaces between them. This enables continuous integration and improvement, while also allowing open collaboration where each component can be developed and owned by developers, scientists, and engineers with different expertise.

There can be multiple pipelines running in production for different algorithms or processes to ensure a more streamlined approach for enabling continuous monitoring and maintenance of algorithms in production. Figure 4.7 show one example of how a pipeline can be designed for deployed algorithms. Another example would be another pipeline to train and update the diagnostics models with newly obtained datasets or a pipeline to monitor the generalisation performance of the algorithms.
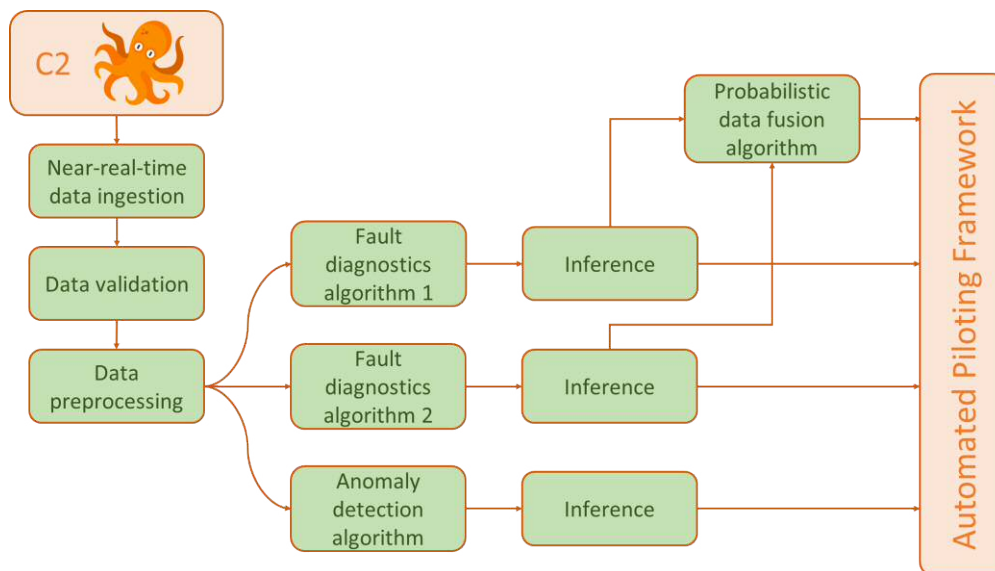

Figure 4.7: Example pipeline for an anomaly detection and fault diagnostics system.

The ML pipeline design enables a more efficient and structured workflow for designing, debug-and-tracing, validating, and verifying ML models for researchers and developers. Furthermore, as we validate this process further during field operations and as the pipeline matures, this infrastructure will help accelerate the progress of designing and deploying more complex ML algorithms.

C2 Automated Piloting Framework Architecture

NOC C2 system has an Automated Piloting Framework (APF) [38] which manages the high-level settings of autonomy and piloting algorithms. The APF is a server-side autonomous piloting framework for long-range marine autonomous systems (MAS) which automates piloting routines and SOPs for a diverse range of MAS, with user-adjustable autonomy. The APF also maintains a level of human oversight and accountability over MAS operations.

Whilst this is not a requirement for implementing and deploying ML models to production, the anomaly detection and fault diagnostics system has shown to be highly essential for autonomous operations, especially on heterogenous and multi-vehicle deployments. Therefore, we had designed and integrated the ML pipeline to work with the APF on NOC C2 as shown in Figure 4.8.
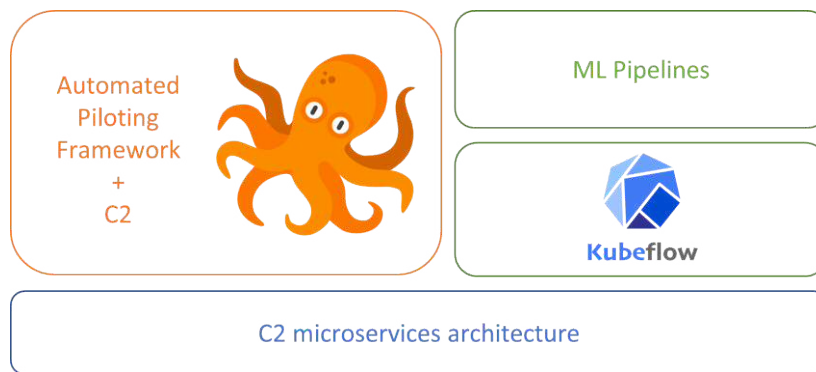
Figure 4.8: ML Pipeline with APF on C2 system architecture.

We have opted for the Kubeflow open-source ML platform to host the pipelines and orchestrate the ML workflows that we have in production. Kubeflow fits nicely into the microservices architecture of the NOC's C2 system, while also providing ample features that are useful for developing and deploying ML algorithms. Figure 4.9 shows the user interface of Kubeflow with an example pipeline.
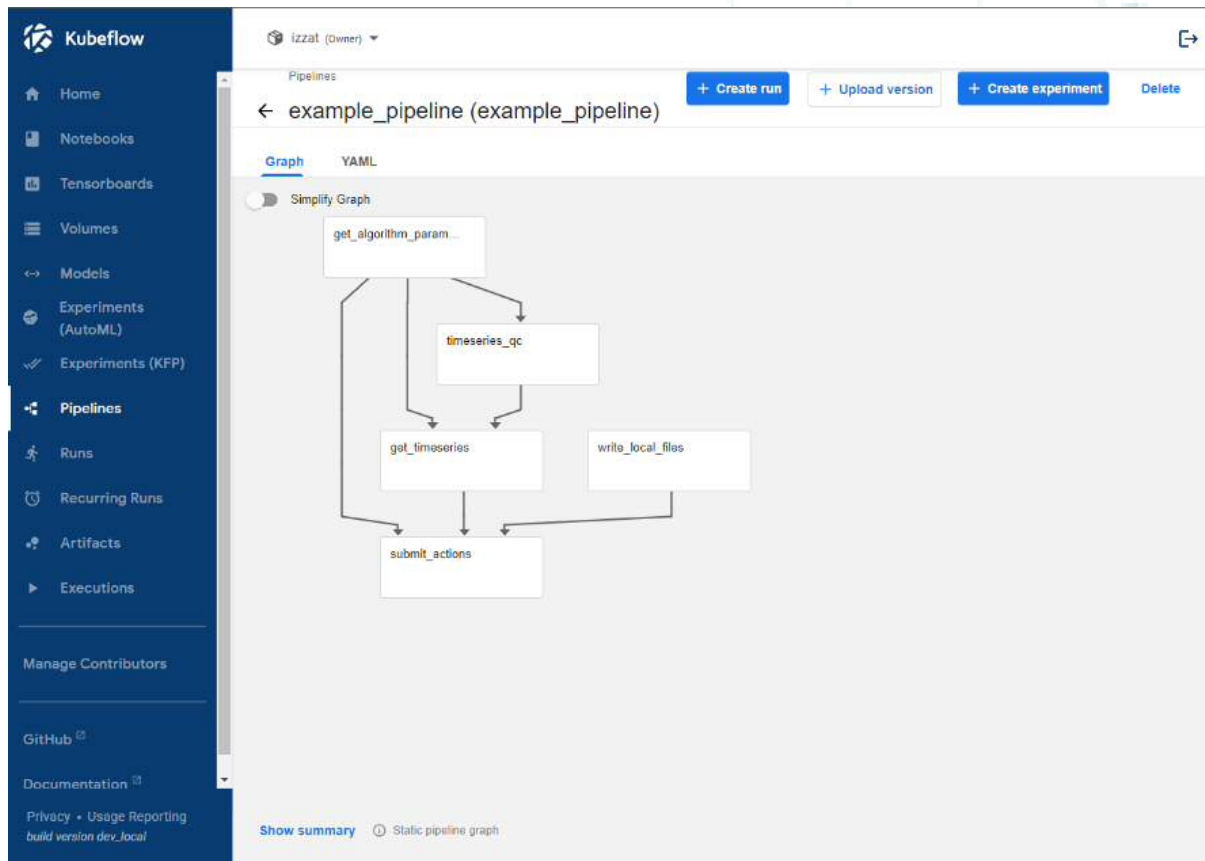
Figure 4.9: Kubeflow user interface with example pipeline.

Notifications are also set up through the C2 notification service to notify pilots of events and alerts via email or text messaging, to utilise the continuous monitoring feature of the APF and ML pipeline. This notification feature allows pilots to run more complicated MAS operations and focus on meeting the science goals of the deployment without needing to monitor the C2 constantly. Additionally, we also collect feedbacks and piloting logs from pilots through the log entry feature as shown in Figure 4.10. Pilot's will be able to add notes or labels to noteworthy events that occurs throughout a deployment. These logs will be essential when generating new annotated glider operation datasets to train and improve algorithms for the APF in the future.
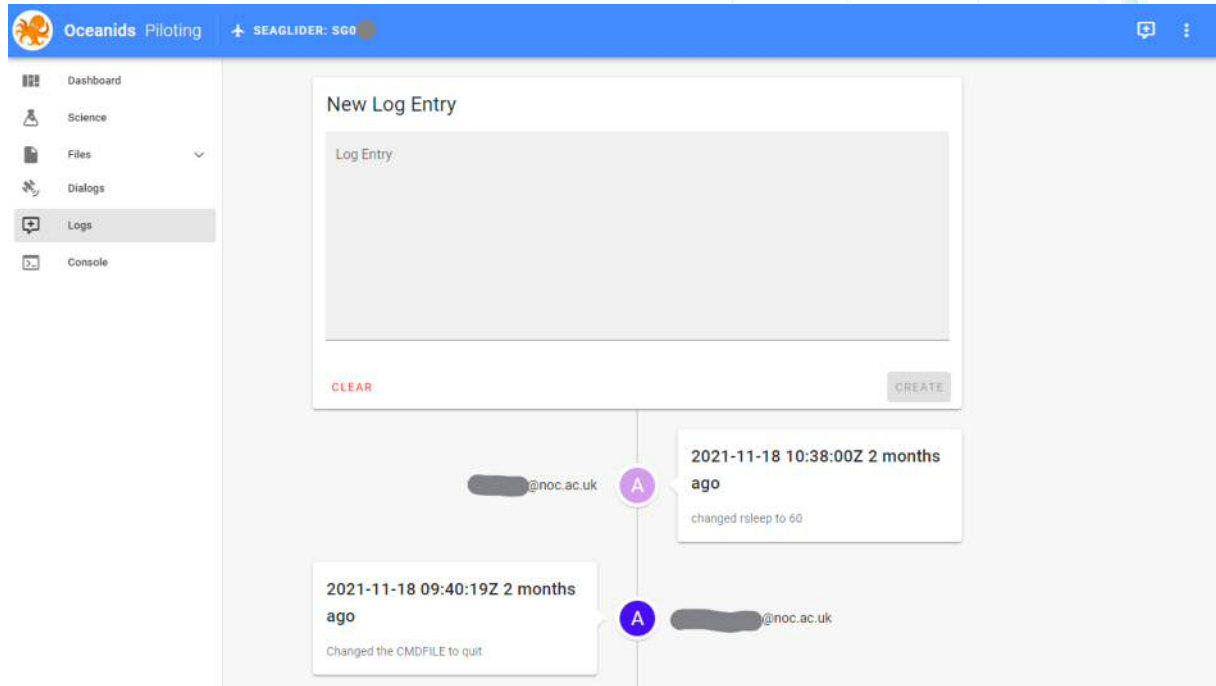
Figure 4.10: Piloting log entry feature user interface on C2.

5.   Project management

5.1.   Summary

The project was managed by the PI at UCL with UCL's mentor Co-I providing guidance when required.

To ensure the project achieves the desired impact on the regulations and operational practices of MAS, a steering committee was created, which included representatives from LR, the leading classification society for MAS, MCA, the key regulator for MCA in the UK, BOM, an exploration company and Dr Alexander Phillips, Head of Marine Autonomous Systems Development at NOC. In addition, the LRF Assuring Autonomy team at the University of York were part of the project stakeholders. The stakeholders met three times during the project duration to:

- Initially discuss the gathered MAS requirements and discuss the proposed solution for the autonomous detection system (M1),
- Update regulators on the development of the autonomous diagnosis system and determine how to approach WP4 (M2) (see Section 5.2),
- Inform stakeholders' decisions on the regulation of MAS (M3).

The completion of Body of knowledge is summarised in Table 5.1.

Table 5.1: Summary of Body of Knowledge.

Objective number	Description of output	Guidance document
2.2.1.1	Defining sensing requirements	Submitted in 03/2021, also attached to the final project report.
2.2.1.2	Defining understanding requirements	Submitted in 11/2021, also attached to the final project report.
2.2.4.1	Verification of sensing requirements	Draft submitted in 11/2021, updated version attached to the final project report.
2.2.4.2	Verification of understanding requirements	Draft submitted in 11/2021, updated version attached to the final project report.
3.1.1	Identifying sensing deviations	Submitted in March 2021, also attached to the final project report.

5.2.   Project ALADDIN: Workshop – 8th June 2021

We were delighted to welcome 31 collaborators from all over Europe to the first workshop for project "Assuring Long-term Autonomy through Detection and Diagnosis of Irregularities in Normal operation (ALADDIN)" on 8th June 2021. The workshop covered challenges and solutions with fault-tolerant operations of marine autonomous systems. We took advantage of the technology and moved from plenary sessions for the introduction to the project and panel discussion into breakout groups for detailed discussion, giving each of us a chance to meet and talk with people from a variety of disciplines, backgrounds, and domains.

This report provides a short summary of the interesting and informative discussions held during the workshop.

If you are interested in learning more about the work and outcomes of ALADDIN or in collaborating with us during or beyond the project, please feel free to contact us at E.Anderlini@ucl.ac.uk and catherine.harris@noc.ac.uk.

Panel Session

The panel consisted of the following four experts:

Joseph Morelos, Technology Innovation Manager, Lloyd's Register Marine and Offshore,

Dr Olga Fink, Assistant Professor in Intelligent Maintenance Systems, ETH Zürich,

Prof. Ralf Bachmayer, Professor for Marine Environmental Technology and Deep-sea Engineering, MARUM, University of Bremen,

Roland Rogers, former Advisor on Marine Law and Policy, National Oceanography Centre.

13 questions were posted on Slido by the audience, receiving 31 likes and engaging 30 participants. The following three questions were discussed by the panel.

1. **What main steps are needed to transition from human-in-the-loop to human-on-the-loop operations of marine autonomous systems?** The consensus was that there is no single solution, as the transition is really dependent on the operational environment, with strong differences between congested waterways and the open ocean. From a regulatory perspective, the differences in requirements between human-in- and human-on-the-loop need to be defined explicitly.

2. **What are your thoughts on scaling to larger and more complicated systems with regards to the amount of data required for a data driven methodology?** A pure data-driven solution is difficult, as vessel pilots and/or crew are unlikely to record failures consistently and verification is challenging. A hybrid solution can enhance performance, combining dynamic models with data-driven methods to improve fault diagnostics performance. Hierarchical methods with case-based algorithms are an interesting research direction. Scaling to complex vehicles is challenging, as behaviours need to be reliable.

3. **What about the transparency of autonomous systems and explainable AI?** Significant work is being done on explainability in the area of machine learning and computer vision, as this is a requirement for certification. The hybrid approach can improve the interpretability of the algorithms through physics induced learning, helping increase trust in the results and decisions made by the model. Structured metadata is needed for the validation of the models and to inform operators. A measure of transparency can be the ability to reconstruct errors. Standardised methods for demonstration are needed, but it is even more important to define what best practice is so as to aid accident report investigations in the future. The best advice is to document the demonstration work done to date to create legal precedent.

Breakout Room 1 (12 participants)

12 questions. Identified priority challenges and solutions:

1. **Deep understanding of technology and its transparency, explainability both for experts, users, regulators and general public.** Classification of marine autonomous systems based on roles, applications and size.

2. **Importance of data sharing.** Creation of standardised data and metadata formats based on existing ontologies and vocabularies for conventional ships, considering the interoperability and integration of different system types.

3. **Standardisation across the industry to enable interoperability.** Standardisation of the machine interface needed to help pilots' training with and operating heterogeneous systems. Proving the system works is key.

4. **Digital twins: how to prove that a system works well.** Uncertainty needs to be quantified and regulators are uncomfortable making decisions purely on synthetic data due to the compounding of errors. For remotely operated vehicle development, hardware twins based on hardware-in-the-loop solutions are an interesting solution to improve verifiability.

5. **Closing the loop and deciding and acting onboard, transferable autonomy, computational and other resources constrain (on-board vs remotely control).** Clear rules and boundaries are fundamental for regulators. The operator can be asked to make the decision on critical tasks. Models are limited so that prioritising what needs to be predicted on board is important. At present, reliability and trust are the real challenge.

Breakout Room 2 (12 participants)

15 questions on Slido with 23 votes. Identified priority challenges and **solutions**:

1. **How do you find the balance between using direct sensors vs sophisticated algorithms for failure detection?** The balance depends on the complexity of the system and a scalable, transparent solution is possible with hybrid methods. There are no specific regulations on sensor selection at present, so more work is needed. Ideally, a metric needs to be specified to guide the selection. The design should be based on functionality and be an iterative process: design, test, adapt.
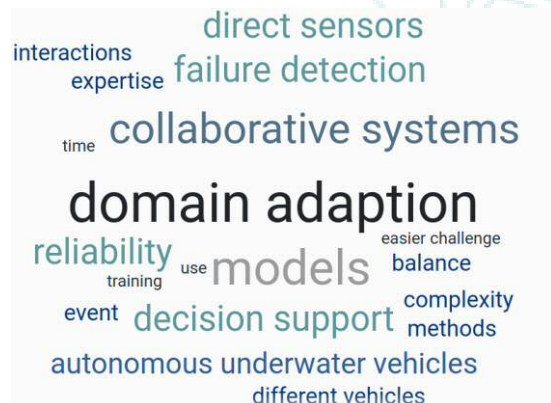
2. **Too many failure modes.** Focus on the critical failure modes. Understanding the root cause is not key to operations, but it is important for maintenance.

3. **Transfer learning from one system to another.** Domain adaptation for transferring models between different vehicles and/or operating conditions. However, there are no guarantees. Abstracting the system's behaviour and recollecting some data for the new system can help.

4. **Limited underwater communication.** Event driven and with data compression and standardisation. Open frameworks a solution, but cybersecurity is important.

6.   Acknowledgement

The work in the project has been possible only thanks to the help and guidance of the project steering partners:

- Joseph Morelos and Duncan Duffy at Lloyd's Register,
- Robert Gale, Saskia Daggett and Katrina Kemp at the Marine and Coastguard Agency,
- Robert Ayrton and James King at Blue Ocean Monitoring Ltd,
- Alexander B. Phillips at the National Oceanography Centre.

Additionally, we would also like to acknowledge the help that we have received from:

- James Bacon at the Scottish Association for Marine Science for the sourcing of information on quality control systems for underwater gliders,
- Justin Buck at the National Oceanography Centre for the sourcing of the quality control terminology and the datasets of MAS deployment and remote sensing,
- Steven A. Woodward at the National Oceanography Centre for the identification of deployments of underwater gliders with anomalous behaviour,
- Albert Miralles at SOCIB for the implementation of field tests in Mallorca of underwater gliders with anomalous behaviour to verify the developed anomaly detection and fault diagnostics tools,
- Maria Costa at the University of Porto for the implementation of field tests of AUV with anomalous behaviour to verify the developed anomaly detection and fault diagnostics tools.

7. References

[1] MSC, "MSC 102-5-18 - Proposed terminology for MASS (ISO)," 2020.

[2] R. Veal, M. Tsimplis, and A. Serdyc, "The legal status and operation of unmanned maritime vehicles," *Ocean Development and International Law*, vol. 50, no. 1, pp. 23–48, Jan. 2019, doi: 10.1080/00908320.2018.1502500.

[3] DNV GL, "Rules for Classification: Underwater technology, Part 5: Types of UWT systems, Chapter 8: Autonomous underwater vehicles," 2015.

[4] Lloyd's Register, "ShipRight: Design and Construction, Additional Design Procedures, Design Code for Unmanned Marine Systems," 2017.

[5] R. Isermann and P. Ballé, "Trends in the Application of Model-Based Fault Detection and Diagnosis of Technical Processes," *Control Engineering Practice*, vol. 5, no. 5, pp. 709–719, 1997.

[6] ISO, "ISO 13372:2012(en) — Condition monitoring and diagnostics of machines — Vocabulary," 2012.

[7] ISO, "ISO 26262-1:2018(en) Road vehicles — Functional safety — Part 1: Vocabulary," 2018.

[8] Lloyd's Register, "ShipRight: Design and Construction - Digital Compliance: Procedure for the Approval of Digital Health Management Systems," 2018.

[9] M. Brito, D. Smeed, and G. Griffiths, "Underwater glider reliability and implications for survey design," *Journal of Atmospheric and Oceanic Technology*, vol. 31, no. 12, pp. 2858–2870, 2014, doi: 10.1175/JTECH-D-13-00138.1.

[10] O. Fink, Q. Wang, M. Svensén, P. Dersin, W.-J. Lee, and M. Ducoffe, "Potential, challenges and future directions for deep learning in prognostics and health management applications," *Engineering Applications of Artificial Intelligence*, vol. 92, p. 103678, 2020, doi: 10.1016/j.engappai.2020.103678.

[11] K. Hamilton, D. M. Lane, K. E. Brown, J. Evans, and N. K. Taylor, "An integrated diagnostic architecture for autonomous underwater vehicles," *Journal of Field Robotics*, vol. 24, no. 6, pp. 497–526, Jun. 2007, doi: 10.1002/rob.20202.

[12] E. Anderlini, C. A. Harris, G. Salavasidis, A. Lorenzo, A. B. Phillips, and G. Thomas, "Autonomous Detection of the Loss of a Wing for Underwater Gliders," in *IEEE/OES Autonomous Underwater Vehicle Symposium (AUV)*, 2020, pp. 1–6. doi: 10.1109/AUV50043.2020.9267895.

[13] A. Zolghadri *et al.*, "Signal and model-based fault detection for aircraft systems," in *IFAC-PapersOnLine*, Sep. 2015, vol. 28, no. 21, pp. 1096–1101. doi: 10.1016/j.ifacol.2015.09.673.

[14] E. Anderlini *et al.*, "Identification of the Dynamics of Biofouled Underwater Gliders," in *IEEE/OES Autonomous Underwater Vehicle Symposium (AUV)*, 2020, pp. 1–6. doi: 10.1109/AUV50043.2020.9267919.

[15]   M. M. Islam, "Ship Smart System Design (S3D) and Digital Twin," in *VFD Challenges for Shipboard Electrical Power System Design*, Wiley-IEEE Press, 2019. doi: https://doi.org/10.1002/9781119463474.ch9.

[16]   A. L. Ellefsen, V. Asoy, S. Ushakov, and H. Zhang, "A comprehensive survey of prognostics and health management based on deep learning for autonomous ships," *IEEE Transactions on Reliability*, vol. 68, no. 2. Institute of Electrical and Electronics Engineers Inc., pp. 720–740, Jun. 2019. doi: 10.1109/TR.2019.2907402.

[17]   W. Zhang, D. Yang, and H. Wang, "Data-Driven Methods for Predictive Maintenance of Industrial Equipment: A Survey," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2213–2227, Sep. 2019, doi: 10.1109/JSYST.2019.2905565.

[18]   Y. Ran, X. Zhou, P. Lin, Y. Wen, and R. Deng, "A Survey of Predictive Maintenance: Systems, Purposes and Approaches," Dec. 2019.

[19]   B. Rezaeianjouybari and Y. Shang, "Deep learning for prognostics and health management: State of the art, challenges, and opportunities," *Measurement: Journal of the International Measurement Confederation*, vol. 163, Oct. 2020, doi: 10.1016/j.measurement.2020.107929.

[20]   M. A. Chao, C. Kulkarni, K. Goebel, and O. Fink, "Hybrid deep fault detection and isolation: Combining deep neural networks and system performance models," Aug. 2019.

[21]   M. A. Chao, B. T. Adey, and O. Fink, "Knowledge-Induced Learning with Adaptive Sampling Variational Autoencoders for Open Set Fault Diagnostics," Dec. 2019.

[22]   ISO, "ISO 13374-3:2012(en) Condition monitoring and diagnostics of machines — Data processing, communication and presentation — Part 3: Communication," 2012.

[23]   iRobot, "1KA Seaglider User's Guide," 2012.

[24]   T. W. Research, "Slocum G2 Glider Operators Training Guide," North Falmouth, MA, 2014.

[25]   D. C. Webb, P. J. Simonetti, and C. P. Jones, "SLOCUM: An underwater glider propelled by environmental energy," *IEEE Journal of Oceanic Engineering*, 2001, doi: 10.1109/48.972077.

[26]   O. Schofield *et al.*, "Slocum Gliders: Robust and ready," *Journal of Field Robotics*, vol. 24, no. 6, pp. 474–485, 2007, doi: 10.1002/rob.20200.

[27]   C. C. Eriksen *et al.*, "Seaglider: A long-range autonomous underwater vehicle for oceanographic research," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 424–436, 2001, doi: 10.1109/48.972073.

[28]   E. Anderlini, C. Harris, A. B. Phillips, A. Lorenzo Lopez, M. Woo, and G. Thomas, "Towards autonomy: A recommender system for the determination of trim and flight parameters for Seagliders," *Ocean Engineering*, vol. 189, no. August, p. 106338, 2019, doi: 10.1016/j.oceaneng.2019.106338.

[29]   M. E. Furlong, D. Paxton, P. Stevenson, M. Pebody, S. D. McPhail, and J. Perrett, "Autosub Long Range: A long range deep diving AUV for ocean monitoring," *2012*

IEEE/OES *Autonomous Underwater Vehicles, AUV 2012*, 2012, doi: 10.1109/AUV.2012.6380737.

[30]   D. Li, D. Chen, J. Goh, and S. Ng, "Anomaly Detection with Generative Adversarial Networks for Multivariate Time Series," pp. 1–10, 2018.

[31]   F. Di Mattia, P. Galeone, M. De Simoni, and E. Ghelfi, "A Survey on GANs for Anomaly Detection," 2019.

[32]   P. Wu *et al.*, "Unsupervised anomaly detection for underwater gliders using generative adversarial networks," *Engineering Applications of Artificial Intelligence*, vol. 104, p. 104379, Sep. 2021, doi: 10.1016/J.ENGAPPAI.2021.104379.

[33]   Z. Bedja-Johnson, P. Wu, D. Grande, and E. Anderlini, "Smart Anomaly Detection for Slocum Underwater Gliders with a Variational Autoencoder with Long Short-Term Memory Networks," *Applied Ocean Research*, 2022.

[34]   Department for Transport, "Technology and Innovation in UK Maritime: The case of Autonomy," London, UK, 2019.

[35]   C. A. Thieme and I. B. Utne, "Safety performance monitoring of autonomous marine systems," *Reliability Engineering and System Safety*, vol. 159, no. October 2016, pp. 264–275, 2017, doi: 10.1016/j.ress.2016.11.024.

[36]   Department for Transport, "Maritime 2050: Navigating the Future," London, UK, 2019.

[37]   J. Farley, A. W. Morris, O. D. Jones, C. A. Harris, and A. Lorenzo, "Marine Science from an Armchair: A Unified Piloting Framework for Autonomous Marine Vehicles," in *IEEE Oceans*, 2019, pp. 1–10.

[38]   C. A. Harris *et al.*, "Oceanids C2: An integrated command,control and data infrastructure for the over-the-horizon operation of marine autonomous systems," *Frontiers in Marine Science*, no. submitted in 2019, 2020.

[39]   P. S. Dias *et al.*, "Neptus-a framework to support multiple vehicle operation," in *Europe Oceans 2005*, 2005, vol. 2, pp. 963–968. doi: 10.1109/OCEANSE.2005.1513187.

[40]   L. Madureira *et al.*, "The light autonomous underwater vehicle: Evolutions and networking," in *2013 MTS/IEEE OCEANS-Bergen*, 2013, pp. 1–6. doi: 10.1109/OCEANS-Bergen.2013.6608189.

[41]   J. Pinto, P. S. Dias, R. Martins, J. Fortuna, E. Marques, and J. Sousa, "The LSTS toolchain for networked vehicle systems," in *2013 MTS/IEEE OCEANS-Bergen*, 2013, pp. 1–9. doi: 10.1109/OCEANS-Bergen.2013.6608148.

[42]   O. Fink, "Data-driven intelligent predictive maintenance of industrial assets," in *Women in Industrial and Systems Engineering*, Springer, 2020, pp. 589–605. doi: 10.1007/978-3-030-11866-2_25.

[43]   G. Michau and O. Fink, "Unsupervised transfer learning for anomaly detection: Application to complementary operating condition transfer," *Knowledge-Based Systems*, vol. 216, p. 106816, 2021, doi: 10.1016/j.knosys.2021.106816.

[44] X. Hong, R. J. Mitchell, S. Chen, C. J. Harris, K. Li, and G. W. Irwin, "Model selection approaches for non-linear system identification: A review," *International Journal of Systems Science*, vol. 39, no. 10, pp. 925–946, Oct. 2008, doi: 10.1080/00207720802083018.

[45] D. Crestani, K. Godary-Dejean, and L. Lapierre, "Enhancing fault tolerance of autonomous mobile robots," *Robotics and Autonomous Systems*, vol. 68, pp. 140–155, Jun. 2015, doi: 10.1016/j.robot.2014.12.015.

[46] A. Freddi, S. Longhi, and A. Monteriù, "Actuator fault detection system for a remotely operated vehicle," in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2013, vol. 46, no. 33 PART 1, pp. 356–361. doi: 10.3182/20130918-4-JP-3022.00050.

[47] Y. Wang and M. Zhang, "Research on Test-platform and Condition Monitoring Method for AUV," 2006. doi: 10.1109/ICMA.2006.257448.

[48] Y. S. Sun, X. R. Ran, Y. M. Li, G. C. Zhang, and Y. H. Zhang, "Thruster fault diagnosis method based on Gaussian particle filter for autonomous underwater vehicles," *International Journal of Naval Architecture and Ocean Engineering*, vol. 8, no. 3, pp. 243–251, May 2016, doi: 10.1016/j.ijnaoe.2016.03.003.

[49] F. Yao, F. Wang, and M. Zhang, "Weak thruster fault detection for autonomous underwater vehicle based on artificial immune and signal pre-processing," *Advances in Mechanical Engineering*, vol. 10, no. 2, Feb. 2018, doi: 10.1177/1687814018758739.

[50] W. Ray Harris, "Anomaly Detection Methods for Unmanned Underwater Vehicle Signature redacted Signature redacted Signature redacted," Massachusetts Institute of Technology, 2015.

[51] B.-Y. Raanan *et al.*, "Automatic Fault Diagnosis for Autonomous Underwater Vehicles using Online Topic Models Ben-Yair," 2016. doi: 10.1109/OCEANS.2016.7761139.

[52] B. Y. Raanan *et al.*, "Detection of unanticipated faults for autonomous underwater vehicles using online topic models," *Journal of Field Robotics*, vol. 35, no. 5, pp. 705–716, Aug. 2018, doi: 10.1002/rob.21771.

[53] C. A. Thieme and I. B. Utne, "Safety performance monitoring of autonomous marine systems," *Reliability Engineering and System Safety*, vol. 159, pp. 264–275, Mar. 2017, doi: 10.1016/j.ress.2016.11.024.

[54] E. Anderlini, D. A. Real-arce, T. Morales, C. Barrera, and G. Thomas, "An Innovative Marine Growth Detection System for Underwater Gliders," *Journal of Oceanic Engineering*, no. June, pp. 1–15, 2021, doi: 10.1109/JOE.2021.3066373.

[55] G. Pang, C. Shen, L. Cao, and A. Van Den Hengel, "Deep learning for anomaly detection: A review," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1–38, 2021, doi: doi.org/10.1145/3439950.

[56] M. Sakurada and T. Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction," in *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*, 2014, pp. 4–11.

[57] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, 2017, pp. 665–674.

[58] A. Borghesi, A. Bartolini, M. Lombardi, M. Milano, and L. Benini, "A semisupervised autoencoder-based approach for anomaly detection in high performance computing systems," *Engineering Applications of Artificial Intelligence*, vol. 85, pp. 634–644, 2019, doi: 10.1016/j.engappai.2019.07.008.

[59] E. Anderlini *et al.*, "A Remote Anomaly Detection System for Slocum Underwater Gliders," *Ocean Engineering*, 2021.

[60] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, and D. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery \& Data Mining*, 2019, pp. 2828–2837. doi: 10.1145/3292500.3330672.

[61] L. Li, J. Yan, H. Wang, and Y. Jin, "Anomaly detection of time series with smoothness-inducing sequential variational auto-encoder," *IEEE transactions on neural networks and learning systems*, 2020, doi: 10.1109/TNNLS.2020.2980749.

[62] I. Goodfellow *et al.*, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[63] F. Di Mattia, P. Galeone, M. De Simoni, and E. Ghelfi, "A survey on gans for anomaly detection," *arXiv preprint arXiv:1906.11632*, 2019.

[64] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," *arXiv preprint arXiv:1605.09782*, 2016.

[65] V. Dumoulin *et al.*, "Adversarially learned inference," *arXiv preprint arXiv:1606.00704*, 2016.

[66] H. Zenati, M. Romain, C.-S. Foo, B. Lecouat, and V. Chandrasekhar, "Adversarially learned anomaly detection," in *2018 IEEE International Conference on Data Mining (ICDM)*, 2018, pp. 727–736. doi: 10.1109/ICDM.2018.00088.

[67] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *International conference on information processing in medical imaging*, 2017, pp. 146–157. doi: 10.1007/978-3-319-59050-9_12.

[68] T. Schlegl, P. Seeböck, S. M. Waldstein, G. Langs, and U. Schmidt-Erfurth, "f-anogan: Fast unsupervised anomaly detection with generative adversarial networks," *Medical image analysis*, vol. 54, pp. 30–44, 2019, doi: 10.1016/j.media.2019.01.010.

[69] D. Li, D. Chen, J. Goh, and S. Ng, "Anomaly detection with generative adversarial networks for multivariate time series," *arXiv preprint arXiv:1809.04758*, 2018.

[70] U. Mutlu and E. Alpaydin, "Training bidirectional generative adversarial networks with hints," *Pattern Recognition*, p. 107320, 2020, doi: 10.1016/j.patcog.2020.107320.

[71] Teledyne Webb Research, "Slocum G2 Glider Operators Manual," 2012.

[72]   BODC, "Glider inventory." 2019. Accessed: Apr. 09, 2021. [Online]. Available: https://www.bodc.ac.uk/data/bodc%5C_database/gliders/

[73]   A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[74]   C. D. I. I. Haldeman, D. K. Aragon, T. Miles, Scott M. Glenn, and A. G. Ramos, "Lessening biofouling on long-duration AUV flights: Behavior modifications and lessons learned," 2016. doi: 10.1109/OCEANS.2016.7761236.

[75]   P. Wu *et al.*, "Anomaly Detection and Fault Diagnostics for Underwater Gliders Using Deep Learning," 2021.

[76]   D. L. Rudnick, "Ocean Research Enabled by Underwater Gliders," *Annual Review of Marine Science*, vol. 8, no. 1, pp. 519–541, 2016, doi: 10.1146/annurev-marine-122414-033913.

[77]   S. Wood, "Autonomous Underwater Gliders," in *Underwater Vehicles*, A. Inzartsev, Ed. IntechOpen, 2009, pp. 499–524.

[78]   E. Anderlini *et al.*, "A Marine Growth Detection System for Underwater Gliders," *IEEE Journal of Oceanic Engineering*, 2021.

[79]   D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," *arXiv preprint arXiv:1905.02249*, 2019.