DOCTORAL THESIS

Reliable Object Recognition and Assessment in Adverse Weather and Environmental Conditions

Jürgen Soom

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY
TALLINN 2026

TALLINN UNIVERSITY OF TECHNOLOGY
DOCTORAL THESIS
10/2026

Reliable Object Recognition and Assessment in Adverse Weather and Environmental Conditions

JÜRGEN SOOM

TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies
Department of Computer Systems

The dissertation was accepted for the defence of the degree of Doctor of Philosophy in Computer and Systems Engineering, on 9th January 2026

Supervisor: Assoc. Prof. Jeffrey Andrew Tuhtan, PhD
Department of Computer Systems, School of Information Technologies,
Tallinn University of Technology,
Tallinn, Estonia

Co-supervisor: Mairo Leier, PhD
Department of Computer Systems, School of Information Technologies,
Tallinn University of Technology,
Tallinn, Estonia

Opponents: Katharina Bensing, PhD
Department of Civil and Environmental Engineering,
Technical University of Darmstadt,
Darmstadt, Germany

Assoc. Prof. Matteo Fumagalli, PhD
Department of Electrical and Photonics Engineering,
Technical University of Denmark,
Kongens Lyngby, Denmark

Defence of the thesis: 2nd February 2026

Declaration:
Hereby I declare that this doctoral thesis, my original investigation and achievement, submitted for the doctoral degree at Tallinn University of Technology, has not been submitted for any academic degree elsewhere.

Jürgen Soom ______________________
signature

Copyright: Jürgen Soom, 2026
ISSN 2585-6898 (publication)
ISBN 978-9916-80-457-5 (publication)
ISSN 2585-6901 (PDF)
ISBN 978-9916-80-458-2 (PDF)
DOI https://doi.org/10.23658/taltech.10/2026
Printed by Koopia Niini & Rauam

Soom, J. (2026). *Reliable Object Recognition and Assessment in Adverse Weather and Environmental Conditions* [TalTech Press]. https://doi.org/10.23658/taltech.10/2026

TALLINNA TEHNIKAÜLIKOOL
DOKTORITÖÖ
10/2026

Usaldusväärne objektide tuvastamine ja hindamine ebasoodsates ilmastiku- ja keskkonnatingimustes

JÜRGEN SOOM

TAL
TECH
KIRJASTUS

Contents

List of Publications 7

Author's Contributions to the Publications 8

Abbreviations 9

1 Introduction 10
- 1.1 Object Detection and Object Assessment 11
- 1.2 Object Detection, Assessment, and Evaluation Challenges 14
- 1.3 Hardware 14
- 1.4 Hardware Challenges 15
- 1.5 Research Gaps, Questions and Contributions 16
- 1.6 Thesis Organization 19

2 Environmental Classification Model 20
- 2.1 Background and Motivation 20
- 2.2 Environmental Conditions 20
- 2.3 Environmental Condition Classification 22
- 2.4 Results and Discussion 24

3 Traffic Monitoring 26
- 3.1 Background and Motivation 26
- 3.2 Dataset 26
- 3.3 Monitoring System Hardware 29
- 3.4 Vehicle Detection and Classification 30
- 3.5 Results and Discussion 30

4 Object Assessment 32
- 4.1 Background and Motivation 32
- 4.2 Dataset 33
- 4.3 System Hardware Overview 35
- 4.4 Detection and Size Estimation 36
- 4.5 Results and Discussion 37

5 Conclusion 39
- 5.1 Limitations 40
- 5.2 Future Work 41

List of Figures 43

List of Tables 44

References 45

Acknowledgements 53

Abstract 55

Kokkuvõte 56

Appendix 1 57

Appendix 2 73

Appendix 3 89

Curriculum Vitae 105

Elulookirjeldus 107

List of Publications

This Ph.D. thesis is based on the following three journal publications.

I J. Soom, V. Pattanaik, M. Leier, and J. A. Tuhtan. Environmentally adaptive fish or no-fish classification for river video fish counters using high-performance desktop and embedded hardware. *Ecological Informatics*, 72, 2022

II J. Soom, M. Leier, K. Janson, and J. A. Tuhtan. Open urban mmwave radar and camera vehicle classification dataset for traffic monitoring. *IEEE Access*, 12:65128–65140, 2024

III J. Soom, I. Boavida, R. Leite, M. J. Costa, G. Toming, M. Leier, and J. A. Tuhtan. Open real-time, non-invasive fish detection and size estimation utilizing binocular camera system in a portuguese river affected by hydropeaking. *Ecological Informatics*, 90, 2025

Author’s Contributions to the Publications

I In **Publication I**, the author served as the corresponding author. The key responsibilities and contributions included conducting the initial literature review, designing, developing, and validating the pipeline, annotating and curating the dataset, and writing the manuscript.

II In **Publication II**, the author’s contributions included defining the research problem, conducting the initial literature review, and overseeing the annotation and curation processes. The author also led the development, training, and evaluation of both models and was the primary contributor to the manuscript.

III In **Publication III**, the author conducted the literature review, assisted in creating the dataset, and developed and tested the detection and assessment pipeline. Additionally, the author was the primary contributor to the manuscript and coordinated the work between the universities.

Abbreviations

AP	Average Precision
CIM	Critical Infrastructure Monitoring
CNN	Convolutional Neural Network
COCO	Common Objects in Context
DL	Deep Learning
ECCM	Environmental Condition Classification Model
EU	European Union
GPU	Graphical Processing Unit
HPC	High-Performance Computing
ILD	Inductive Loop Detector
IoU	Intersection over Union
IR	Infrared
KNN	k-Nearest Neighbors Algorithm
mAP	mean Average Precision
ML	Machine Learning
mmWave	millimeter-wave
NDVI	Normalized Difference Vegetation Index
NPU	Neural Processing Unit
PHF	Peak Hour Factor
PPW	Performance Per Watt
PTZ	Pan-Tilt-Zoom
RG	Research Gap
RGB	Red, Green, Blue
RQ	Research Question
SBC	Single-Board Computer
SfM	Structure from Motion
TOPS	Tera Operations Per Second
TQ	Technological Gaps
TPU	Tensor Processing Unit
UAV	Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle
UN	United Nations
UW	Underwater
VGG	Visual Geometry Group
YOLO	You Only Look Once

1 Introduction

Outdoor monitoring systems play a central role in the acquisition of information in several domains, each defined by unique operational requirements (see Figure 1). Recent advances in sensing hardware, data processing capabilities, and computational technologies have significantly enhanced their adaptability, facilitating deployment in increasingly diverse and complex environments. Broadly generalized, these systems can be categorized based on their operational environment: underwater or terrestrial.

To study species in their natural habitats contributes to understanding ecological dynamics and assists in developing conservation strategies that address threats to biodiversity and the overall ecosystem [52]. Over the years, numerous non-invasive methods have been developed to advance ecological research on species that are difficult to observe or capture, such as cryptic species or those living in harsh environments. Camera traps have been utilized for more than a century and have proven to be versatile, serving various purposes, including monitoring the status of the wildlife population, searching for rare species, estimating biodiversity, studying habitat preferences and behavior, and detecting poachers [41, 16]. Monitoring freshwater ecosystems is equally vital, yet it presents own unique challenges. Despite comprising only 0.01% of the water on Earth, the freshwater ecosystems (rivers, lakes, and wetland) host one-third of all vertebrate species and are experiencing a rapid decline [20]. Destruction of the global wetlands is occurring at a pace three times faster than that of forests, and the compounding impacts of climatic and anthropogenic changes are reducing freshwater vertebrate populations at more than twice the rate of terrestrial or marine populations [84, 30]. Increasing uncertainty in important fisheries worldwide poses a threat to both economic and food security in many parts of the world. Underwater camera-based fish monitoring can be used to assess and understand fish ecology and to manage populations appropriately, which requires accurate data on species occurrence, abundance, body size, distribution, and behavior [62, 40]. Considering freshwater fish species, a broader and more accurate representation of their daily migration activities and counts is required to study, understand, predict, and support sustainable freshwater fisheries [54, 26].

In addition to natural environments, cameras have also been used successfully to monitor human activity, especially to remediate traffic problems in urban areas. With the growing number of vehicles worldwide, the development and management of the city's transportation infrastructure has become a substantial and persistent challenge. Frequent problems include traffic congestion, environmental pollution, and noise pollution. To address these challenges, traffic monitoring systems are deployed to collect data on traffic flow [2, 3, 27, 1, 4]. Methods and approaches for monitoring and collecting data have seen constant changes over the past several decades. State-of-the-art solutions rely on sensors, usually camera sensors with added additional modalities (radar or LiDAR), capable of covering multiple lanes simultaneously, while requiring minimal or no maintenance [76, 9].

A camera-based outdoor monitoring system, regardless of its target domain, serves the purpose of collecting, analyzing, and interpreting data to facilitate informed decision-making. The process can be divided into two primary procedures: object detection and object assessment. **Object detection** involves two main functions: identifying the type of object and determining the object's spatial coordinates within an image (see Fig. 2 (c)). **Object assessment** goes beyond localization and classification. It includes evaluating an object's condition, behavior, characteristics, or attributes (as shown in Fig. 2 (d)). In traffic monitoring, object assessment may involve measuring vehicle speed, assessing potential bottlenecks, or analyzing motorist behavioral patterns.

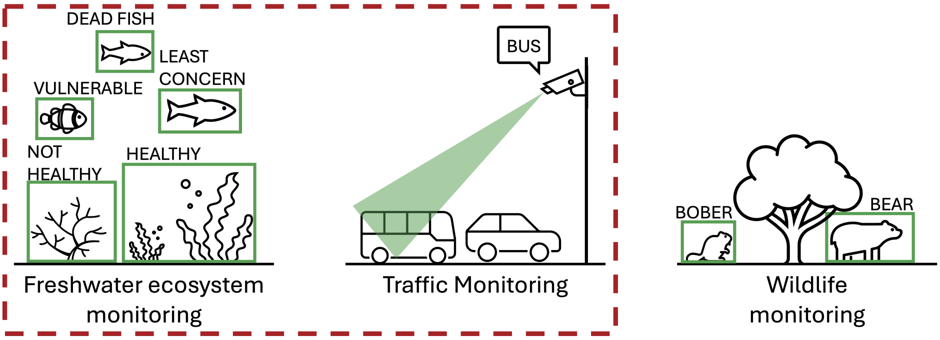


Figure 1: Outdoor monitoring applications: (1) Coral reef and fish monitoring: assessing reef health and detecting fish species using underwater imaging. (2) Traffic monitoring: identifying vehicles and buses for real-time traffic management. (3) Wildlife monitoring: identifying and tracking animals, such as bears and beavers, in natural habitats. The research covered in this dissertation focuses on infrastructure-based traffic monitoring and the evaluation of fish populations and migration.

In the context of fish monitoring, assessments may include evaluating individual fish, such as their species and size, assessing health indicators (including signs of stress or disease), and analyzing behavioral responses (like swimming speed or reactions to environmental stimuli). To visually represent this process, Fig. 2 outlines the typical stages of a monitoring system.

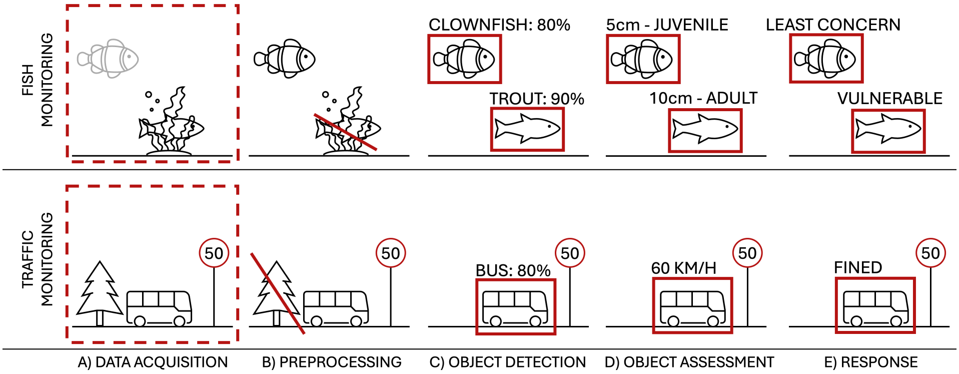


Figure 2: A visual representation of a monitoring pipeline applied to traffic and fish ecosystem monitoring. The workflows progress through data acquisition, preprocessing (removal of irrelevant elements and improvement of visibility), object detection (with classification and confidence scores), object assessment (e.g., speed or size estimation), and response (e.g., issuing fines or assessing conservation status).

1.1 Object Detection and Object Assessment

Visualized in Fig. 3, the timeline of object detection can be divided into two distinct eras: before the emergence of AlexNet in 2012 [51] (*Traditional Detection Methods*) and the era of *Deep Learning Methods*. Before deep learning, traditional object detection methods relied on hand-made features and traditional machine learning algorithms such as Viola-Jones (VJ) [87], Histogram of Oriented Gradients (HOG) [25], or Deformable Parts

Model (DPM) [31, 32, 33]. However, these older approaches often suffered from limited scalability and generalization to new object classes. Before transformer-based models, Convolutional Neural Network (CNN) models were divided into two groups: two-stage algorithms (candidate-based algorithms) and single-stage algorithms (regression-based algorithms).

Two-stage object detection algorithms operate by first extracting candidate regions from the image, known as region proposals. These proposals are then processed using a convolutional neural network to classify and localize the objects within them. While two-stage methods typically achieve higher accuracy, they tend to have slower detection speeds compared to one-stage approaches. Representative examples of two-stage object detectors include R-CNN [38], SPP-Net [43], Fast R-CNN [37], and Faster R-CNN [72].

Single-stage algorithms directly generate the positioning coordinates and classification probability of objects in the image without the need to generate a region proposal in advance. Because there is one less computationally intensive step, their detection speed is relatively quick compared to two-stage options, while showing similar or improved performance as summerized in Fig. 4. You Only Look Once (YOLO) [71] versions and Single Shot MultiBox Detector (SSD) [59] are two of the most well-known single-stage object detection algorithms.

The development and implementation of transformer-based object detection models have emerged as a result of advancements in natural language processing and the successful application of transformers in various other computer vision tasks. Researchers and engineers have recognized the potential in terms of detection performance (Fig. 4) of transformers to capture long-range dependencies in image data, leading to the exploration and adoption of transformer architectures in numerous computer vision applications [28, 8, 77, 57].

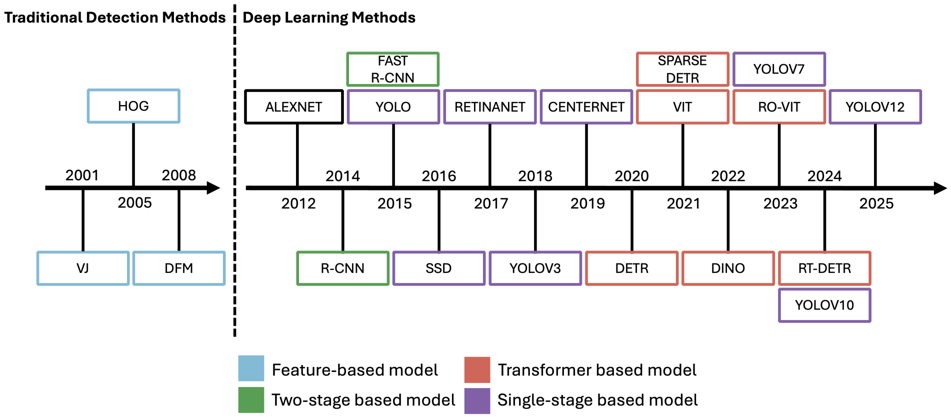


Figure 3: Timeline of advancements in object detection models, spanning from 2001 to 2025. The progression begins with traditional feature-based approaches, such as VJ (Viola-Jones) and HOG, advancing through the introduction of deep learning-based models such as AlexNet and R-CNN, and evolving toward modern architectures that are based on transformers, namely ViT (Vision Transformer) and RT-DETR.

The distinctive attributes and features of digital camera's unique characteristics highlight its versatility and capabilities across multiple domains and applications, capturing rich information, including color, texture, shape, and spatial context. For example, high-resolution imaging has already been integrated into stream measurement systems.

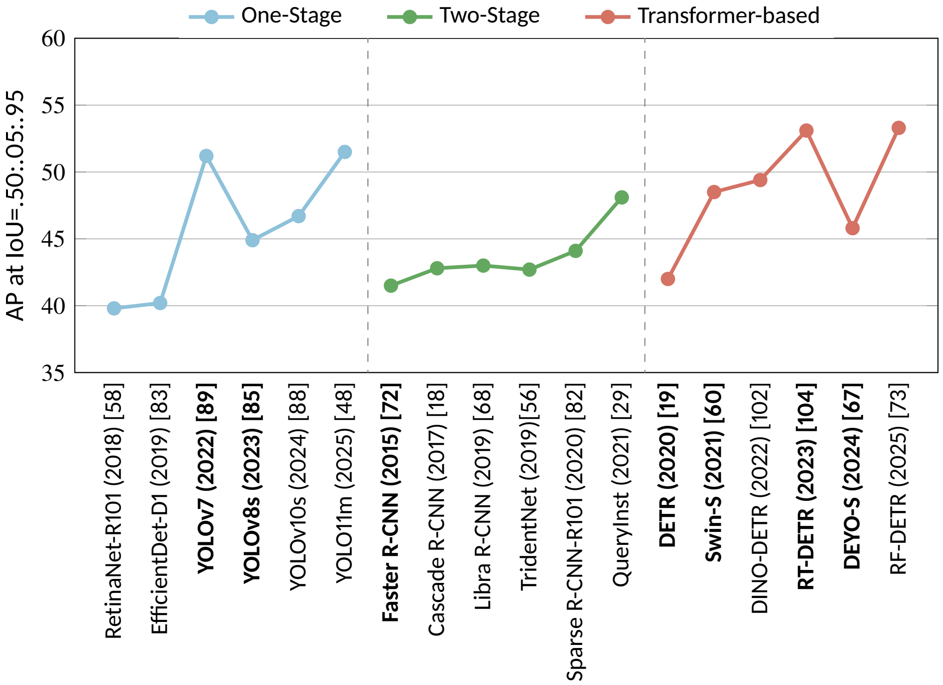


Figure 4: Object Detection Accuracy on COCO 2017 Validation Set by Architecture Type (2015–2025). Each point represents a model's mAP score, grouped by architecture class. ***Highlighted markers indicate the models tested and utilized during the research.***

These systems employ segmentation algorithms and geometric calculations of points of interest to accurately assess water levels, aiding in environmental monitoring and informed decision-making. Another use case is water quality monitoring, which involves detecting floating debris and contaminants in dynamic aquatic environments [92].

Compared to standard RGB color data capture, cameras equipped with multispectral and hyperspectral imaging capabilities can analyze information beyond the visible spectrum. Detecting infrared, ultraviolet, and other spectral bands, facilitating sophisticated environmental assessments - including evaluations of plant health, pollutant detection, and comprehensive water quality monitoring. Multispectral imaging has been utilized effectively in outdoor agricultural settings to assess the well-being of vegetation. By analyzing different spectral bands, these systems can compute vegetation indices such as the Normalized Difference Vegetation Index (NDVI), allowing early detection of crop stress and improving precision of agricultural practices [53]. In underwater environments, the deployment of hyperspectral imaging systems enables monitoring of coral health. These systems utilize spectral analysis to distinguish between healthy corals and bleached or stressed corals by capturing subtle color changes and specific spectral signatures that are imperceptible to the naked eye or standard RGB cameras [64].

In addition to capturing static objects, camera systems offer invaluable contextual information on surrounding environments. Advanced techniques, including semantic segmentation and object classification, facilitate scene interpretation, offering insights into spatial relationships and dynamic ecological changes. For example, a camera-based system designed to monitor wildlife habitats in outdoor settings utilized semantic segmentation models to distinguish between natural elements—such as trees, rocks, and water—and animal populations. This capability significantly advanced automated monitoring of species behavior and habitat use, thus supporting conservation efforts [52, 41, 16]. Similarly, an underwater camera system was developed to analyze coral reef ecosystems, utilizing deep learning to classify various types of coral and identify signs of bleaching. By providing a comprehensive view of reef health, this system helped identify stress factors that impact ecosystem health [62, 40, 39, 7].

1.2 Object Detection, Assessment, and Evaluation Challenges

Both object detection and assessment encounter substantial challenges in dynamic and uncontrolled environments. Fog reduces visibility by scattering light and diminishing contrast, making it difficult to discern objects due to the lack of detail. Precipitation events, such as rain and snow, further impair visibility, distorting the image. Low light conditions introduce noise and blur, complicating detection efforts. In contrast, excessive illumination—such as reflective glare or various light sources, can cause overexposure, distorting the object, thus negatively impacting detection or assessment performance [98, 97].

The underwater environment presents even more pronounced challenges. The accumulation of biofilm on the lens of the camera reduces the clarity, and bubbles moving through the water column introduce additional distortions. Turbidity, caused by suspended particles, increases light absorption and scattering, creating a diffuse effect that deteriorates overall image quality. Excess sunlight in shallow waters creates glare, obscuring finer details, and complicating image analysis. On the other hand, limited lighting conditions reduce the overall visibility of the scene [105].

External interference adds another layer of complexity, creating additional challenges. Poor camera positioning, lens obstructions (e.g. dirt and biofilm), and environmental vibrations can disrupt image stability and clarity. Refraction and geometric distortion underwater challenge depth estimation and feature extraction. Motion blur is particularly problematic in situations where stabilization may not be fully effective due to the system's movement or vibrations coming from the environment, such as traffic. Depth and size estimation challenges are prominent with monocular cameras, which struggle to provide accurate measurements without the use of stereo or multi-angle views. The absence of reference objects further complicates the estimation of the size using only visual data. Occlusion can also introduce difficulties in detecting or assessing objects in densely populated or vegetated settings. Dynamic and cluttered backgrounds, such as those found in urban or aquatic settings, complicate object detection and tracking, particularly when partially or fully obscured objects are present.

1.3 Hardware

Edge computing has transformed outdoor monitoring systems in the past decade by processing data closer to its source. This shift has reduced latency, thereby improving real-time decision-making capabilities, and reducing bandwidth consumption. It has been driven by advances in computational hardware and the growing need for solutions that effectively address challenges related to real-time processing, infrastructure limitations (such as power grids and communications), data security, and system reliability.

A significant factor in this transformation has been the emergence of neural network architectures, which enable the precise analysis of complex visual scenes. However, this significant performance improvement brings with it an increased demand for computational resources, especially in applications that involve segmentation, tracking, and size estimation. To improve computational efficiency, tensor cores have been integrated into traditional graphics processing units (GPUs) [65], allowing faster calculations involving matrices. Additionally, hardware accelerators such as Intel's Neural Compute Stick [46] and Google's Tensor Processing Unit (TPU) [49] have been utilized to a greater extent, thus expanding the applications of edge computing while minimizing power consumption.

Fig. 5 presents a comparative analysis of AI computing devices, measuring performance in TOPS (Tera Operations Per Second) against power consumption in watts for selected low-power embedded hardware released between 2017 and 2024. This period has witnessed a remarkable increase in AI performance, with newer devices showcasing significantly enhanced computational power relative to their predecessors.

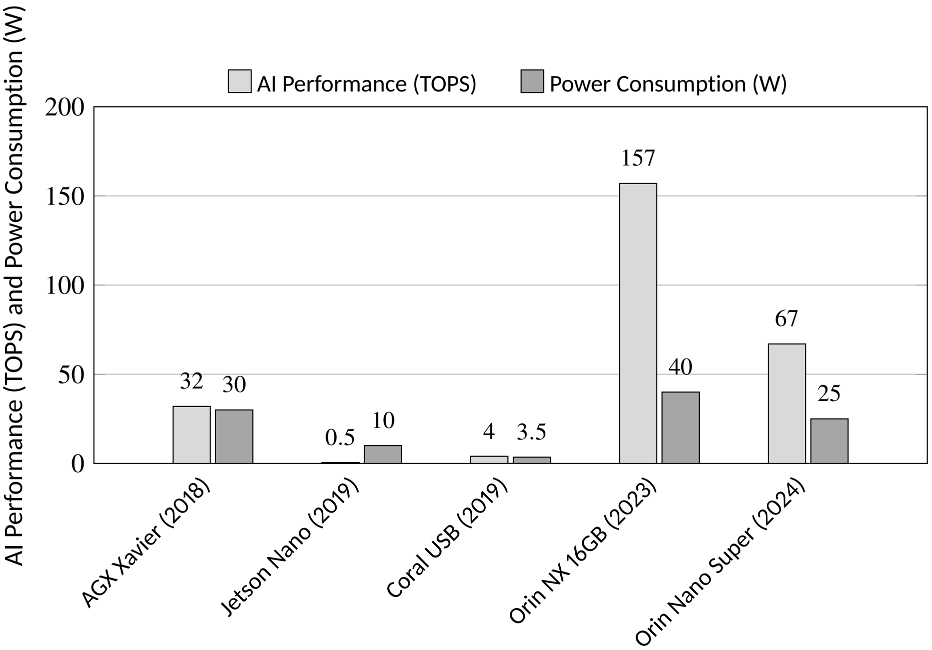


Figure 5: AI performance (TOPS) vs. power consumption (W) of edge computing devices released between 2018 and 2024. Early models, such as the Jetson Nano (2019) and Coral USB (2019), offer limited performance at moderate power consumption. Newer devices, such as the Orin NX 16GB (2023) and Orin Nano Super (2024), show significant improvements, with the latter reaching 67 TOPS at 25 W. The devices featured in the graph were used at various stages throughout the research discussed in the dissertation. Although numerous devices are available from various manufacturers, as described in the publications, the selection criteria were primarily influenced by factors such as capabilities, availability, cost, and documentation [80, 79].

1.4 Hardware Challenges

The deployment of models dedicated to object detection or assessment on constrained hardware presents numerous challenges, including, but not limited to, computational resource restrictions (size, memory limitations, and access to specialized hardware) and limited access to infrastructure. Each of these factors may impose specific constraints on processing capabilities, resulting in scenarios that are either inadequately slow, unable to fit within available memory, or reliant on power availability for processing. A crucial consideration is the overall feasibility from both practical and economic perspectives, which often renders specific solutions unsustainable despite their inherent advantages over current state-of-the-art options. Initiatives have been launched to increase the adoption rates of precision agriculture technologies; from 2017 to 2021, the U.S. Department of Agriculture (USDA) and the National Science Foundation (NSF) allocated approximately $200 million for research and development in this field [5]. However, despite these substantial investments, only 27% of agricultural entities have implemented precision agriculture technologies, mainly due to the high costs associated with their acquisition.

1.5 Research Gaps, Questions and Contributions

These challenges underscore limitations in current outdoor and underwater camera-based monitoring solutions, particularly in adverse and varying weather and other environmental conditions. Based on the challenges and needs identified in **Sections 1.2** and **1.4**, three research gaps (RG) were identified:

RG 1: In underwater environments and during adverse weather conditions on land, the ability to detect, assess, and evaluate objects is greatly hindered.

RG 2: Object detection and assessment require significant computational resources, which presents major challenges when these tasks are executed on hardware with limited resources in outdoor settings.

RG 3: Scarcity of outdoor datasets available to the public that are specifically created for the development and evaluation of object detection and assessment. In particular, there is a need for datasets that include challenging environmental or weather conditions.

The associated research questions driving the current dissertation focus on developing practical and applicable solutions to address these three research gaps. Grounded in applied research, this work primarily aligns with Edison's quadrant (see Fig. 6), emphasizing immediate utility over theoretical considerations. By integrating fundamental research with problem-solving objectives, this study seeks to contribute to the development of effective methodologies and solutions for real-world deployment. Addressing these research gaps, the following research questions are formulated. Each question is designed to explore a specific aspect of the identified challenges, with corresponding contributions detailed in relevant publications.

RQ 1: What are the potential benefits of including an environmental condition classification model for object detection and assessment in harsh and adverse environments?
Publication I
Contributions: The main contribution of **Publication I** was the development of an environmental classification model capable of distinguishing six distinct environmental conditions: clear, low lighting, air bubbles, biofilm growth, turbidity and overexposure, along with their respective severities. This model facilitated the selection of appropriate preprocessing techniques tailored to each environmental condition, thus enhancing the accuracy of fish detection. The trained model demonstrated a high level of accuracy while maintaining computational efficiency, making it highly suitable for real-time applications. This outcome highlights the model's considerable potential for broader applicability, suggesting its utility in other domains that face similar challenges in object detection under varying environmental or weather conditions.

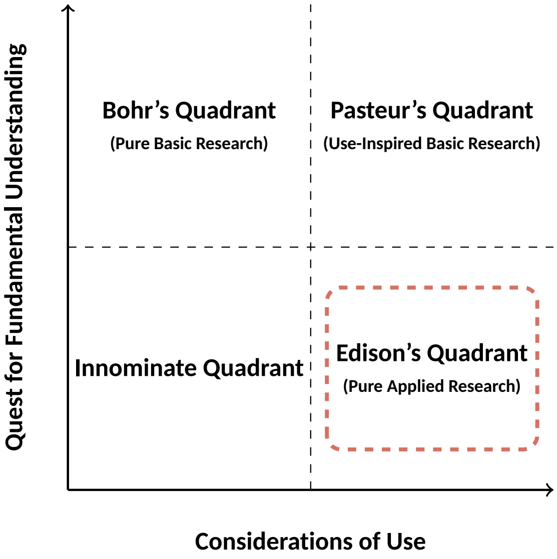


Figure 6: Pasteur's Quadrant: Relationship between fundamental understanding and practical use [15]. The diagram maps research efforts based on their pursuit of fundamental understanding (vertical axis) and considerations of practical use (horizontal axis). As highlighted, the conducted research focuses on practical applications without necessarily seeking deep theoretical insight—characteristic of pure applied research.

RQ 2: How effective is a multi-modal system in improving object detection and assessment performance under adverse and dynamic weather or environmental conditions? *Publication II and III*
Contributions: In **Publication II**, the integration of a camera with millimeter wave (mmWave) radar substantially improved object classification, particularly in conditions where the camera alone was rendered ineffective. These challenging conditions included rain, sleet, and nighttime scenarios, where visibility was severely compromised. By leveraging only the point cloud data generated by the mmWave radar, without relying on camera input, the system accurately distinguished between various vehicle classes.

In **Publication III**, a binocular vision system using color and infrared cameras was employed to estimate the size of fish using non-invasive methods. This approach enables biologists to collect quantitative data on aquatic ecosystems more efficiently and cost-effectively. By facilitating the analysis of critical ecological phenomena, such as the effects of hydropeaking, this system minimizes resource demands, including personnel time and effort, while simultaneously enhancing the scale and accuracy of biological assessments.

RQ 3: In what ways can the combination of machine learning and data-driven methods address challenges related to efficiency in embedded hardware with low-power restrictions? *Publication II and III*
Contributions: In **Publication II**, an approach was presented that integrated a machine learning-based object detection model with a point cloud-based classification model based on k-nearest neighbors (KNN). This hybrid methodology demonstrated a high degree of accuracy in situ, while maintaining a computationally efficient design. The solution was successfully deployed on low-power, resource-constrained embedded hardware, achieving near real-time operation. This outcome highlights the effectiveness of integrating machine learning algorithms with data-driven methods in addressing performance and efficiency

challenges within the context of low-power embedded systems. The developed framework not only advances the state-of-the-art in lightweight computational models but also highlights the practical feasibility of deploying sophisticated algorithms in constrained environments.

In **Publication III**, an approach was presented that integrated a machine learning-based object detection model with data-driven binocular vision techniques to estimate the distance from the camera and the size of the fish. Such a hybrid methodology demonstrated a high degree of accuracy in situ while maintaining a computationally efficient design. The solution was successfully deployed on low-power, resource-constrained embedded hardware, achieving near real-time operation. This outcome highlights the effectiveness of integrating machine learning algorithms with data-driven methods in addressing performance and efficiency challenges within the context of low-power embedded systems. The developed framework not only advances the state-of-the-art in lightweight computational models but also highlights the practical feasibility of deploying sophisticated algorithms in constrained environments.

The main focus of this dissertation are **novel computationally lightweight and adaptive computer vision pipelines, dedicated to supporting and improving object detection and assessment performance and efficiency in adverse weather and environmental conditions, on land and underwater.** The research encompasses a range of fields, including sensors, machine learning, and hardware as depicted in Fig. 7. Moreover, it is intentionally crafted to be highly adaptable for deployment across various applications and domains, while maintaining feasibility from both practical and economic perspectives. As previously hinted in Fig. 1, the effectiveness of the pipeline will be rigorously tested and validated in two distinct domains: infrastructure-based traffic monitoring and freshwater ecosystem monitoring. These diverse use cases underscore the interdisciplinary nature of outdoor monitoring systems, where successful implementation relies on the integration of core components, including sensors, machine learning, and specialized hardware. The Venn diagram illustrated in Fig. 7 shows the interplay among these components, outlining the foundational pillars that support this research.

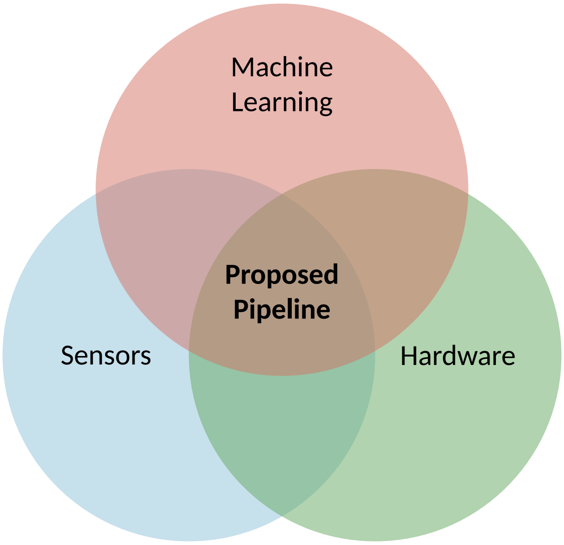


Figure 7: The research presented integrates Machine Learning, Sensors, and Hardware, highlighting their intersection and comprehensive scope.

1.6 Thesis Organization

The remainder of the dissertation is organized into four chapters, addressing the identified research gaps and the formulated research questions:

Chapter 2 presents a novel environmental classification model that enables the pipeline to adapt effectively to mitigate and alleviate the influence of environmental factors, thus improving object detection performance in challenging and complex environments.

Chapter 3 investigates how a multi-modal solution, composed of mmWave radar and a high-resolution camera, can improve object detection and classification performance in traffic monitoring during adverse weather conditions. The presented work also addresses the limited number of publicly available datasets for infrastructure-based traffic monitoring by introducing a novel dataset that encompasses a variety of vehicle classes, adverse weather conditions, and traffic configurations.

Chapter 4 presents an innovative solution that uses stereo color and infrared cameras to evaluate the impacts of hydropeaking on freshwater ecosystems. This system integrates machine learning-based object detection with data-driven techniques for depth and size estimation, allowing for accurate measurement of fish size in their natural habitats. Notable contributions include the development of a novel data set that encompasses multi-modal imagery.

Chapter 5 concludes the dissertation offering a comprehensive summary of the research questions, objectives, and solutions developed. It also addresses the challenges and limitations encountered during the research, while exploring potential avenues for future research and development.

2 Environmental Classification Model

2.1 Background and Motivation

All monitoring systems must fulfill a primary purpose: collecting, analyzing, and interpreting data to facilitate informed decision-making. In the case of fish monitoring, it means collecting information about fish presence / absence, species size, and migration direction. However, the underwater environment is inherently dynamic and unpredictable, shaped by constantly shifting currents, variable light penetration, and other factors. Temperature, salinity, and flow changes contribute to a complex ecosystem where visibility and conditions can fluctuate rapidly. Even in seemingly stable conditions, subtle disturbances can alter the behavior of the freshwater biota and the physical properties of the water column within moments, potentially influencing the performance of the monitoring system.

State-of-the-art solutions [55, 44, 36, 23] and commercially available fish monitoring systems such as the River Watcher (Vaki, Iceland) [61], Bravo G3 (Biotactic, Canada) [12], and the Yanmar Marine System (Yanmar, Japan) [99] have several deficiencies (high cost, infrastructure requirements) and poor performance, especially in harsh and unpredictable environments. Freshwater ecosystems are among the most biodiverse yet most threatened habitats worldwide. While supporting approximately one-third of all vertebrate species, but are undergoing decline due to accelerated wetland destruction, climate change, and anthropogenic pressure [20, 84, 30]. Although prior efforts have made substantial progress in fish detection, they often lack adaptability to dynamic environmental conditions, limiting their performance and scalability. These reasons guide the research and development of **novel computationally lightweight and adaptive computer vision pipelines dedicated to supporting and improving object detection and assessment performance and efficiency in adverse weather and environmental conditions**, using freshwater fish monitoring as one of its target applications. The work covered in *Publication I* focuses solely on the preprocessing stage, which uses environmentally adaptive machine learning methods, considerably different from those of previous publications [55, 44, 36, 23]. This work directly addresses *RG1: The effectiveness of object detection, assessment, and evaluation is significantly impeded in underwater environments and adverse weather conditions* and tries to answer *RQ 1: What are the potential benefits of employing an environmental condition classification model for object detection and assessment in adverse and harsh environments?* This will be the first, yet crucial, step in the proposed pipeline, which is dedicated to supporting monitoring capabilities, both detection and assessment, under challenging environmental conditions.

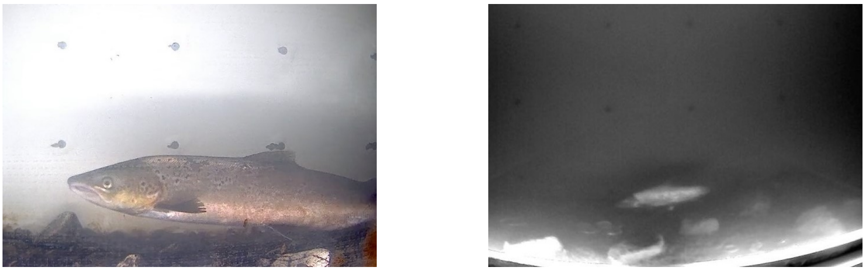
2.2 Environmental Conditions

Underwater environments present a unique set of conditions that influence visibility and perception. This work will focus on six distinct environmental conditions: clear visibility, low lighting, air bubbles, periphytic biofilm growth, turbidity, overexposure, and various combinations. Fig. 8 provides a visual representation of these environmental conditions to enhance understanding [81].

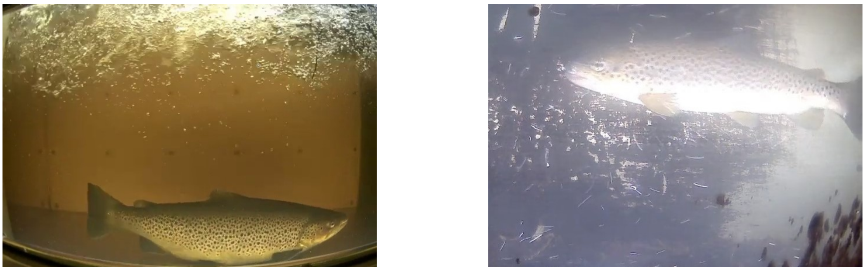
- **Periphytic Biofilm**: A condition characterized by the accumulation of organic and inorganic matter on the glass surface, forming a static layer that may obscure visibility.

- **Bubbles**: The persistent entrainment of air within the water column, resulting in the transportation of air pockets across the imaging field and potentially affecting visual clarity.

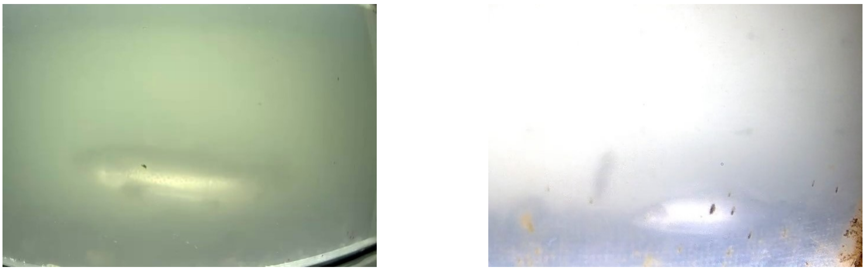
- **Low Light**: A scenario in which inadequate illumination adversely affects image quality by reducing visibility within the observation counter, typically due to insufficient external or internal lighting.
- **Overexposure**: A condition arising from excessive illumination leading to image saturation. This issue is commonly induced by internal lighting sources within the counter or by direct sunlight entering the water from a low angle.
- **Turbidity**: Attenuation of light transmission caused by suspended fine sediment particles and biological organisms that absorb and scatter illumination, producing a diffuse and hazy imaging environment.
- **Clear**: The optimal environmental condition, characterized by minimal interference with light transmission through the water column and the glass surface, ensures maximal imaging clarity.



(a) Clear *(b) Low-light*



(c) Bubbles *(d) Periphytic biofilm*

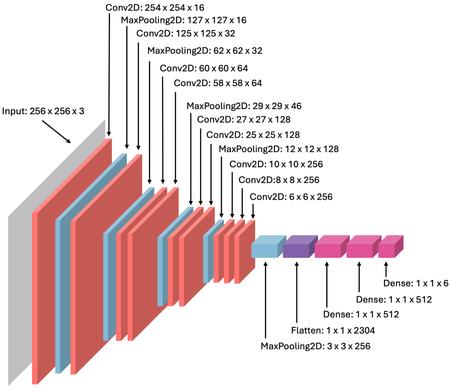


(e) Turbidity *(f) Overexposure*

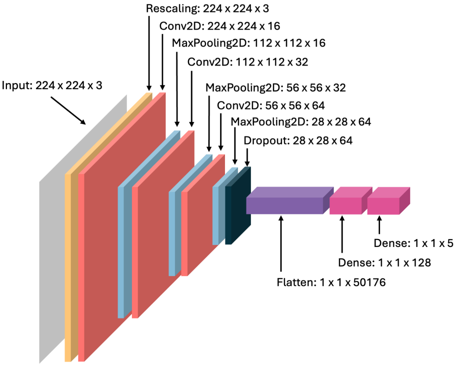
Figure 8: Examples of the six different environmental conditions. In the majority of situations, multiple conditions co-occur, with the exception of the clear condition. For example, the environmental condition overexposure (f) also encompasses biofilm growth (d) and turbidity (e) [81].

2.3 Environmental Condition Classification

The initial environmental condition classification model (ECCM) was based on a modified Visual Geometry Group 16-layer CNN (VGG-16) architecture [78], taking inspiration from research on weather conditions from single images [22, 45, 94]. The initial version of ECCM (ECCMv1) was designed to classify six environmental conditions, providing predicted class labels and associated probabilities. The architecture developed for this model is illustrated in Fig. 9 (a). However, several limitations of ECCMv1 were recognized, particularly concerning generalization performance and sensitivity to specific visual features. Consequently, the second version (ECCMv2) balanced computational efficiency and predictive performance, thus facilitating deployment on low-power, resource-constrained hardware were investigated further. The architecture of ECCMv2 is shown in Fig. 9 (b). Both versions (ECCMv1 and ECCMv2) were trained and validated in an identical process, visualized in Fig. 10.



(a) ECCMv1 architecture



(b) ECCMv2 architecture

Figure 9: (a) ECCMv1 (VGG16-based architecture): A deep convolutional neural network featuring multiple convolutional and max-pooling layers followed by fully connected dense layers, processing an input of size 256 × 256 × 3. The model increases feature depth progressively and concludes with three dense layers, outputting six classes. (b) ECCMv2 architecture: A custom convolutional neural network optimized for compactness and performance, taking an input of size 224 × 224 × 3. It includes fewer convolutional and pooling layers, a dropout for regularization, and a simplified fully connected structure ending in a five-class output.

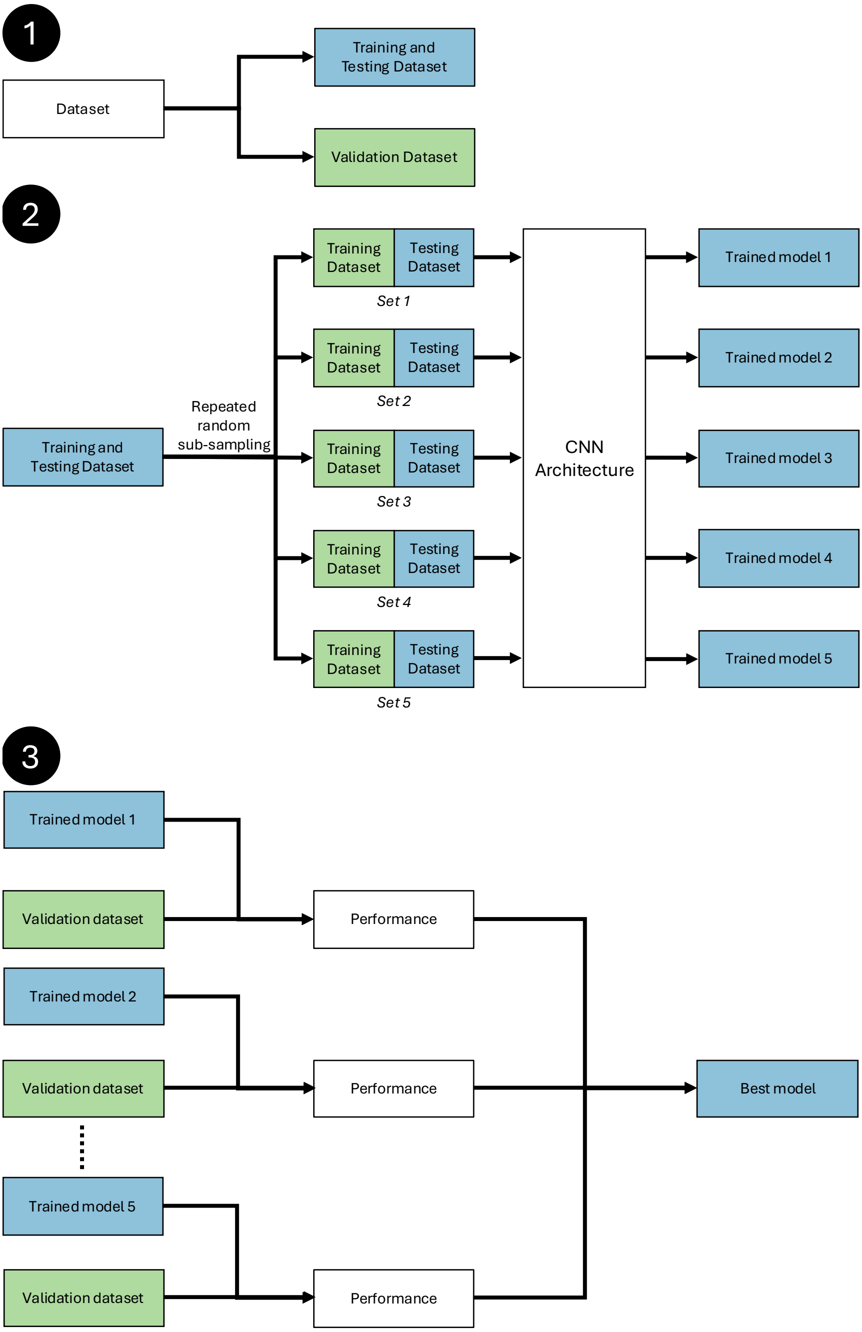


Figure 10: Illustration of the hold-out procedure used for training, testing, and validation of the environmental condition classification model. Separation of the dataset into training and hold-out validation datasets (1). Repeated random sub-sampling was applied for testing and training, and resulted in five CNN models with identical model architecture (2). The best-performing CNN model in terms of accuracy was used in the fish or no-fish video classification method. The top three environmental conditions, ranked by their probabilities, used to evaluate the model accuracy (3).

2.4 Results and Discussion

Table 1 depicts a preliminary overview of the environmental classification models, covering the model parameters and performance evaluation criteria. The first version (ECCMv1) demonstrated high accuracy, achieving a 99.2% accuracy on average rate across all environmental conditions, excluding *clear*. The second version shows similar performance, reaching 98.7% on average, being 0.5% less accurate. However, as depicted in Table 1, the second version was tested and evaluated on an enhanced dataset that featured a four times higher number of samples. Another difference is the input image size, which was slightly decreased, dropping down from 256 × 256 to 224 × 224.

Table 1: Comparison of both environmental condition classification models, ECCMv1 and ECCMv2, across various evaluation metrics. As depicted in Fig. 11, both behave similarly when trying to classify various environmental conditions. However, ECCMv2 is efficient, on average taking only 12 ms instead of 19 ms. Inference time measurements were taken using the Nvidia Jetson Orin Developer Kit.

Metric	**ECCMv1**	**ECCMv2**
Model size (MB)	36.7	75.6
Parameter count (M)	3.2	6.44
Size (pixels)	256 × 256	224 × 224
Inference time (ms)	19	12
Number of samples	3000	12000

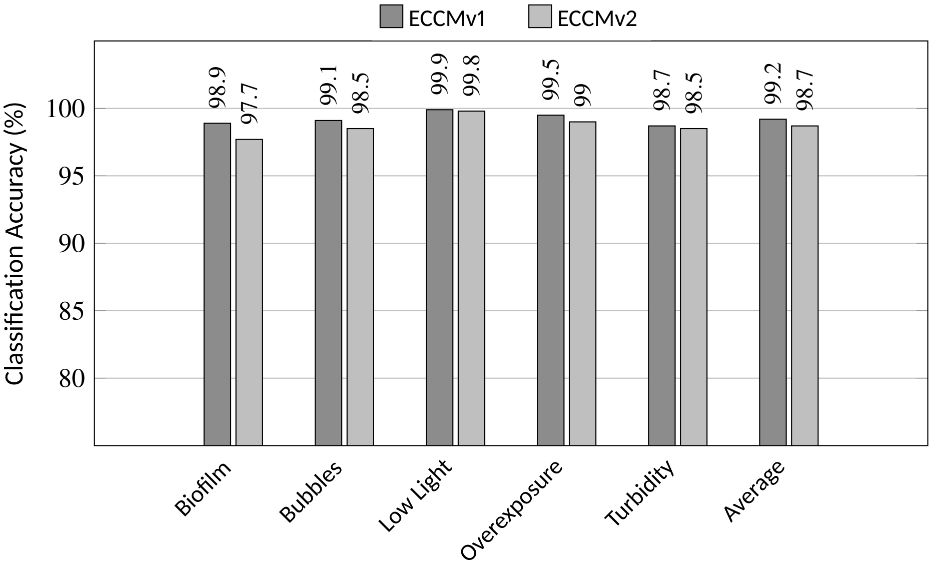


Figure 11: Classification accuracy (%) of two environmental classification models (ECCMv1 and ECCMv2) under various underwater image degradation conditions: biofilm, bubbles, low light, overexposure, and turbidity. Both ECCMv1 and ECCMv2 exhibit high and stable accuracy across all environmental conditions, with ECCMv1 based achieving the highest performance overall by a tiny margin, however as described in Table 1, ECCMv2 has a faster inference time.

Publication I (ECCMv1) and the accompanying unpublished research (ECCMv2) addressed two key gaps related to object detection and assessment in challenging environments. Specifically, they respond to the reduced effectiveness of detection in underwater and adverse weather conditions (**RG1**) by introducing an environmental condition classification model that enhances adaptability and accuracy. In addition, they address the challenge of high computational demands on resource-constrained hardware (**RG2**) by developing a model that remains lightweight while maintaining strong performance. In answering the research question (**RQ1**), the findings demonstrate that incorporating environmental condition awareness into the detection process offers clear benefits, achieving high accuracy alongside computational efficiency.

These results highlight that it is both feasible and effective to enhance object detection systems with context-awareness without exceeding practical hardware limits, providing a path toward real-world deployment in field conditions where both environmental variability and limited processing power are significant constraints. As the first stage in a broader outdoor monitoring pipeline, this work establishes a solid and scalable foundation for subsequent stages, ensuring reliable performance in diverse environments and contributing to the development of intelligent, efficient monitoring systems. This work advances the state of the art in object detection under challenging environmental conditions by introducing context-aware environmental condition classification models that are both highly accurate and computationally efficient, making them suitable for resource-constrained hardware.

In response to **RQ1**: "What are the potential benefits of employing an environmental condition classification model for object detection and assessment in adverse and harsh environments?", the findings demonstrate that integrating environmental conditions into the detection pipeline significantly improves both adaptability and reliability. The models are shown to enhance object detection performance under variable conditions while maintaining computational efficiency, thus enabling practical deployment in real-world field scenarios. In conclusion, the key contributions are the following:

- Environmental classification context-awareness is integrated into the monitoring workflow, improving object detection robustness in dynamic and visually degraded environments, thus answering RQ1 and addressing RG1, as well as supporting and improving real-world monitoring in harsh and dynamic environments.
- Prediction accuracy greater than 97% in five challenging environmental conditions, while maintaining computational efficiency, with lightweight classification models, enabling deployment on hardware with limited processing power.

3 Traffic Monitoring

3.1 Background and Motivation

Terrestrial environments may initially seem less challenging compared to underwater environments; however, they present substantial complexities due to environmental and weather variability that can hinder object detection or assessment performance. Conditions such as dense fog, heavy rain, or sleet limit the visual range and clarity, compromising the effectiveness of camera-based monitoring.

Camera-based traffic monitoring systems are widely adopted for their high spatial resolution, detailed feature extraction capabilities, flexibility in installation, and relatively low maintenance requirements. However, as already mentioned, their performance significantly degrades under adverse weather conditions. Alternative options, such as in-roadway (inductive loop detectors (ILD) [50, 63, 66, 93], magnetic [95, 21]) or non-intrusive, side- and over-roadway traffic monitoring systems, are typically comprised of acoustic sensors, light detection and range (LiDAR), or radio detection and range (radar) also have their unique advantages as well as disadvantages. Therefore, most state-of-the-art solutions employ sensor fusion-based approaches, typically involving a camera, which leverages complementary data from other modalities to enhance system reliability. Despite its potential, it also introduces new challenges and obstacles.

The objectives of this work were two-fold: first, to continue developing the pipeline, improve the performance of the outdoor monitoring systems in harsh and complex conditions, and deploy it on low-power, constrained embedded hardware. Another obstacle in developing, testing, and validating multi-sensor-based traffic monitoring systems is the limited number of openly available datasets for infrastructure-based traffic monitoring. The work complements existing open datasets by providing novel camera and mmWave radar data, covering multiple weather conditions, locations, and vehicle classes commonly found in urban traffic monitoring locations.

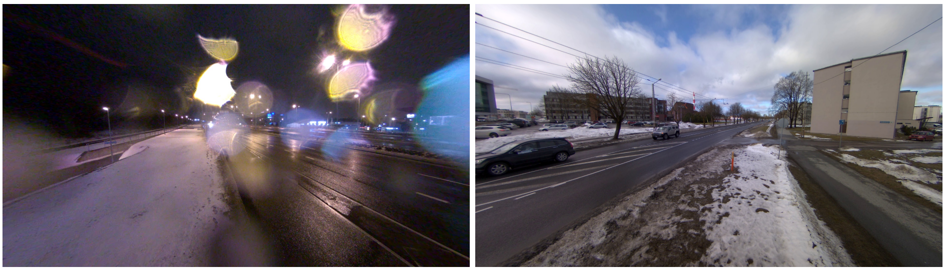
The work conducted and discussed in the following subsections aims to address all identified research gaps. To be more concrete, two research questions were devised to guide the work presented in *Publication II*. **RQ2**: *How effective is a multi-modality system in improving object detection and assessment performance under adverse and dynamic weather or environmental conditions?* **RQ3**: *How can machine learning and data-driven methods address efficiency challenges in low-power, resource-constrained embedded hardware?*

3.2 Dataset

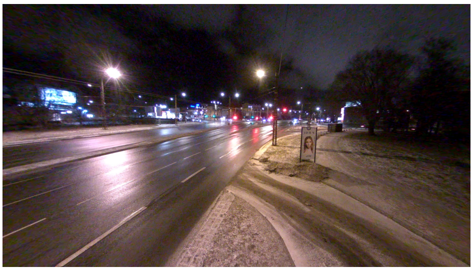
The novel dataset, titled Critical Infrastructure Monitoring (CIM), comprises 8,393 manually annotated frames, each synchronized with point cloud data obtained from mmWave radar. The recordings cover a variety of traffic scenarios, including intersections, merging zones, highways, and residential streets, captured during the late winter and early spring seasons. This temporal range reflects a broad range of weather and environmental conditions typical of temperate and humid continental climate regions. The dataset includes five primary weather categories: clear, cloudy, rainy, partially cloudy, and night (see Fig. 12), as well as mixed conditions. In addition to the diverse environmental conditions, the dataset includes vehicle annotations organized into four distinct classes: passenger cars, vans, busses, and trucks.

Openly available datasets are crucial to advance the research of computer vision and sensor fusion. Notable datasets such as *DAIR-V2X-I* [101], *A9* [24], *LUMPI* [17], and *Rope3D* [100] have contributed significantly to the field; however, they are predominantly based

(a) Clear - Järvevana tee *(b) Cloudy - Akadeemia tee (Raja junction)*



(c) Rain - Järvevana tee *(d) Partially cloudy - (Fujitsu sign)*



(e) Night - Kristiine (Intersection)

Figure 12: Examples from recording locations with different weather conditions. (a) Clear, ideal conditions of the roadway and vehicles. (b) Cloudy conditions, where some regions of the roadway have poor illumination at a distance. (c) Rain and other non-ideal conditions in which the camera lens may have water droplets, and where sections of the roadway may have blurred imagery. (d) Partially cloudy, dynamic changes in near and far-field illumination occur on the roadway due to variations in cloud cover. (e) Night, considerable variability in the roadway illumination levels due to static street lighting in conjunction with automobile head and tail lights.

on LiDAR-camera configurations. Although effective in many scenarios, LiDAR-based systems face limitations in adverse weather conditions, such as mist and fog, where sensor performance can degrade. In contrast, radar systems exhibit robust performance in such challenging environments [103]. Despite the advantage, publicly available datasets that incorporate radar-camera sensor fusion remain scarce. Datasets such as *TJRD TS*, *Radar LAB*, and *UTIMR* lack open access or provide limited scope, thus hindering broader research and development in this area. Table 2 presents a comparative overview of the datasets discussed. The proposed dataset offers several distinctive features:

- **Multi-modal data**: Synchronized point cloud and high-resolution images.
- **Diverse environmental conditions**: Collected across diverse weather and traffic

scenarios to support robust model evaluation.

- **Multiple vehicle classes**: Four distinct vehicle categories – passenger car, bus, truck, and van.

- **Open access**: Publicly available for academic and research use, promoting reproducibility and community-driven advancements.

Table 2: Comparison of open multi-modal infrastructure-based datasets for vehicle detection and classification. CIM provides the most extensive open dataset for camera and mmWave radar to date.

Dataset	**Frames**	**Resolution**	**Conditions**	**Vehicle classes**	**Modality**	**Access**
DAIR-V2X-I [101]	10084	1920x1080	Sunny Cloudy Nighttime Rain	Passenger car Truck Bus Van Motorcycle	Camera LiDAR	Open
A9 [24]	1098	1920x1200	Cloudy Snow Fog Sunny	Passenger car Truck Van Bus Motorcycle Trailer	Camera LiDAR	Open
LUMPI [17]	200k	1640x1232 1920x1080	Sunny Cloudy Night Rain	Passenger car Truck Van Bus Motorcycle Trailer	Camera LiDAR	Open
Rope3D [100]	50k	1920x1080	Clear Rain Night Dawn/Dusk	Passenger car Motorcycle Van Bus Truck Bicycle Tricycle Barrow	Camera LiDAR	Open
IPS300+ [90]	14198	1920x1080	-	Passenger car Bicycle Tricycle Bus Truck Engineer Car	Camera LiDAR	Open
RainSnow [10]	2200	640x480	Snow Rain Night Blizzard	Passenger car Bus Truck Van	Camera Thermal Camera	Open
TJRD TS [91]	-	-	-	Passenger car Bus Truck Van	Camera mmWave Radar	On request
Radar LAB [47]	8035	-	Clear Partially-cloudy	Passenger car	Camera Radar	Not available
UTIMR [96]	-	-	-	Small car[1] Medium car[1] Large car[1]	Camera Radar	Not available
CIM [80]	8393	1920x1080	Sunny Partially cloudy Rain/Sleet Cloudy Night	Passenger car Van Truck Bus	Camera mmWave Radar	Open

[1] Vehicles are classified by length: small car (L < 4.3 m), medium-sized car (4.3 m < L < 7 m), and large bus (L > 8 m).

3.3 Monitoring System Hardware

The components of the monitoring system were selected based on criteria such as computational performance, energy efficiency, availability, and suitability for machine learning tasks. At the core of the system is the Nvidia Jetson Orin Nano 4GB, chosen for its compact size and excellent computational performance-to-power consumption ratio. Featuring a hexacore ARM Cortex-A78AE CPU and a 512-core Ampere GPU with 16 Tensor Cores, enabling the deployment of machine learning-based models on the edge while consuming less than 15 W.

The monitoring system utilizes the AWR1843BOOST mmWave radar from Texas Instruments, which operates in the 77 GHz frequency band. The radar offers medium-range sensing capabilities that generate real-time point clouds. Its narrow beamwidth and the ability to configure parameters on the fly make it particularly suitable for dense urban environments, where fine-tuning might be necessary.

To complement the radar data with visual context, the system employs a Sony IMX-219-120 wide-angle camera. The camera delivers high-resolution imagery with a 120-degree diagonal field of view, enabling comprehensive coverage of the environment. A summary of the hardware components used in the final implementation is provided in Table 3.

Table 3: Overview of the hardware components used in the designed traffic monitoring system.

Component	**Model/Platform**	**Key Specifications**
Single board computer (SBC)	Nvidia Jetson Orin Nano 4GB	Hexa-core ARM Cortex-A78AE @ 1.7 GHz 512-core Ampere GPU with 16 Tensor Cores 4 GB LPDDR5 RAM MicroSD / NVMe Storage Support Supports MIPI CSI-2 Cameras Power consumption: <15 W
Radar	TI AWR1843BOOST mmWave	77 GHz mmWave automotive radar Range resolution: 0.586 m Velocity resolution: 1.33 km/h Max unambiguous range: 30 m (tested up to 100 m) Max velocity: 82.98 km/h Frame rate: 15 Hz; Azimuth: 15° Clutter removal enabled Onboard signal processing for point cloud generation
Camera	Sony IMX-219-120	Resolution: 3280 x 2464 pixels Aperture: f/2.2; Focal length: 1.79 mm Diagonal Field of View: 120° Low distortion: < 13.6%

Key Features of the Monitoring System Hardware

The selected hardware configuration for the monitoring system emphasizes deployability, performance, and robustness in real-world operational conditions. This system incorporates several key features that collectively facilitate efficient and reliable edge-based sensing and inference:

- **Edge AI Computing:** Utilizing the Nvidia Jetson Orin Nano, the platform provides real-time inference capabilities with minimal power consumption. This approach eliminates dependence on cloud-based processing, thereby enhancing latency performance and ensuring operational independence.

- **All-Weather Object Detection:** The integration of mmWave radar guarantees consistent detection performance, even in low-visibility and adverse weather conditions, thus maintaining reliability in challenging scenarios.

- **Enhanced Visual Context:** The system features wide-angle high-resolution imaging, which allows for accurate object classification across multiple traffic lanes.

- **Compact and Scalable Design:** The compact form factor and low power requirements support scalable implementation within distributed roadside infrastructures.

3.4 Vehicle Detection and Classification

The secondary contribution of *Publication II* is introducing a sensor fusion detection classification system that combines machine learning-based object detection with a data-driven point cloud classification. The core novelty lies in integrating two complementary sensing modalities to enhance robustness, particularly under adverse environmental and weather conditions. An object detection model based on the YOLOv7 architecture was trained on the complementary datasets discussed in Section 3.2, using a similar training process to that used in *Publications I* and *III*, shown in Fig. 10. The architecture was chosen based on the balance between Average precision (AP) and inference time. The trained model achieved a mean Average Precision (mAP@0.5...0.95) of 0.681, with F1 scores across vehicle categories ranging from 0.891 for buses and 0.819 for trucks.

By transforming radar data into the camera coordinate system through precise extrinsic and intrinsic calibration, a semantically labeled radar dataset was constructed. The proposed model demonstrated robust classification performance, achieving F1 scores of 0.85 for the "car" category and 0.83 for the combined "bus/truck" class. These results confirm the system’s effectiveness in maintaining high classification accuracy, even under conditions where visual input may be degraded or unreliable.

These preliminary results support the research addressing RG1 and conclusively answer RQ1, which investigates the effectiveness of a multi-modality system in enhancing object detection under dynamic and adverse conditions. The demonstrated fusion approach provides perceptual redundancy and enables fallback operation, ensuring continued performance when one sensor becomes degraded.

3.5 Results and Discussion

The system demonstrates strong detection and classification performance that aligns with the defined requirements. However, computational efficiency remains a critical consideration, particularly given the challenges of deploying complex monitoring solutions on resource-constrained hardware, as emphasized in **RG2**. To evaluate this aspect, camera and radar subsystems were tested on the Nvidia Jetson Orin Nano — an embedded platform representative of edge-computing scenarios. The YOLOv7 model sustained 20 frames per second (FPS), around 20 ms, with a power draw of approximately 8.9 W, resulting in a performance-per-watt (PPW) of 2.25. The radar classifier, benefiting from its lower computational complexity, achieved an inference time of only 3.3 ms and a PPW of 52.24. These results reinforce the proposition in **RQ3** that combining deep learning with lightweight data-driven methods can produce an efficient architecture suitable for low-power real-time deployment. Table 4 summarizes the performance of the system on the Jetson Orin Nano. The camera model provides high semantic accuracy, while the radar classifier offers excellent throughput and energy efficiency. Together, these components confirm the system’s ability to operate a fairly complex monitoring system effectively on constrained embedded hardware in real-time.

Arguments could be made that, if the trained model using only point cloud information

Table 4: Summary of performance metrics across the vehicle perception pipeline. The table presents key detection and classification results for both subsystems of the sensor fusion architecture. The camera module (YOLOv7) was evaluated for multiclass vehicle detection accuracy, localization robustness (mAP), and embedded inference speed and power consumption. The radar classification module (KNN) was evaluated for binary classification accuracy across car and bus/truck classes, and benchmarked for high-speed, energy-efficient performance on embedded hardware (Jetson Orin Nano). Metrics are reported using the best-performing configurations selected during validation.

Component	Metric	Result	Summary
Camera Detection (YOLOv7)	Precision (All classes)	0.805	Reliable and balanced multiclass detection with accurate localization across four vehicle types.
	Recall (All classes)	0.798	
	F1 Score (All classes)	0.801	
	mAP@0.5 / mAP@0.5...0.95	0.850 / 0.681	
	Inference Time	20 FPS / 50 ms	Achieves real-time inference on embedded hardware.
	Power Usage	8.9 W	Efficient power consumption suitable for edge deployment.
Radar Classification (KNN)	Precision (Car / Bus+Truck)	0.79 / 0.90	Accurate binary classification using sparse radar input, with strong recall for cars and high precision for larger vehicles.
	Recall (Car / Bus+Truck)	0.92 / 0.76	
	F1 Score (Car / Bus+Truck)	0.85 / 0.83	
	Inference Time	303 FPS / 3.3 ms	Extremely fast and lightweight inference suitable for high-throughput processing.
	Power Consumption	5.8 W	Highly energy-efficient for continuous embedded operation.

could already accurately distinguish multiple vehicle classes in harsh weather [103], what is the point of the camera? As mentioned earlier, detection and classification are only a small part of the monitoring system. Unlike other sensing modalities, camera sensors capture features, such as color, shape, and luminance, which provide a unique advantage for evaluating objects or surrounding environments. In situations where the camera is rendered unusable, mmWave can be used as a backup to collect traffic data.

Together, the system demonstrates how sensor fusion can overcome limitations of single-modality approaches, maintain classification accuracy in varied conditions, and run efficiently on embedded platforms. It provides a reproducible, modular framework with practical utility for intelligent transportation systems, roadside infrastructure, and autonomous platforms. Beyond model performance, the work also addresses **RG3**, which highlights the lack of publicly available datasets featuring radar data labeled under challenging environmental conditions. Although creating a public dataset was not within scope, a novel method was developed to annotate radar point clouds using aligned camera detections. This semi-automated labeling process resulted in a custom radar dataset of 423 samples, providing a scalable approach for future dataset development efforts in multimodal perception. Key contributions and novelty of this work include:

- Open access dataset, featuring annotated and synchronized radar point cloud information with high-resolution imagery.
- A sensor fusion architecture combining YOLOv7-based camera detection and KNN-based radar classification for robust, real-time vehicle classification.
- Demonstrated resilience of the system under adverse environmental conditions using complementary sensing modalities.
- Development and validation of an energy-efficient perception pipeline that can be deployed on low-power embedded hardware (Nvidia Jetson Orin Nano 4GB).

4 Object Assessment

4.1 Background and Motivation

Object detection and classification represent only a portion of a comprehensive monitoring system. Object assessment is crucial for evaluating the condition, behavior, characteristics, and attributes of objects, allowing informed decision-making. The accuracy with which a system can estimate and assess these parameters directly influences its decision-making capabilities, which can have significant repercussions within a domain and its adjacent fields. As previously highlighted, detection in underwater environments poses considerable challenges. Introducing assessment adds further complexity, often resulting in trade-offs between performance and efficiency. Nevertheless, insights derived from *Publication II* illustrate that a machine learning-based approach to object detection, combined with a data-driven methodology, can achieve precision and efficiency, even when implemented on low-power, resource-constrained hardware. The findings of this previous research have profoundly shaped and influenced the work presented in *Publication III*.

The primary motivation behind *Publication III* stems from the ability to analyze and evaluate the effects of hydropeaking in freshwater ecosystems. Hydropeaking results from rapid and frequent flow fluctuations caused by intermittent water releases through turbines to meet peak energy demand. These fluctuations alter flow patterns, affect water temperature, affect sediment transport, and change dissolved gas levels within ecosystems downstream of hydropower operations. These alterations affect various aspects of aquatic ecosystems, including fish growth, behavior, reproductive success, habitat, and migration patterns, among others [13, 42]. Considering fish communities, these fluctuations have been reported to cause lateral and longitudinal displacements, leading to habitat shifts, reducing the survival rates due to stranding, and disrupting key life-cycle events such as growth, reproductive migration, and spawning. Furthermore, hydropeaking can also lead to habitat fragmentation, erosion, and loss of riparian vegetation, impacting terrestrial ecosystems that depend on the aquatic environment [11]. However, there is limited understanding of the long-term ecological consequences of hydropeaking and its cumulative effects on aquatic ecosystems [13, 75].

Fortunately, fish length can reveal much about population structure, including growth rates, age distribution, juvenile-to-adult ratio, and overall weight [70, 35]. These metrics can provide significant information about the ecosystem, allowing a complete understanding of its dynamics and overall well-being. However, performing automated size estimation in situ remains challenging for several reasons. A common cause of poor size estimation accuracy is the motion and orientation of the body of a swimming fish. The presence of foreign objects or other fish can cause partial occlusion, making it difficult to obtain accurate size estimates. The findings of these previous studies highlight the need for a non-invasive, in-situ camera-based monitoring system capable of estimating fish size.

The objectives of this work were two-fold: first, to continue developing the pipeline and introduce the ability to assess objects, and second, to demonstrate the capabilities by studying the effects of hydropeaking. As stated before, a significant obstacle in developing, testing, and validating outdoor monitoring systems is the limited number of publicly available datasets. The work complements existing open datasets by providing novel in situ multi-modal data, comprised of RGB and IR imagery, originating from two freshwater rivers located in Portugal. Conducted work discussed in the following subsections addresses all identified research gaps - (RG1, RG2, and RG3) and research questions RQ2 and RQ3, which drove the research.

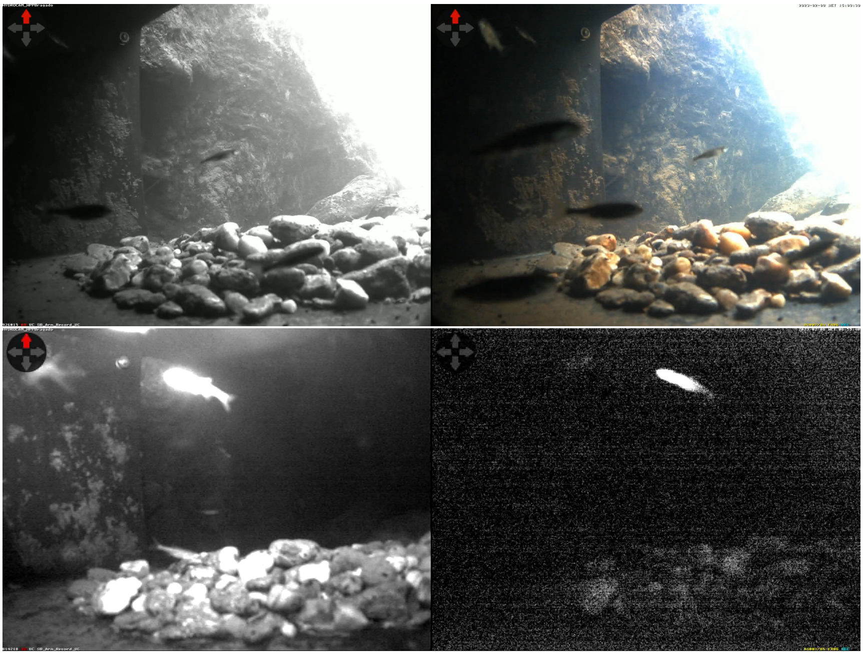
4.2 Dataset

The footage utilized in the PTFish dataset was acquired through the research initiative EcoPeak4Fish [14]. The recordings were collected during spring and late summer periods from two separate sites: Bragado (Fig. 13 (a)), located in the Avelames River, and Covas Do Barroso in the Couto River (Fig. 13 (b)), both tributaries of the Tamega River (Douro River basin), Portugal. The curated dataset contains 18,523 manually annotated frames from IR and RGB cameras.

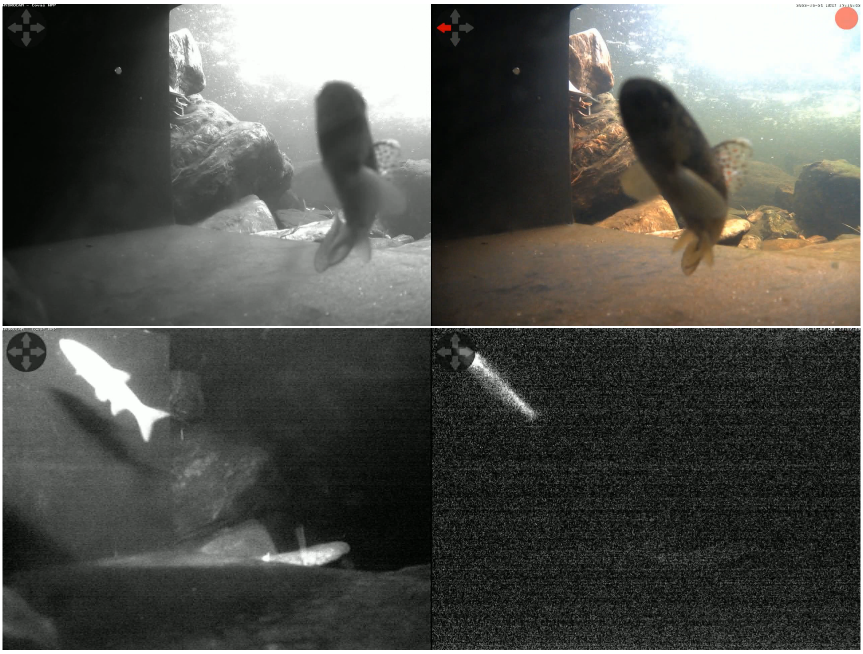
Existing open-access datasets, such as *DeepFish* [74], *Fish4Knowledge* [34], and *AffiNe* [86], exhibit several notable limitations, as summarized in Table 5. Upon evaluation, it is evident that the majority of these datasets are dedicated to marine species, rendering them unsuitable for the development of methods applicable to freshwater environments. In addition, many data sets are based on samples collected in laboratory settings (*in vitro*) [6]. In contrast, others consist of *ex vivo* data derived from deceased specimens [69, 86], thus limiting their applicability to ecological monitoring in situ. A significant distinction among these datasets is that the PTFish dataset provides stereo imagery, which is particularly advantageous for depth perception and three-dimensional understanding—key components for tasks such as size estimation (SzE). In summary, the proposed data set differentiates itself from others by offering the following features:

- **Stereo Imaging**: Multi-modal image data incorporating both RGB and IR, facilitating depth perception and enhancing accuracy in size estimation tasks.
- **High Resolution**: A resolution of 2560 × 960 is provided, supporting detailed visual analysis.
- **Freshwater Focus**: Among the very few datasets containing freshwater data, it is the only one with in situ (natural habitat) freshwater capture.
- **In Situ Capture**: Unlike ex vivo or in vitro datasets, this dataset features images directly captured in natural environments, preserving real-world behaviors and context.

A detailed comparison highlighting the key differences between PTFish and other publicly available datasets is presented in Table 5. As illustrated, PTFish uniquely combines multi-modal data, ecological authenticity, and a freshwater focus—characteristics that are rarely found together in existing datasets. This dataset offers significant novelty by capturing fish in natural settings, thereby integrating real-world environmental challenges such as water turbidity and fluctuating lighting conditions. These elements are vital for both the creation and assessment of computer vision techniques designed for ecological purposes, especially in complex aquatic settings. One of the primary contributions of this work is the introduction of the PTFish dataset, an openly accessible multi-modal resource featuring both infrared (IR) and RGB imagery. This dataset addresses a significant gap in the availability of resources specifically designed for freshwater fish monitoring and size estimation tasks. By enabling the development of more adaptable and accurate object detection methods, it supports broader scientific objectives and fosters enhanced collaboration within the research community.



(a) Bragado during the day and at night. Left images are taken from the infrared camera, right images are from the color camera.



(b) Covas Do Barroso during the day and at night.

Figure 13: Examples of recording locations situated in tributaries of the Tamega River (Douro River basin), Portugal: (a) Bragado, located in the Avelames River. (b) Covas Do Barroso, located in Couto River [79].

Table 5: Comparison of open access fish datasets for computer vision tasks. In situ: on site, in vitro: in lab, ex vivo: on dead specimens. ObD: object detection, FiC: classification (fish/ no fish), SpC: species classification, SzE: size estimation, Seg: segmentation. The number of frames corresponds to the number of available images before augmentation [79].

Dataset	**Environment**	**Task**	**Number of Frames**	**Resolution**	**Mono/Stereo**
DeepFish	in situ/marine	ObD Seg	39766	1920 × 1080	Mono
Rockfish	in situ/marine	ObD	4307	1280 × 720	Mono
Fish4Knowledge	in situ/marine	ObD	27370	352 × 240	Mono
QUT	in vitro ex vivo/marine	SpC	3960	480 × 360	Mono
Brakish	in situ/marine	ObD SzE	14518	1920 × 1080	Mono
AFFiNe	ex vivo/freshwater	ObD SpC SzE	7000	710 × 852	Mono
PTFish	in situ/freshwater	ObD SzE	18523	2560 × 960	Stereo

4.3 System Hardware Overview

The embedded vision system developed combines a compact, low-power compute module with a dual-camera sensor setup. The hardware configuration, presented in Table 6, was chosen to prioritize deployability, performance, and robustness under real-world conditions. At the core of the system is the Nvidia Jetson Orin Nano Developer Kit, selected for its optimal balance between AI processing capability and energy efficiency. This module features integrated Hexacore ARM microprocessor, complemented by an Ampere architecture-based GPU with dedicated Tensor Cores, facilitating real-time inference and vision-driven workloads.

The vision component is based on a stereo binocular camera system that consists of a single RGB sensor and an infrared (IR) sensor. Both sensors exhibit identical characteristics, thereby enabling synchronous multi-modal imaging during day and night. This configuration is particularly advantageous for applications that require depth perception, environmental awareness, or enhanced visibility under varying lighting conditions. Key features include:

- **Edge AI computing:** The Nvidia Jetson Orin Nano provides real-time on-device inference with minimal power consumption, thereby negating the necessity for cloud-based processing.

- **Rich visual context:** The Mobotix camera sensors provide high-resolution RGB and infrared imagery with an extensive field of view, supporting reliable classification across diverse lighting environments.

- **Compact and scalable:** All components are designed for embedded deployment, characterized by low power consumption and a small form factor that is ideal for hard-to-reach and remote locations.

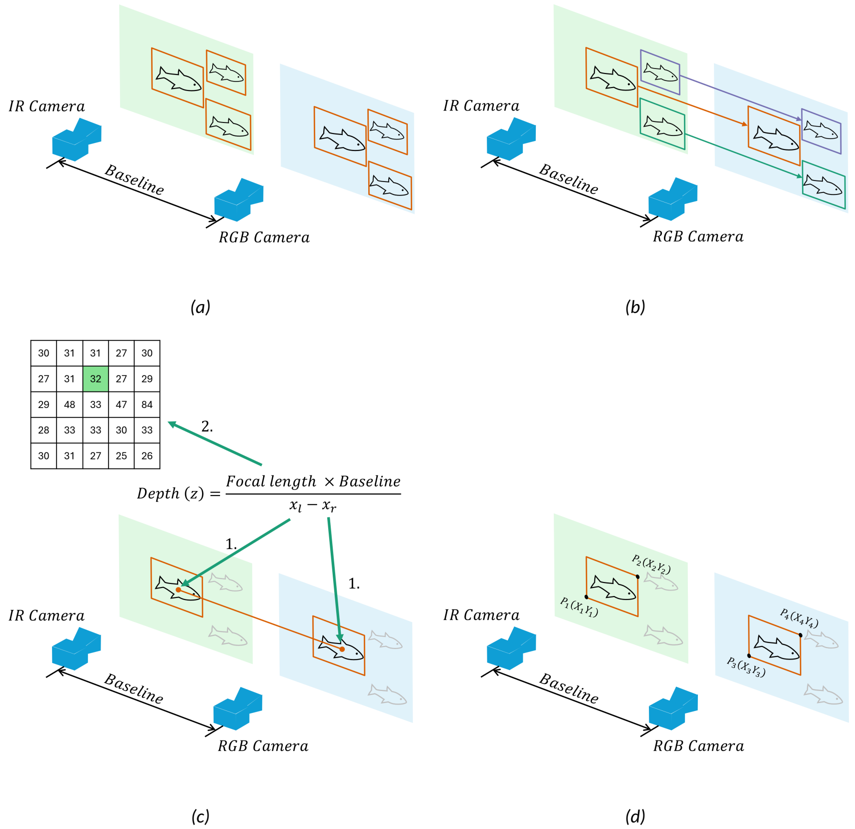
Table 6: Overview of the hardware components used in the designed embedded vision system.

Component	**Model/Platform**	**Key Specifications**
Embedded Processing Unit	Nvidia Jetson Orin Nano 8GB	Hexa-core ARM Cortex-A78AE @ 1.5 GHz 1024-core Ampere GPU with 32 Tensor Cores 8 GB LPDDR5 RAM MicroSD / NVMe Storage Support Power Consumption: < 15W
Camera Sensor (RGB & IR)	Mobotix Mx-O-SMA-S-6N016	Resolution: 1280×960 Aperture: f/1.8; Focal Length: 4.1 mm Field of View: 90° (H), 67° (V)

4.4 Detection and Size Estimation

Size estimation is a multi-stage process that utilizes machine learning models, combined with data-driven methods, to remain computationally inexpensive. As presented in Fig. 14 (a), the pipeline detects fish within a multi-modal frame by integrating IR and RGB imagery through an object detection model based on the YOLOv8s architecture. Subsequently, as illustrated in Fig. 14 (b), the algorithm evaluates the quality of the bounding box pairs by analyzing their geometric and spatial attributes. The disparity between these bounding boxes is computed and subsequently utilized to estimate depth, with the focal length and baseline serving as critical scene parameters. To enhance the robustness of depth measurements, the median value (e.g. 32 in this specific instance) is extracted from a neighborhood of depth estimates at each pixel, thus mitigating the influence of potential outliers, as shown in Fig. 14 (c).

After retrieving the depth, the method calculates the coordinates x, y and z by transforming the depth data into three-dimensional space, as shown in Fig. 14 (d). The coordinates of the two corners of each bounding box are taken to compute the Euclidean distances associated with each bounding box. Finally, a size threshold is established to classify the fish as juvenile or adult, with all fish less than 10 cm categorized as juvenile.



30	31	31	27	30
27	31	32	27	29
29	48	33	47	84
28	33	33	30	33
30	31	27	25	26

(a)

(b)

(c)

(d)

Figure 14: a) Binocular vision system setup consisting of an IR and an RGB camera. First, the proposed solution detects and locates fish from a single frame, using both the IR and RGB cameras. b) After detecting fish on both images, the algorithm determines the quality of the bounding box pairs by analyzing their geometric and spatial properties. c) The disparity is first computed and subsequently utilized to estimate depth, leveraging the focal length and baseline as scene parameters. Following the depth estimation for each point, the median value (e.g., 32 in this example) is selected from a neighborhood of depth estimates at each pixel to mitigate the influence of outliers and ensure robust depth measurement. d) The fish size is determined by calculating the Euclidean distance utilizing the coordinates of the bottom left and top right corners (x and y) for both bounding boxes, which are then averaged to estimate the fish's size (total body length).

4.5 Results and Discussion

The dissertation presents a lightweight and adaptive multimodal system for real-time fish detection and size estimation in dynamic and resource-constrained aquatic environments. The system integrates IR and RGB imaging, state-of-the-art object detection models, and classical stereo vision to address the dual challenges of degraded visibility and limited computational resources. It was specifically designed for deployment in real-world environments using low-power embedded hardware. Table 7 provides a concise overview of the main experimental results of the object detection, depth/size estimation, and embedded performance evaluations. The proposed system successfully combines modern machine learning techniques with efficient computational methods to meet real-world constraints. It enables accurate detection and estimation of fish size in challenging conditions and

Table 7: Summary of performance metrics across the pipeline.

Component	Metric	Best Result	Summary
Object Detection	F1 Score	0.85	The selected object detection model showed good overall performance, but in terms of accuracy across 3,705-frame hold-out dataset.
	mAP@0.5	0.88	
Depth Estimation	Mean Absolute Error	0.83 cm (10×10), 0.63 cm (5×5)	Sub-centimeter accuracy confirmed with ArUco marker validation in 3 test scenes.
	Relative Error	2.74% / 1.54%	Consistent results regardless of marker size and position.
Size Estimation	Width Error	0.12 cm / 2.33% (5×5)	Most precise estimation observed in smaller markers.
	Height Error	0.27 cm / 5.47% (5×5)	Accurate body-length measurement using Euclidean distance from depth data.
Efficiency	Frames per second	10	Real-time frame rate achieved with high-resolution inputs (2560×960).
	Power Consumption	10.7 W	Supports deployment in battery-powered and remote environments.

on constrained hardware, fulfilling the stated research objectives. The solution lays a strong foundation for future extensions such as behavioral tracking, species recognition, or integration into broader ecological monitoring systems. Key contributions and novelty of this work include:

- **Multi-modal integration:** Combining synchronized IR and RGB camera streams to enhance object detection robustness under adverse environmental conditions such as turbidity or low lighting.

- **Lightweight depth and size estimation:** Implementing a computationally efficient stereo vision method using disparity, achieving sub-centimeter accuracy for object size estimation.

- **Real-time embedded deployment:** Validating the full pipeline on the Nvidia Jetson Orin Nano platform, achieving 10 frames per second with under 11W power consumption.

- **Field validation under hydropeaking conditions:** Applying the pipeline in real-world monitoring scenarios and discovering behavioral insights in fish activity not evident through visual review alone.

- **Open access data set:** Dataset for academic and research use, featuring in-situ synchronized IR and RGB camera footage, collected from two freshwater sites (Covas Do Barroso and Bragado). Open access promotes reproducibility and community-driven advancements.

5 Conclusion

The dissertation effectively accomplished its primary objective: to create a computationally efficient and adaptable system that improves the performance and efficiency of object detection and assessment in harsh and challenging weather and environmental conditions. Research questions guiding the work, discussed in Section 1.5, were formulated based on the initial literature review and the identified gaps. Each publication was centered on addressing one or more of these questions, each targeting specific deficiencies in the area.

As described in the initial literature review, **RG1** identifies a significant flaw: the effectiveness of object detection, assessment, and evaluation is considerably hindered in underwater settings and challenging weather conditions. Current advanced outdoor monitoring systems are based on deep learning models that are computationally intensive, thus restricting their practicality due to the high demand for computational resources. In *Publication I*, the proposed framework addressed this issue by introducing an innovative model for classifying environmental conditions. This model can differentiate between six specific environmental conditions—clear, low lighting, air bubbles, biofilm growth, turbidity, and overexposure—and evaluate their intensity. The results directly address the **RQ1** (*What are the potential benefits of including an environmental condition classification model for object detection and assessment in harsh and adverse environments?*). The integration of this capability enables the system to dynamically adjust preprocessing parameters. This adaptability significantly enhances object detection performance in challenging and fluctuating underwater conditions. Although the proposed methodology was specifically demonstrated and validated in an underwater setting for fish monitoring and assessment purposes, its applicability extends to various other fields and applications.

In *Publication II*, the focus of the deployment environment moved from underwater applications to those on land, specifically targeting infrastructure-mounted traffic monitoring using cameras and mm-wave radar. This transition introduced new challenges and obstacles (**RG1**) that affected the system's overall effectiveness. An initial literature review revealed that existing solutions predominantly rely on sensor fusion, with the combination of LiDAR and camera technologies being the most common. Nevertheless, research conducted by Nagoya University pointed out significant limitations of LiDAR under challenging conditions such as mist and fog. In contrast, radar performance is minimally affected by adverse weather [103]. By integrating machine learning models with data-driven techniques, such as KNN-based point cloud classification combined with camera-based object detection, a robust solution was developed. This system accurately distinguishes vehicle types, even in severe weather conditions, such as low light, sleet, and rain, and when the camera functionality is impaired. These findings provide an adequate response to **RQ2** *(How effective is a multi-modal system in improving object detection and assessment performance under adverse and dynamic weather or environmental conditions?)*. Not only that, but the findings also address the problem of poor performance and efficiency on a resource-constrained hardware, resulting in trade-offs between one and another (**RG2**). The results provide an explicit answer to **RQ3** *(In what ways can the combination of machine learning and data-driven methods address challenges related to efficiency in embedded hardware with low-power restrictions)*, by showing that the approach has been successfully deployed on low-power, resource-limited embedded hardware platforms, achieving near real-time performance while maintaining detection accuracy and keeping power consumption under 15 W.

Publication III introduces a stereo vision system designed for non-invasive, on-site estimation of fish size to study the ecological effects of hydropeaking. This system utilizes a stereo camera setup that incorporates both RGB and infrared imaging. Unlike previous research, this study shows significant progress by validating the effectiveness of the system in a field environment. The innovative solution was tested at two field locations in Portugal—Bragado and Covas do Barroso. By integrating a camera-based object detection model with a computationally efficient, data-driven method for depth and size estimation, the system achieves accuracy within sub-centimeter levels in natural aquatic settings. This outcome directly addresses the aforementioned **RQ2**, emphasizing the challenges in object detection, assessment, and evaluation in underwater environments and challenging weather conditions. Moreover, the system has been effectively deployed on low-power, resource-limited embedded hardware platforms, achieving near real-time performance without sacrificing accuracy. These findings contribute to answering **RQ3**, focusing on the critical computational challenges associated with object detection and assessment methods on hardware with restricted processing power. The system's low energy requirements and minimal infrastructure demands further enhance its viability for use in remote or logistically challenging locations.

Moreover, *Publication II* and *Publication III* specifically tackle **RG3** (*the lack of publicly available datasets specifically crafted for the development and evaluation of object detection and assessment techniques in harsh and challenging environmental or weather conditions.*) A key component of this work is the creation of publicly accessible datasets designed to bridge this critical gap. These datasets are pivotal for advancing scientific research as they support reproducibility, facilitate benchmarking, and encourage collaboration. By providing standard and varied data, these resources enable researchers to validate algorithms, compare methodologies, and promote innovation within the field.

This dissertation presents a computationally efficient and adaptive pipeline for object detection and assessment in both outdoor and underwater environments. The goal is to enhance performance and efficiency in challenging weather and environmental conditions. This pipeline incorporates environmental condition classification, multimodal sensing, hybrid machine learning, and data-driven methodologies to enable adaptive processing based on the identified environmental state. By balancing detection accuracy with computational efficiency, the pipeline is optimized for use on resource-limited embedded systems, making it practically applicable for monitoring in both underwater and terrestrial environments. Its adaptive nature allows it to dynamically modify processing strategies in response to changing environmental conditions, thus optimizing performance while keeping computational demands low.

5.1 Limitations

Although the research employs an innovative methodology and has demonstrated advantages, it is not free of shortcomings. The datasets used in this study have several limitations. In particular, the dataset described in *Publication III* has limited species diversity and an uneven sample distribution between the Bragado and Covas do Barroso areas. Likewise, the CIM dataset from *Publication II* contains relatively sparse point cloud data, mostly consisting of a few vehicle classes, mainly cars, with a significantly smaller number of larger vehicle samples, such as busses and trucks. These limitations highlight the challenges inherent in collecting field data. In addition, the research on the environmental classification model is still underexplored by the community at large. Although the model has been successfully applied in underwater environments, it has yet to be tested or used for terrestrial monitoring applications, leaving its effectiveness on land unverified.

5.2 Future Work

Future research stemming from this dissertation is underway and is focused on several areas. The techniques outlined in *Publication III* are being applied in Portugal and will soon be used in Uzbekistan to monitor the health of freshwater ecosystems and track the fish migration of understudied native species. As previously mentioned, there is an effort to expand existing datasets by adding new recording locations, enhancing the diversity of species and vehicles, and balancing class distributions. For example, collecting data at busy intersections or in industrial areas could significantly improve the representation of different vehicle types and increase the generalizability of the models.

In addition, the author proposes developing a set of unified benchmarking tools designed specifically for outdoor monitoring systems. This toolkit would aim to standardize the evaluation of system performance in various environmental conditions, such as changes in weather, lighting, and terrain. The framework will enable objective validation of sensor accuracy, data transmission reliability, and overall system responsiveness by simulating realistic scenarios and providing consistent performance metrics. This strategy will support the development of more robust, scalable and interoperable outdoor monitoring solutions.

List of Figures

1 Outdoor monitoring applications: (1) Coral reef and fish monitoring: assessing reef health and detecting fish species using underwater imaging. (2) Traffic monitoring: identifying vehicles and buses for real-time traffic management. (3) Wildlife monitoring: identifying and tracking animals, such as bears and beavers, in natural habitats. The research covered in this dissertation focuses on infrastructure-based traffic monitoring and the evaluation of fish populations and migration. 11

2 A visual representation of a monitoring pipeline applied to traffic and fish ecosystem monitoring. The workflows progress through data acquisition, preprocessing (removal of irrelevant elements and improvement of visibility), object detection (with classification and confidence scores), object assessment (e.g., speed or size estimation), and response (e.g., issuing fines or assessing conservation status). 11

3 Timeline of advancements in object detection models, spanning from 2001 to 2025. The progression begins with traditional feature-based approaches, such as VJ (Viola-Jones) and HOG, advancing through the introduction of deep learning-based models such as AlexNet and R-CNN, and evolving toward modern architectures that are based on transformers, namely ViT (Vision Transformer) and RT-DETR. 12

4 Object Detection Accuracy on COCO 2017 Validation Set by Architecture Type (2015–2025). Each point represents a model's mAP score, grouped by architecture class. **Highlighted markers indicate the models tested and utilized during the research.** 13

5 AI performance (TOPS) vs. power consumption (W) of edge computing devices released between 2018 and 2024. Early models, such as the Jetson Nano (2019) and Coral USB (2019), offer limited performance at moderate power consumption. Newer devices, such as the Orin NX 16GB (2023) and Orin Nano Super (2024), show significant improvements, with the latter reaching 67 TOPS at 25 W. The devices featured in the graph were used at various stages throughout the research discussed in the dissertation. Although numerous devices are available from various manufacturers, as described in the publications, the selection criteria were primarily influenced by factors such as capabilities, availability, cost, and documentation [80, 79]. 15

6 Pasteur's Quadrant: Relationship between fundamental understanding and practical use [15]. The diagram maps research efforts based on their pursuit of fundamental understanding (vertical axis) and considerations of practical use (horizontal axis). As highlighted, the conducted research focuses on practical applications without necessarily seeking deep theoretical insight—characteristic of pure applied research. 17

7 The research presented integrates Machine Learning, Sensors, and Hardware, highlighting their intersection and comprehensive scope. 18

8 Examples of the six different environmental conditions. In the majority of situations, multiple conditions co-occur, with the exception of the clear condition. For example, the environmental condition overexposure (f) also encompasses biofilm growth (d) and turbidity (e) [81]. 21

9 (a) ECCMv1 (VGG16-based architecture): A deep convolutional neural network featuring multiple convolutional and max-pooling layers followed by fully connected dense layers, processing an input of size 256 × 256 × 3. The model increases feature depth progressively and concludes with three dense layers, outputting six classes. (b) ECCMv2 architecture: A custom convolutional neural network optimized for compactness and performance, taking an input of size 224 × 224 × 3. It includes fewer convolutional and pooling layers, a dropout for regularization, and a simplified fully connected structure ending in a five-class output. 22

10 Illustration of the hold-out procedure used for training, testing, and validation of the environmental condition classification model. Separation of the dataset into training and hold-out validation datasets (1). Repeated random sub-sampling was applied for testing and training, and resulted in five CNN models with identical model architecture (2). The best-performing CNN model in terms of accuracy was used in the fish or no-fish video classification method. The top three environmental conditions, ranked by their probabilities, used to evaluate the model accuracy (3). 23

11 Classification accuracy (%) of two environmental classification models (ECCMv1 and ECCMv2) under various underwater image degradation conditions: biofilm, bubbles, low light, overexposure, and turbidity. Both ECCMv1 and ECCMv2 exhibit high and stable accuracy across all environmental conditions, with ECCMv1 based achieving the highest performance overall by a tiny margin, however as described in Table 1, ECCMv2 has a faster inference time. 24

12 Examples from recording locations with different weather conditions. (a) Clear, ideal conditions of the roadway and vehicles. (b) Cloudy conditions, where some regions of the roadway have poor illumination at a distance. (c) Rain and other non-ideal conditions in which the camera lens may have water droplets, and where sections of the roadway may have blurred imagery. (d) Partially cloudy, dynamic changes in near and far-field illumination occur on the roadway due to variations in cloud cover. (e) Night, considerable variability in the roadway illumination levels due to static street lighting in conjunction with automobile head and tail lights. 27

13 Examples of recording locations situated in tributaries of the Tamega River (Douro River basin), Portugal: (a) Bragado, located in the Avelames River. (b) Covas Do Barroso, located in Couto River [79]. 34

14 a) Binocular vision system setup consisting of an IR and an RGB camera. First, the proposed solution detects and locates fish from a single frame, using both the IR and RGB cameras. b) After detecting fish on both images, the algorithm determines the quality of the bounding box pairs by analyzing their geometric and spatial properties. c) The disparity is first computed and subsequently utilized to estimate depth, leveraging the focal length and baseline as scene parameters. Following the depth estimation for each point, the median value (e.g., 32 in this example) is selected from a neighborhood of depth estimates at each pixel to mitigate the influence of outliers and ensure robust depth measurement. d) The fish size is determined by calculating the Euclidean distance utilizing the coordinates of the bottom left and top right corners (x and y) for both bounding boxes, which are then averaged to estimate the fish's size (total body length). 37

List of Tables

1 Comparison of both environmental condition classification models, ECCMv1 and ECCMv2, across various evaluation metrics. As depicted in Fig. 11, both behave similarly when trying to classify various environmental conditions. However, ECCMv2 is efficient, on average taking only 12 ms instead of 19 ms. Inference time measurements were taken using the Nvidia Jetson Orin Developer Kit. 24

2 Comparison of open multi-modal infrastructure-based datasets for vehicle detection and classification. CIM provides the most extensive open dataset for camera and mmWave radar to date. 28

3 Overview of the hardware components used in the designed traffic monitoring system. 29

4 Summary of performance metrics across the vehicle perception pipeline. The table presents key detection and classification results for both subsystems of the sensor fusion architecture. The camera module (YOLOv7) was evaluated for multiclass vehicle detection accuracy, localization robustness (mAP), and embedded inference speed and power consumption. The radar classification module (KNN) was evaluated for binary classification accuracy across car and bus/truck classes, and benchmarked for high-speed, energy-efficient performance on embedded hardware (Jetson Orin Nano). Metrics are reported using the best-performing configurations selected during validation. 31

5 Comparison of open access fish datasets for computer vision tasks. In situ: on site, in vitro: in lab, ex vivo: on dead specimens. ObD: object detection, FiC: classification (fish/ no fish), SpC: species classification, SzE: size estimation, Seg: segmentation. The number of frames corresponds to the number of available images before augmentation [79]. 35

6 Overview of the hardware components used in the designed embedded vision system. 36

7 Summary of performance metrics across the pipeline. 38

References

[1] Bureau of Transportation Statistics (BTS). National Transportation Statistics. `https://www.bts.gov/content/number-us-aircraft-vehicles-vessels-and-other-conveyances`. [Accessed: 2022-08-18].

[2] European Automobile Manufacturers' Association ACEA. New passenger car registrations and annual GDP growth in the EU. `https://www.acea.auto/figure/world-new-motor-vehicle-registrations-in-units/`. [Accessed: 2023-10-03].

[3] European Automobile Manufacturers' Association ACEA. New passenger car registrations and annual GDP growth in the EU. `https://www.acea.auto/figure/new-passenger-car-registrations-and-annual-gdp-growth-in-the-eu/`. [Accessed: 2022-08-19].

[4] U.S. Energy Information Administration (EIA). Today in Energy. `https://www.eia.gov/todayinenergy/detail.php?id=50096`. [Accessed: 2022-08-18].

[5] U.S. Government Accountability Office (GAO). Precision Agriculture: Benefits and Challenges for Technology Adoption and Use. `https://www.gao.gov/products/gao-24-105962`, 2023. [Accessed 16-07-2024].

[6] K. Anantharajah, Z. Ge, C. McCool, S. Denman, C. Fookes, P. Corke, D. Tjondronegoro, and S. Sridharan. Local inter-session variability modelling for object classification. In *IEEE Winter Conference on Applications of Computer Vision*, pages 309–316, 2014.

[7] A. Apprill, Y. Girdhar, A. Mooney, C. Hansel, M. Long, Y. Liu, W. Zhang, J. Kapit, K. Hughen, J. Coogan, and A. Greene. Toward a new era of coral reef monitoring. *Environmental science & technology*, 57(13):5117–5124, 2023.

[8] E. Arkin, N. Yadikar, X. Xu, A. Aysa, and K. Ubul. A survey: object detection methods from cnn to transformer. *Multimedia Tools and Applications*, 82(14):21353–21383, 2022.

[9] J. Arts, W. Leendertse, and T. Tillema. *Road infrastructure: planning, impact and management*, pages 360–372. Elsevier, 2021.

[10] C. H. Bahnsen and T. B. Moeslund. Rain removal in traffic surveillance: Does it matter? *IEEE Transactions on Intelligent Transportation Systems*, 20(8):2802–2819, 2019.

[11] M. D. Bejarano, R. Jansson, and C. Nilsson. The effects of hydropeaking on riverine plants: a review. *Biological Reviews*, 93(1):658–673, 2018.

[12] Biotactic Inc. Bravo g3 fish monitoring system. `https://www.biotactic.com/bravo-fish-monitoring-systems`, 2025. [Accessed: 2025-05-27].

[13] N. J. Bipa, G. Stradiotti, M. Righetti, and G. R. Pisaturo. Impacts of hydropeaking: A systematic review. *Science of The Total Environment*, 912, 2024.

[14] I. Boavida, J. Santos, M. J. Costa, R. Leite, M. Portela, F. Godinho, P. Leitão, R. Mota, J. Tuhtan, and A. Pinheiro. The ecopeak4fish project: an integrated approach to support self-sustaining fish populations downstream hydropower plants. In *Proceedings of the 39th IAHR World Congress*, 2022.

[15] J. E. Brandl. Pasteur's quadrant: Basic science and technological innovation. *Journal of Policy Analysis and Management*, 17(4):734–736, 1998.

[16] A. C. Burton, E. Neilson, D. Moreira, A. Ladle, R. Steenweg, J. T. Fisher, E. Bayne, and S. Boutin. Review: Wildlife camera trapping: a review and recommendations for linking surveys to ecological processes. *Journal of Applied Ecology*, 52(3):675–685, 2015.

[17] S. Busch, C. Koetsier, J. Axmann, and C. Brenner. Lumpi: The leibniz university multi-perspective intersection dataset. In *2022 IEEE Intelligent Vehicles Symposium (IV)*, pages 1127–1134, 2022.

[18] Z. Cai and N. Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6154–6162, 2018.

[19] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. In *Computer Vision – ECCV 2020*, pages 213–229, 2020.

[20] G. Ceballos, P. R. Ehrlich, and R. Dirzo. Biological annihilation via the ongoing sixth mass extinction signaled by vertebrate population losses and declines. *Proceedings of the National Academy of Sciences*, 114(30):E6089–E6096, 2017.

[21] S. Y. Cheung, S. Coleri, B. Dundar, S. Ganesh, C.-W. Tan, and P. Varaiya. Traffic measurement and vehicle classification with single magnetic sensor. *Transportation Research Record*, 1917(1):173–181, 2005.

[22] W.-T. Chu, X.-Y. Zheng, and D.-S. Ding. Camera as weather sensor: Estimating weather information from single images. *Journal of Visual Communication and Image Representation*, 46:233–249, 2017.

[23] G. Coro and M. Bjerregaard Walsh. An intelligent and cost-effective remote underwater video device for fish size monitoring. *Ecological Informatics*, 63:101311, 2021.

[24] C. Creß, W. Zimmer, L. Strand, M. Fortkord, S. Dai, V. Lakshminarasimhan, and A. Knoll. A9-dataset: Multi-sensor infrastructure-based dataset for mobility research. In *2022 IEEE Intelligent Vehicles Symposium (IV)*, pages 965–970, 2022.

[25] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893, 2005.

[26] S. Deinet, K. Scott-Gatty, H. Rotton, W. Twardek, V. Marconi, L. McRae, L. Baumgartner, K. Brink, J. Claussen, S. Cooke, W. Darwall, B. Eriksson, C. Garcia de Leaniz, Z. Hogan, J. Royte, L. Silva, M. Thieme, D. Tickner, J. Waldman, H. Wanningen, O. Weyl, and A. Berkhuysen. *The Living Planet Index (LPI) for migratory freshwater fish: Technical Report*. World Fish Migration Foundation, 2020.

[27] S. Diaz, M. Bernard, Y. Bernard, G. Bieker, K. Lee, P. Mock, E. Mulholland, P. Ragon, F. Rodriguez, U. Tietge, and S. Wappelhorst. European vehicle market statistics. `https://theicct.org/publication/european-vehicle-market-statistics-2021-2022/`. [Accessed: 2022-08-18].

[28] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR 2021*, 2020.

[29] Y. Fang, S. Yang, X. Wang, Y. Li, C. Fang, Y. Shan, B. Feng, and W. Liu. Instances as queries. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6890–6899, 2021.

[30] M. J. Feio, R. M. Hughes, M. Callisto, S. J. Nichols, O. N. Odume, B. R. Quintella, M. Kuemmerlen, F. C. Aguiar, S. F. Almeida, P. Alonso-EguíaLis, F. O. Arimoro, F. J. Dyer, J. S. Harding, S. Jang, P. R. Kaufmann, S. Lee, J. Li, D. R. Macedo, A. Mendes, N. Mercado-Silva, W. Monk, K. Nakamura, G. G. Ndiritu, R. Ogden, M. Peat, T. B. Reynoldson, B. Rios-Touma, P. Segurado, and A. G. Yates. The biological assessment and rehabilitation of the world's rivers: An overview. *Water*, 13(3), 2021.

[31] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[32] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester. Cascade object detection with deformable part models. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2241–2248, 2010.

[33] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[34] R. B. Fisher, Y.-H. Chen-Burger, D. Giordano, L. Hardman, F.-P. Lin, et al. *Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data*. Springer Cham, 2016.

[35] R. Froese, J. T. Thorson, and R. B. Reyes Jr. A bayesian approach for estimating length-weight relationships in fishes. *Journal of Applied Ichthyology*, 30(1):78–85, 2014.

[36] J. F. Fuentes-Pérez, A. García-Vega, F. J. Bravo-Córdoba, and F. J. Sanz-Ronda. A step to smart fishways: An autonomous obstruction detection system using hydraulic modeling and sensor networks. *Sensors*, 21(20), 2021.

[37] R. Girshick. Fast r-cnn. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015.

[38] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014.

[39] A. Greene, Z. Forsman, R. J. Toonen, and M. J. Donahue. Coralcam: A flexible, low-cost ecological monitoring platform. *HardwareX*, 7, 2020.

[40] C. Haas, P. Thumser, B. Mockenhaupt, and M. Schletterer. The system vaki riverwatcher as a tool for long-term monitoring of fish migration in fishways. *WASSERWIRTSCHAFT*, 108(9):41–48, 2018.

[41] S. Hamel, S. T. Killengreen, J.-A. Henden, N. E. Eide, L. Roed-Eriksen, R. A. Ims, and N. G. Yoccoz. Towards good practice guidance in using camera-traps in ecology: influence of sampling design on validity of ecological inferences. *Methods in Ecology and Evolution*, 4(2):105–113, 2013.

[42] F. He, C. Zarfl, K. Tockner, J. D. Olden, Z. Campos, F. Muniz, J. Svenning, and S. C. Jähnig. Hydropower impacts on riverine biodiversity. *Nature Reviews Earth & Environment*, 5(11):755–772, 2024.

[43] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Computer Vision – ECCV 2014*, pages 346–361, 2014.

[44] J. Hu, D. Zhao, Y. Zhang, C. Zhou, and W. Chen. Real-time nondestructive fish behavior detecting in mixed polyculture system using deep-learning and low-cost devices. *Expert Systems with Applications*, 178:115051, 2021.

[45] M. R. Ibrahim, J. Haworth, and T. Cheng. Weathernet: Recognising weather and visual conditions from street-level images using deep residual learning. *ISPRS International Journal of Geo-Information*, 8(12), 2019.

[46] Intel Corporation. Intel neural compute stick 2, 2018. [Accessed: 2025-02-09].

[47] F. Jin, A. Sengupta, S. Cao, and Y.-J. Wu. Mmwave radar point cloud segmentation using gmm in multimodal traffic monitoring. In *2020 IEEE International Radar Conference (RADAR)*, pages 732–737, 2020.

[48] G. Jocher and J. Qiu. Ultralytics yolo11. `https://github.com/ultralytics/ultralytics/`. [Accessed: 2025-08-19].

[49] N. P. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, N. Boden, A. Borchers, R. Boyle, P.-l. Cantin, C. Chao, C. Clark, J. Coriell, M. Daley, M. Dau, J. Dean, B. Gelb, T. V. Ghaemmaghami, R. Gottipati, W. Gulland, R. Hagmann, C. R. Ho, D. Hogberg, J. Hu, R. Hundt, D. Hurt, J. Ibarz, A. Jaffey, A. Jaworski, A. Kaplan, H. Khaitan, D. Killebrew, A. Koch, N. Kumar, S. Lacy, J. Laudon, J. Law, D. Le, C. Leary, Z. Liu, K. Lucke, A. Lundin, G. MacKean, A. Maggiore, M. Mahony, K. Miller, R. Nagarajan, R. Narayanaswami, R. Ni, K. Nix, T. Norrie, M. Omernick, N. Penukonda, A. Phelps, J. Ross, M. Ross, A. Salek, E. Samadiani, C. Severn, G. Sizikov, M. Snelham, J. Souter, D. Steinberg, A. Swing, M. Tan, G. Thorson, B. Tian, H. Toma, E. Tuttle, V. Vasudevan, R. Walter, W. Wang, E. Wilcox, and D. H. Yoon. In-datacenter performance analysis of a tensor processing unit. In *Proceedings of the 44th Annual International Symposium on Computer Architecture*, page 1–12, 2017.

[50] Y.-K. Ki and D.-K. Baik. Vehicle-classification algorithm for single-loop detectors using neural networks. *IEEE Transactions on Vehicular Technology*, 55(6):1704–1711, 2006.

[51] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90, 2017.

[52] J. J. Lahoz-Monfort and M. J. L. Magrath. A comprehensive overview of technologies for species and habitat monitoring and conservation. *BioScience*, 71(10):1038–1062, 2021.

[53] M. Larrea-Gomez, A. Peña, J. D. Martinez-Vargas, I. Ochoa, and T. Ramirez-Guerrero. Modeling detecting plant diseases in precision agriculture: A NDVI analysis for early and accurate diagnosis. In *Advances in Computing*, pages 297–310, 2024.

[54] R. J. Lennox, C. P. Paukert, K. Aarestrup, M. Auger-Méthé, L. Baumgartner, K. Birnie-Gauvin, K. Bøe, K. Brink, J. W. Brownscombe, Y. Chen, J. G. Davidsen, E. J. Eliason, A. Filous, B. M. Gillanders, I. P. Helland, A. Z. Horodysky, S. R. Januchowski-Hartley, S. K. Lowerre-Barbieri, M. C. Lucas, E. G. Martins, K. J. Murchie, P. S. Pompeu, M. Power, R. Raghavan, F. J. Rahel, D. Secor, J. D. Thiem, E. B. Thorstad, H. Ueda, F. G. Whoriskey, and S. J. Cooke. One hundred pressing questions on the future of global fish migration science, conservation, and policy. *Frontiers in Ecology and Evolution*, 7, 2019.

[55] D. Li, Y. Hao, and Y. Duan. Nonintrusive methods for biomass estimation in aquaculture with emphasis on fish: a review. *Reviews in Aquaculture*, 12(3):1390–1411, 2020.

[56] Y. Li, Y. Chen, N. Wang, and Z.-X. Zhang. Scale-aware trident networks for object detection. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6053–6062, 2019.

[57] Y. Li, N. Miao, L. Ma, F. Shuang, and X. Huang. Transformer for object detection: Review and benchmark. *Engineering Applications of Artificial Intelligence*, 126, 2023.

[58] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, 2017.

[59] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *Computer Vision – ECCV 2016*, pages 21–37, 2016.

[60] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021.

[61] V. A. S. Ltd. Vaki riverwatcher. `https://www.riverwatcher.is`, 2025. [Accessed: 2025-05-27].

[62] D. Mallet and D. Pelletier. Underwater video techniques for observing coastal marine biodiversity: A review of sixty years of publications (1952–2012). *Fisheries Research*, 154:44–62, 2014.

[63] S. Meta and M. G. Cinsdikici. Vehicle-classification algorithm based on component analysis for single-loop inductive detector. *IEEE Transactions on Vehicular Technology*, 59(6):2795–2805, 2010.

[64] M. Mills, M. Ungermann, G. Rigot, J. Haan, J. Leon, and T. Schils. Assessment of the utility of underwater hyperspectral imaging for surveying and monitoring coral reef ecosystems. *Scientific Reports*, 13(1), 2023.

[65] NVIDIA Corporation. Nvidia tensor cores. *NVIDIA Developer Blog*, 2018. [Accessed: 2025-02-09].

[66] H. A. Oliveira, F. R. Barbosa, O. M. Almeida, and A. P. S. Braga. A vehicle classification based on inductive loop detectors using artificial neural networks. In *2010 9th IEEE/IAS International Conference on Industry Applications - INDUSCON 2010*, pages 1–6, 2010.

[67] H. Ouyang. Deyo: Detr with yolo for end-to-end object detection. *ArXiv*, 2024.

[68] J. Pang, K. Chen, J. Shi, H. Feng, W. Ouyang, and D. Lin. Libra r-cnn: Towards balanced learning for object detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 821–830, 2019.

[69] M. Pedersen, J. Haurum, R. Gade, T. Moeslund, and N. Madsen. Detection of marine animals in a new underwater dataset with varying visibility. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019.

[70] K. L. Pope, S. E. Lochmann, and M. K. Young. Methods for assessing fish populations. *Inland Fisheries Management in North America*, pages 325–351, 2010.

[71] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016.

[72] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017.

[73] I. Robinson, P. Robicheaux, and M. Popov. Rf-detr. `https://github.com/roboflow/rf-detr`, 2025. SOTA Real-Time Object Detection Model.

[74] A. Saleh, I. H. Laradji, D. A. Konovalov, M. Bradley, D. Vazquez, and M. Sheaves. A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis. *Scientific Reports*, 10(1):14671, 2020.

[75] S. Schmutz and J. Sendzimir. *Riverine Ecosystem Management: Science for Governing Towards a Sustainable Future*. Springer International Publishing, 2018.

[76] H. J. Shatz, K. E. Kitchens, S. Rosenbloom, and M. Wachs. *Highway Infrastructure and the Economy: Implications for Federal Policy*. RAND Corporation, 2011.

[77] T. Shehzadi, K. A. Hashmi, D. Stricker, and M. Z. Afzal. Object detection with transformers: A review. *ArXiv*, 2023.

[78] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[79] J. Soom, I. Boavida, R. Leite, M. J. Costa, G. Toming, M. Leier, and J. A. Tuhtan. Open real-time, non-invasive fish detection and size estimation utilizing binocular camera system in a portuguese river affected by hydropeaking. *Ecological Informatics*, 90, 2025.

[80] J. Soom, M. Leier, K. Janson, and J. A. Tuhtan. Open urban mmwave radar and camera vehicle classification dataset for traffic monitoring. *IEEE Access*, 12:65128–65140, 2024.

[81] J. Soom, V. Pattanaik, M. Leier, and J. A. Tuhtan. Environmentally adaptive fish or no-fish classification for river video fish counters using high-performance desktop and embedded hardware. *Ecological Informatics*, 72, 2022.

[82] P. Sun, R. Zhang, Y. Jiang, T. Kong, C. Xu, W. Zhan, M. Tomizuka, L. Li, Z. Yuan, C. Wang, and P. Luo. Sparse r-cnn: End-to-end object detection with learnable proposals. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14449–14458, 2021.

[83] M. Tan, R. Pang, and Q. V. Le. Efficientdet: Scalable and efficient object detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10778–10787, 2020.

[84] D. Tickner, J. J. Opperman, R. Abell, M. Acreman, A. H. Arthington, S. E. Bunn, S. J. Cooke, J. Dalton, W. Darwall, G. Edwards, I. Harrison, K. Hughes, T. Jones, D. Leclère, A. J. Lynch, P. Leonard, M. E. McClain, D. Muruven, J. D. Olden, S. J. Ormerod, J. Robinson, R. E. Tharme, M. Thieme, K. Tockner, M. Wright, and L. Young. Bending the Curve of Global Freshwater Biodiversity Loss: An Emergency Recovery Plan. *BioScience*, 70(4):330–342, 2020.

[85] R. Varghese and S. M. Yolov8: A novel object detection algorithm with enhanced performance and robustness. In *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)*, pages 1–6, 2024.

[86] J. Venema. Affine - angling freshwater fish netherlands. `https://www.kaggle.com/datasets/jorritvenema/affine`, 2021.

[87] P. Viola and M. Jones. Robust real-time face detection. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 747–747, 2001.

[88] A. Wang, H. Chen, and L. Liu. Yolov10: Real-time end-to-end object detection. *ArXiv*, 2024.

[89] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, 2022.

[90] H. Wang, X. Zhang, Z. Li, J. Li, K. Wang, Z. Lei, and R. Haibing. Ips300+: a challenging multi-modal data sets for intersection perception system. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2539–2545, 2022.

[91] J. Wang, T. Fu, J. Xue, C. Li, H. Song, W. Xu, and Q. Shangguan. Realtime wide-area vehicle trajectory tracking using millimeter-wave radar sensors and the open tjrd ts dataset. *International Journal of Transportation Science and Technology*, 12(1):273–290, 2023.

[92] Y. Wang, X. Zhang, J. Chen, Q. Zhang, H. Sun, L. Zhang, and X. Wang. Camera sensor-based contamination detection for water environment monitoring. *Environmental Science and Pollution Research*, 26(3):2722–2733, 2019.

[93] M. Won. Intelligent traffic monitoring systems for vehicle classification: A survey. *IEEE Access*, 8:73340–73358, 2020.

[94] K. Xie, Z. Wei, L. Huang, Q. Qin, and W. Zhang. Graph convolutional networks with attention for multi-label weather recognition. *Neural Computing and Applications*, 33(17):11107–11123, 2021.

[95] B. Yang and Y. Lei. Vehicle detection and classification for low-speed congested traffic with anisotropic magnetoresistive sensor. *IEEE Sensors Journal*, 15(2):1132–1138, 2015.

[96] B. Yang, H. Zhang, Y. Chen, Y. Zhou, and Y. Peng. Urban traffic imaging using millimeter-wave radar. *Remote Sensing*, 14(21), 2022.

[97] M. Yang, X. Han, X. Ping, Z. Li, and J. Xiao. A clearer image: Improving object detection in real rainy conditions with two-stage processing. In *2023 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, pages 57–62, 2023.

[98] X. Yang, M. B. Mi, Y. Yuan, X. Wang, and R. T. Tan. Object detection in foggy scenes by embedding depth and reconstruction into domain adaptation. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 1093–1108, 2022.

[99] L. Yanmar Co. Yanmar marine systems. `https://www.yanmar.com/marine`, 2025. [Accessed: 2025-05-27].

[100] X. Ye, M. Shu, H. Li, Y. Shi, Y. Li, G. Wang, X. Tan, and E. Ding. Rope3d: The roadside perception dataset for autonomous driving and monocular 3d object detection task. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21309–21318, 2022.

[101] H. Yu, Y. Luo, M. Shu, Y. Huo, Z. Yang, Y. Shi, Z. Guo, H. Li, X. Hu, J. Yuan, and Z. Nie. Dair-v2x: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21329–21338, 2022.

[102] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection, 2022.

[103] Y. Zhang, A. Carballo, H. Yang, and K. Takeda. Perception and sensing for autonomous vehicles under adverse weather conditions: A survey. *ISPRS Journal of Photogrammetry and Remote Sensing*, 196:146–177, 2023.

[104] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen. Detrs beat yolos on real-time object detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16965–16974, 2024.

[105] H. Zhou, M. Kong, H. Yuan, Y. Pan, X. Wang, R. Chen, W. Lu, R. Wang, and Q. Yang. Real-time underwater object detection technology for complex underwater environments based on deep learning. *Ecological Informatics*, 82, 2024.

Acknowledgements

There are not enough words to encapsulate this remarkable journey. It has been both challenging and rewarding in numerous ways. This experience transcends mere titles or prestige; it fundamentally shapes your character. Reflecting on these past five years, it’s hard to believe that you once joked about pursuing a PhD simply to add **Dr** to your business card. Yet here I stand, at the final stages of actually completing it. First and foremost, I want to mentioned **Indrek Lambut** for being a inspiration to study IT in the first place. Without you, none of this would have not happened. Wherever you are now, know that I will be always grateful to you. I would like to express my heartfelt appreciation to my supervisors, **Assoc. Prof. Jeffrey A. Tuhtan** and **Dr. Mairo Leier**, for providing me with this incredible opportunity. Their guidance, unwavering support, and constructive feedback during both the good and challenging times have been invaluable. I truly would not have come this far without them.

I would like to express my heartfelt gratitude to my colleagues from the **Centre for Biorobotics and Centre for Environmental Intelligence and Sensing**, including **Gert Toming**, **Jaanus Joasoo**, **Maarja Kruusmaa**, **Asko Ristolainen**, **Jaan Rebane**, **Roza Gkliva**, **Helena Carmen Udu**, **Vishwajeet Pattanaik**, and everyone else. I extend my thanks to **Uljana Reinsalu**, **Karl Janson**, **Jüri Bogatkin**, **Can Ersü**, **Karel Meriste**, **Elvar Liiv**, **Oskar Voorel** and **Priit Ruberg** from the **Embedded AI Research Lab**, as well as **Hardi Selg**, **Risto Heinsaar**, **Katrin Tõemets**, **Margus Kruus**, **Karin Härmat**, **Gert Jervan**, **Vladimir Viies**, **Liisi Ilu**, **Tarmo Robal**, **Peeter Ellervee** and many more from the **Department of Computer Systems** for their invaluable camaraderie and support. On a personal level, I’m grateful to my family for their endless love, patience, and encouragement. My parents, **Kalli** and **Peeter**, along with my brother **Oliver**, have always believed in my dreams and pushed me to keep going. My grandparents **Sirje** and **Villu Soom** as well as **Hans** and **Tiia Kruus** and aunt **Kaie Kruus**. I would like to extend my heartfelt thanks to the people from the research team: **Laura Toodu**, **Martin Lints**, **Jari Lauri Allan Laapas**, **Lauri Vihman**, **Jaak Joonas Uudmäe**, **Kirill Amelin**, **Jörg Miikael Tiit**, **Anna Mandrenko**, **Mohamed Walid Remmas**, **Margit Egerer**, **Uku Kert Paidra** , and office furnado **Ruudi**. Additionally, I would like to express my gratitude to everyone else, with special thanks to: **Hedvi Kink**, **Mariliis Martin**, **Johanna Kruusement**, **Creiton Kojalo**, **Maili Suurekivi**, **Margot Vahtra**, and **Polina Lutsevitsh** and others for the support, encouragements and taking time out of your busy schedule to listen and give feedback about presentations and asking me to bake more cakes and pies. I would like to express my heartfelt gratitude to **Raul Leemet**, **Henry Juhanson**, **Reijo Olavi Komu**, **Karl Laanemets**, **Rait Kurg**, **Piret Kurg**, **Markus Põldmäe**, **Karl Sobak**, **Allar Nõmm**, **Madis Vahtrik**, **Ergo Siht** and everyone else with whom I had the pleasure of studying with. Thank you all for making my journey through both my bachelor’s and master’s degrees so much smoother, more enjoyable and passing few exams easier ;-). Your support and camaraderie have made all the difference. I want to express my heartfelt gratitude to everyone I’ve had the privilege of learning from and spending my Thursday evenings with at Toastmasters. A special thank you to the following individuals: **Sara Sinha**, **Thai Nguyen**, **Dagnija Huberg**,**Ondrej Kokoska**, **Jacek Lazorik**, **Annika Nilson**, **Oskar Andre Oja**, **Patricia Nilson**, **Rasmus Leichter**, **Katrina Van Der Valk**, **Kris Konsap**, **Marleen Leemets**, **Kat Kalda**, **Harri Parker**, and **Joosep Sõnajalg**. Your support and insights have made this experience truly enriching. Special thanks to **Jan Toodre**, **Ekke Tõiv Uustalu**, and **Joonas Tamm**. I am truly grateful for their guidance and support, as they played a very crucial step, which in some way started my academic career. Assisting with programming labs was not only fun but also full of memorable and hilarious events. Their encouragement and mentorship have made this journey exciting and unforgettable. I would like extend my gratitude to

Margus Einpalu and **Lea Einpalu** for all the help, support, encouragement. Also I would like to acknowledge all of the students, colleagues, and friends, that I had the pleasure to meet and work with throughout this journey. Also, I would like extend my gratitude to **Laine** for all the help, support, encouragement, and wisdom. I am deeply grateful for the help and support I have received.

Finally, I want to express my heartfelt gratitude to the funding agencies, institutions, and organizations that have supported my research and studies, particularly the **Tallinn University of Technology**, **IT Academy** and the **Estonian Research Council**. Their generous financial backing and invaluable resources have been instrumental in making this work a reality. I am deeply thankful to everyone who has contributed to my academic and personal journey, both in significant and subtle ways. This dissertation stands as a testament to the belief and support I have received—without which, this achievement would not have been possible. There are so many names to list but I think they will not let me keep going, otherwise the list will be as long as the overall dissertation. If I’ve forgotten some, don’t worry, I’m still grateful for all the help and support I’ve received throughout the years.

Aitäh teile kõigile!

Abstract

Reliable object recognition and assessment in adverse weather and environmental conditions

This doctoral thesis aims to design and develop an adaptive processing pipeline to enhance object detection and assessment performance under adverse weather and environmental conditions. An initial systematic literature review identified three main research gaps, which guided the direction of subsequent publications. The primary contributions of this thesis involve integrating machine learning models with data-driven methods. The developed solution is computationally lightweight, making it suitable for low-power, resource-constrained embedded hardware.

The work also emphasizes the importance of open datasets, as their absence or limited availability hinders the development, testing, and validation of new solutions. During this research, two datasets were created: one focused on detecting and assessing freshwater fish, and the other aimed at identifying and classifying various types of vehicles in urban environments.

Compared to existing methods, this doctoral thesis offers a universal solution applicable to terrestrial and underwater environments. The results offer a deeper understanding of how weather and various environmental conditions impact system performance, paving the way for innovative and novel approaches in future scientific research.

Kokkuvõte

Usaldusväärne objektide tuvastamine ja hindamine ebasoodsates ilmastiku- ja keskkonnatingimustes

Käesoleva doktoritöö eesmärgiks on disainida ja luua adaptiivne töötlusahel, et parandada objektituvastuse ja objektide hindamise sooritusvõimet ebasoodsates ilmastiku- ja keskkonnatingimustes. Esialgse süstemaatilise kirjanduse ülevaate tagajärjel tuvastati kolm peamist uurimislünka, mille põhjal tuletatud küsimused juhtisid avaldatud publikatsioonide suunda. Peamised doktoritöö panused hõlmavad masinõppemudelite sidumist andmepõhiste meetoditega. Loodud lahendus on arvutuslikult kerge, sobides madala voolutarbega piiratud sardriistvarale.

Töö rõhutab ka avatud andmestike olulisust, mille puudumine või kättesaadavus takistab uute lahenduste loomist, testimist ja valideerimist. Antud uurimustöö käigus loodi kaks andmekogu, millest üks oli suunatud mageveekalade tuvastamiseks ja hindamiseks. Teine andmestik oli suunatud erinevate sõidukitüüpide tuvastamiseks.

Võrreldes olemasolevate meetoditega, pakub käesolev doktoritöö universaalset lahendust, mida saab rakendada nii maismaal kui ka veealustes keskkondades. Töö tulemused aitavad põhjalikumalt aru saada, kuidas ilmastik ja erinevad keskkonnatingimused mõjutavad süsteemi sooritusvõimet, sillutades teed uutele meetoditele tulevastes teadusuuringutes.

Appendix 1

I

J. Soom, V. Pattanaik, M. Leier, and J. A. Tuhtan. Environmentally adaptive fish or no-fish classification for river video fish counters using high-performance desktop and embedded hardware. *Ecological Informatics*, 72, 2022

Ecological Informatics 72 (2022) 101817

Contents lists available at ScienceDirect

Ecological Informatics

journal homepage: www.elsevier.com/locate/ecolinf

Environmentally adaptive fish or no-fish classification for river video fish counters using high-performance desktop and embedded hardware

Jürgen Soom [a,b,*], Vishwajeet Pattanaik [b], Mairo Leier [a], Jeffrey A. Tuhtan [b]

[a] Embedded AI Research Lab, Tallinn University of Technology, Akadeemia tee 15A, Tallinn 12618, Harjumaa, Estonia
[b] Department of Computer Systems, Tallinn University of Technology, Akadeemia tee 15A, Tallinn 12618, Harjumaa, Estonia

ARTICLE INFO

Keywords:
Fish detection
Deep learning
Underwater video
Environmental classification
Embedded hardware

ABSTRACT

Automated fish counters featuring robust, real-time computer vision capabilities can provide a cost-effective means to count migrating freshwater fish. In this work, we propose a four-stage process for automatically sorting videos with and without fish. Underwater fish counter videos provide a challenging range of environmental conditions including clear water, biofilm growth, bubbles, turbidity, low light and overexposure. To address this, our method also includes the automated classification of these six environmental conditions. The proposed methods are computationally efficient and can be implemented on servers, high-performance desktop computers and low-cost, energy-efficient embedded hardware. The models were trained, tested, and validated using a collection of 3000 videos taken from underwater fish counter installations in several alpine and lowland European rivers provided by commercial and governmental collaborators. This work demonstrates a fast, accurate, and robust computer vision workflow for large-scale automated freshwater fish counting systems.

1. Introduction

Freshwater ecosystems host one-third of all vertebrate species and are experiencing a rapid decline (Ceballos et al., 2017). Global wetland destruction is occurring at a pace three times faster than that of forests, and the compounding impacts of climatic and anthropogenic changes are reducing freshwater vertebrate populations at more than twice the rate of terrestrial or marine populations (Feio et al., 2021; Tickner et al., 2020). The increase of automated digital monitoring technologies, environmental genomics and citizen science can assist in establishing robust and widespread concepts for freshwater biodiversity monitoring (Dwivedi, 2021).

Where suitable physical conditions prevail, underwater camera-based fish counters can be used to assess fish migration through fish passage structures, both up and downstream (Haas et al., 2018; Mallet and Pelletier, 2014). Considering freshwater fish species, a wider and more accurate representation of their daily migration activities and counts are required to study, understand, predict, and support sustainable freshwater fisheries (Deinet et al., 2020; Lennox et al., 2019). Fish counters can also provide key data to fulfill reporting requirements by the European Union's Habitat and Water Framework Directives. As hardware costs decrease and the quality of contemporary low-light infrared (IR) and color imaging (RGB) systems increases, the use of underwater camera-based fish counters is expected to grow substantially (Fjeldstad et al., 2018). Commensurate with the growth in the number of these systems, computer vision algorithms and hardware now allow for near real-time fish detection and species classification (Fabic et al., 2013; Sharma et al., 2016). Despite these promising advances, underwater video quality varies considerably, largely due to changes in environmental conditions caused by biofouling, irregular lightning, turbidity and debris. The site-specific conditions of each installed camera can vary widely, posing a persistent challenge for human evaluators to consistently and efficiently sort videos with and without fish.

Commercially available systems include, but are not limited to, the River Watcher (Vaki, Iceland), Bravo G3 (Biotactic, Canada) and the Yanmar Marine System (Yanmar, Japan). These systems can be intergrated with water quality, flow and water level sensors to create smart fishways (Fuentes-Pérez et al., 2021). In addition, there is a growing potential in the application of temporary, low-power camera fish counters which use embedded hardware and less complex algorithms than their PC or server-based counterparts (Hu et al., 2021; Li et al., 2020).

In contrast to previous works, our proposed method provides an environmentally-adaptive multi-stage computer vision pipeline. It is

---

* Corresponding author at: Embedded AI Research Lab, Tallinn University of Technology, Akadeemia tee 15A, Tallinn 12618, Harjumaa, Estonia.
E-mail address: jurgen.soom@taltech.ee (J. Soom).

https://doi.org/10.1016/j.ecoinf.2022.101817
Received 31 July 2022; Received in revised form 13 September 2022; Accepted 13 September 2022
Available online 19 September 2022
1574-9541/© 2022 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

environmentally-adaptive because in the first processing stage, six commonly occurring environmental conditions are classified. The main novelty of this work is the application of the six environmental classes to automatically adjust the fish or no-fish binary classification model hyperparameters for each video. This is significant advancement because it opens up new opportunities to use simpler, adaptive and more generalized methods for the binary classification step. Specifically, our proposed method separates the environmental conditions from the fish and no-fish binary classification task. This separation reduces the complexity of testing and implementation, and allows for more explainable machine learning outcomes, because the environmental conditions and their severity can be included as physically interpretable hyperparameters for fish or no-fish binary classification models. Once videos are classified based on the presence or absence of fish, they can be processed in a final step by computer vision methods for the automated classification of fish species (Mader et al., 2020).

1.1. Previous work

Underwater camera-based fish counting systems must accomplish two main tasks. The main objective of this work is to address the first task:

- Binary classification of videos with or without fish under changing environmental conditions.
- Individual fish classification, where taxonomic labels are assigned to the detected fish, corresponding to their family, genus or species.

The pioneering work of Strachan (1993) used computer vision to classify 23 marine fish species, based on their color and geometric descriptors from two separate video sources. Harvey and Shortis (1995) proposed a method in which the fish passed through a controlled illumination chamber equipped with stereo cameras. While the method performed quite well in an artificial environment, it exhibited a significant reduction in performance when applied to unconstrained environments (Zhao et al., 2021). Primary challenges in fish passage facilities are adapting computer vision methods to naturally occurring and constantly changing environmental conditions, and the correct discrimination of fish from non-fish objects, most frequently leaves, debris and bubbles. Promising, recent advancements in machine learning have shown that image-based fish detection accuracy in unconstrained environments can exceed 98.0% (Zhang et al., 2020b). However, due to computational requirements, most computer vision processing and analysis is performed on a server or high performance desktop (HPD) hardware. The underlying computational complexity of the machine learning (ML) methods used in automated fish counters therefore hinders their ubiquitous, real-time application. Solutions using low-cost and low-powered embedded hardware remain sparse among the research community largely due to the technical difficulties of their implementation in the field (Hernández-Ontiveros et al., 2018; Zhang et al., 2020a). A major drawback of using embedded hardware for computer vision applications is the limited computational power available when compared to high-performance desktops or servers.

1.2. Objectives

The primary objective of this work was to create a computationally and energy efficient computer vision pipeline to robustly and automatically classify videos with and without fish. In addition, we also evaluated the suitability of the proposed approach as an edge computer vision system using three commercially-available low-cost embedded hardware devices. The main contributions of this work are three-fold:

- Classification of six commonly occurring environmental conditions (clear, low light, air bubbles, turbidity, periphytic biofilm, and light overexposure) occurring at freshwater underwater camera installations.
- Development of lightweight computer vision algorithms suitable for embedded hardware to classify videos with and without fish.
- Comparison of the proposed approaches on high-performance desktops and low-cost embedded hardware considering the frame rate, hardware costs, and power consumption.

2. Materials and methods

2.1. Embedded hardware

Due to the limited availability of embedded hardware, we restricted our choices to a subset of three feasible systems. It is important that the data processing pipeline was run on the CPU only. This was a necessary step in evaluating the feasibility of running the environmentally-adaptive video classification system on embedded hardware to establish their benchmark performance. The first choice was the Raspberry Pi 4. The board features a Quad-core Cortex-A72 and Broadcom VideoCore VI based dedicated GPU. The board used in testing was the 4 GB version. With support from a large international user community, the Raspberry Pi has the additional benefits of extensive documentation and troubleshooting to assist during development and deployment.

The next choice was the Nvidia Jetson Nano, which features a Quad-core Cortex-A57 microprocessor and Maxwell architecture-based GPU. Compared with the Raspberry Pi 4, the microprocessor uses an older architecture with slightly lower core clocks. Similar to the Raspberry Pi 4, the board does not have built-in FLASH memory, but is supported by comprehensive development documentation.

The final choice was the MediaTek Pumpkin i500, which includes an octa-core microprocessor, which is the combination of A73 and A53-based microprocessors with core clocks up to 2.0 GHz. In addition, the board features a Mali-G72 MP3 GPU and Dual-core Tensilica Vision P6 DSP/AI accelerator. The board also features 16 GB of built-in FLASH memory compared to previous selections. Full hardware specifications of the embedded hardware tested in this work, including preliminary performance metrics are provided in Section 3.3.

2.2. Balanced video selection and annotation

The 3000 videos used in this work have been collected from seven European River Watcher monitoring sites, primarily from lowland, midland and alpine rivers. Table 1 describes the river types, fish species and range of sizes. The testing and validation approach used in this work was inspired by the Fish4Knowledge project, which collected benchmark imagery for the marine environment (Boom et al., 2014; Salman et al., 2019). Specific features of the videos used in this work are:

- A balanced dataset including six different environmental conditions: clear, low lighting, air bubbles, biofilm growth, turbidity and overexposure.
- Changing lighting conditions including natural and artificial lighting and darkness, throughout the year. The majority of videos were recorded in color. However, videos also include grayscale low-light imagery.
- The videos include 17 different fish species. This is key for the testing and validation of motion detection, as each species has semi-unique morphological and behavioural characteristics while migrating through the fishway.

For each environmental condition the dataset contained 250 videos with fish and 250 videos without fish; giving a total of 3000 videos (= 6 environmental conditions * {250 fish videos +250 no-fish videos}). Fig. 1 illustrates representative environmental conditions for each of the six classes. A summary of the properties of the videos used in this work are elucidated in Table 2.

Table 1
Overview of Vaki River Watcher sites including the identifier, river type, fish species (17) and range of total body lengths (min 0.05 m, max 1.0 m) observed in the 3000 videos used for training, testing and validation. Commercial sites (RW_03 to RW_07) are anonymous locations. Fish smaller than 0.05 m were not clearly identifiable, and represent the lower size limit considered in the range of body lengths present in the seven sites evaluated. Latin names are included for each fish species after their first listing in the table, afterwards only the English name is used.

Site ID	River Type	Fish Species Observed at this Site	Range of Sizes of Fish Total Body Length (m)	Comments
Mosel, Koblenz, Germany	Lowland	Asp (*Aspius aspius*), Barbel (*Barbus barbus*), Common bream (*Abramis brama*), Brown trout (*Salmo trutta*), Chub (Squalius cephalus), Common bleak (*Alburnus alburnus*), European carp (*Cyprinus carpio*), European eel (*Anguilla anguilla*), European grayling (*Thymallus thymallus*), Gudgeon (*Gobio gobio*), Nase (*Chondrostoma nasus*), European perch (*Perca fluviatilis*), Rainbow trout (*Oncorhynchus mykiss*), Roach (*Rutilus rutilus*), White-eyed bream (*Ballerus sapa*)	0.05 to 1.0	Many videos included groups of 5 or more fish, mixed species. High biofilm common, wide range of environmental conditions present at this site.
Müritz-Elde-Wasserstrasse, Malliß, Germany	Lowland	Barbel, Brown trout, Chub, Goby (family Gobiidae), Grayling, Nase, Perch, Roach	0.05 to 0.80	Some videos included groups of 5 or more fish, typically small cyprinids. Broad mixture of environmental conditions.
RW_03	Lowland	Barbel, Brown trout, Chub, Grayling, Nase, Perch and Atlantic salmon (*Salmo salar*)	0.05 to 0.65	Mostly videos of individual fish. Biofilm, bubbles and turbidity commonly found.
RW_04	Midland	Barbel, Brown trout, Chub, Nase and Perch	0.05 to 0.55	Mostly videos of individual fish. Biofilm and bubbles present in some videos.
RW_05	Alpine	Atlantic salmon, Brown trout, Rainbow trout	0.25 to 1.0	Mostly videos of individual fish. High turbidity and bubbles commonly found.
RW_06	Alpine	Atlantic salmon, Brown trout, Rainbow trout	0.35 to 1.0	Mostly videos of individual fish. High turbidity and bubbles commonly found.
RW_07	Glacial Alpine	Brown trout, Burbot (*Lota lota*), Grayling	0.20 to 0.65	No groups of fish. Low light and high turbidity common.

2.3. *Four-stage video classification pipeline*

To classify videos based on the presence or absence of fish, we propose an underwater video classification pipeline which relies on two different approaches. The pipeline aims to be adaptive against different environments and computationally lightweight while remaining portable. In this work, portability is defined as the ability to deploy the pipeline with minimal changes to the code on high-performance desktop hardware or low-powered edge computing hardware. Each method will be discussed separately and compared regarding classification accuracy, computational efficiency, and portability. The four stages of the video classification pipeline are depicted in Fig. 2:

- Environmental classification - The workflow begins by importing the videos from collected River Watcher monitoring sites and identifying environmental conditions. The output file from this stage contains three possible environmental condition labels for each video with their respective probabilities. Classification is used to determine the method and settings for image enhancement in the preprocessing phase.
- Preprocessing - The extracted frame is enhanced according to the detected environmental conditions. Enhancement is necessary to improve the fish / no-fish detection accuracy, especially in low light and turbid conditions.
- Processing - Each frame is classified using both frame differencing and scanlines.
- Binary classification - The frame-wise classifications are aggregated and the final classification outcome for the video is determined.

2.4. *Environmental conditions*

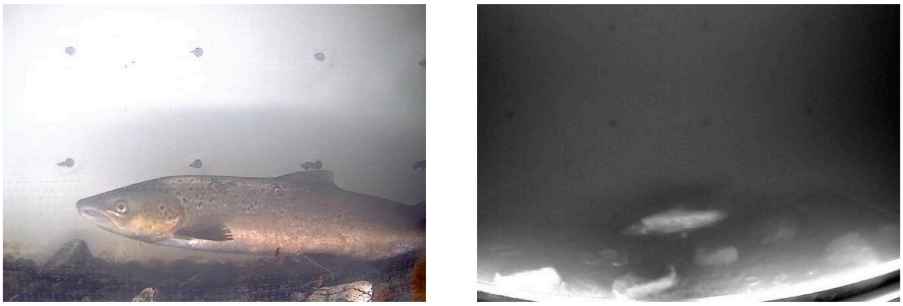
The first step of the proposed fish / no-fish video processing pipeline is environmental condition classification. It enables the system to select the appropriate image enhancement technique to be applied in the preprocessing stage, detailed further in Section 2.5. The step is critical as when capturing underwater images, dissolved substances, and particulate matter affect light attenuation (Schettini and Corchs, 2010). This in turn can cause scattering, non-uniform lighting, and create shadows (Lu et al., 2017), making it difficult to detect fish in underwater videos.

To classify freshwater videos based on environmental conditions, we designed a custom Convolutional Neural Network (CNN). The proposed CNN model is trained to detect six environmental conditions, namely, biofilm, bubbles, clear, low lighting, overexposure, and turbidity. The CNN analyzes random frames from input videos and returns probabilities for each of the corresponding environmental conditions. The labels and probabilities of the three (most) prominent environmental conditions are used in the preprocessing stage of the system pipeline. We selected this approach, drawing influence from available literature on weather information estimation from single images (Chu et al., 2017; Ibrahim et al., 2019; Xie et al., 2021), and underwater image classification (Aridoss et al., 2020). The proposed CNN-based machine learning model is illustrated in Fig. 3. The model architecture stems from the VGG16 (also referred to as the OxfordNet) (Simonyan and Zisserman, 2014) CNN architecture and was selected primarily due to its performance on the ImageNet (Deng et al., 2009) dataset.

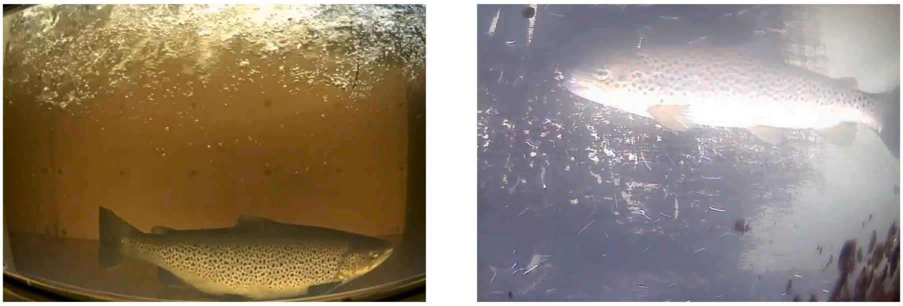
.

2.4.1. *Training the environmental condition model*

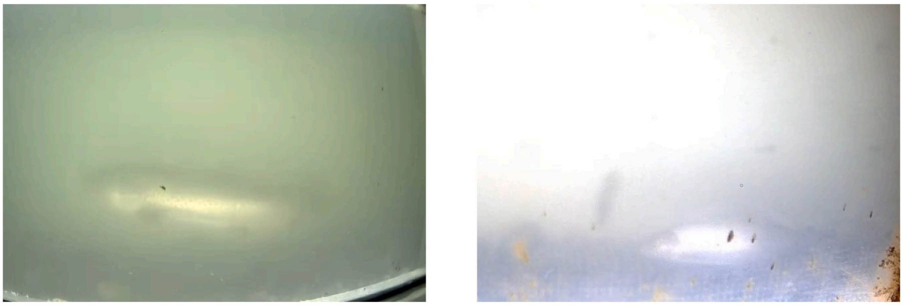
The CNN model for environmental condition classification was developed using a two-step process. First, five separate models based on the proposed CNN architecture illustrated in Fig. 3 were trained and tested. Then, the five models were validated, using a hold-out validation dataset, and the model with the highest accuracy was selected. To train, test and validate the models, the collection of 3000 videos was split into two balanced datasets containing an equal number of fish and no-fish videos from each environmental condition. This resulted in a collection of 1500 videos for the training and testing as well as the validation dataset, indicated in Fig. 4 (1). This unconventional split was required due to the co-occurance of many environmental conditions, where a more typical hold-out strategy using 10 or 20% of all videos for the validation dataset would have resulted in poor coverage of many combinations of environmental conditions. For the training and testing phases, eight random frames from each of the training and testing videos were extracted irrespective of the video lengths. The frame selection process was carried out using a Python script which randomly selected the frames using a uniform distribution. This implied that there could be instances where frames from fish videos did not contain any fish. However, given that the frames were only used for training the



(a) Clear condition (b) Low lighting condition



(c) Air bubbles (d) Periphytic biofilm



(e) Turbidity (f) Light overexposure

Fig. 1. Typical examples of the six different environmental conditions. In the majority of situations, different conditions co-occur, with the exception of the clear condition. As an example, the environmental condition overexposure (f) also includes biofilm growth (d) and turbidity (e).

Table 2
Overview of the videos used in training, testing and validation.

Metric	Value/Description
Video length	5 s up to 5 min
Video format	AVI, MPEG4
Video bit-rate	120 to 3000 kbps
Video frame-rate	5 to 30 fps
Video resolution	320 × 240 to 800 × 600 px
Video file size	0.074 to 102 MB
Camera type	Color and grayscale

environment condition classification models, the potential influence of fish being in the extracted frames was considered negligible.

The 12,000 frames (8 frames * 1500 videos) extracted from the training and testing dataset were then used to generate five different data sets using repeated random sub-sampling (see Fig. 4 (2)). Each of the sets was further split into a training/testing split of 80/20, containing 9600 frames for training and 2400 for testing. Five CNN models using the same model architecture were then trained and tested. All models were trained over 100 epochs and returned an average loss value of 0.0126 with an average accuracy of 99.3% during training and testing. Figures for the individual models are provided in the supplementary material.

To select the best performing model, the five CNN models were validated in a final step using the 1500 videos of the validation dataset. It should be noted that the validation videos were held-out from training and thus contained videos that the models had not encountered before. To validate the models, 1 random frame was extracted from each validation dataset video. After extracting 1500 random frames (1 frame * 1500 videos) all five CNN models were used to classify the videos into the six environmental condition classes. The results of the validation phase are detailed in Section 3.1.

2.5. *Preprocessing*

After completing the environmental classification phase, the system extracts a frame. Image filtering/enhancement is applied then to

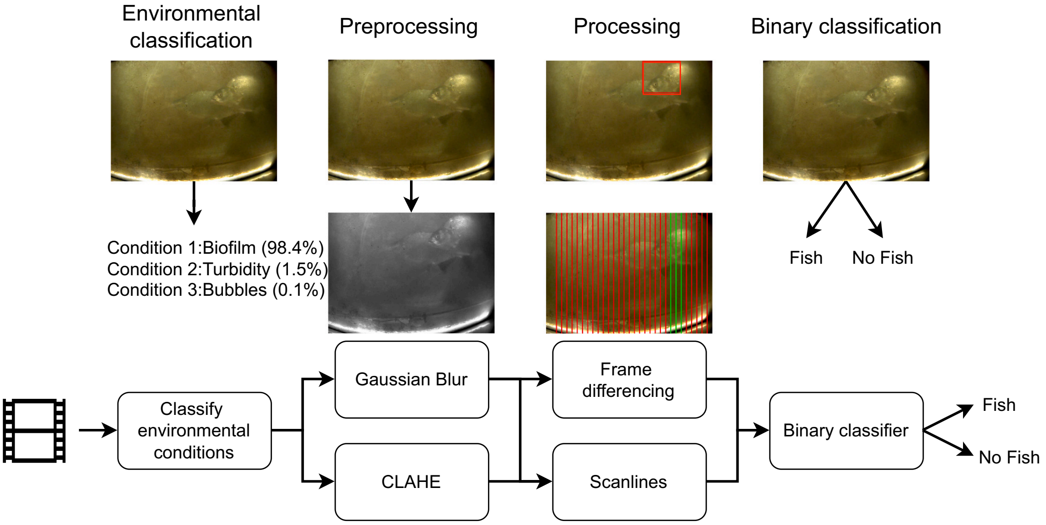

Environmental classification
Preprocessing
Processing
Binary classification
Condition 1:Biofilm (98.4%)
Condition 2:Turbidity (1.5%)
Condition 3:Bubbles (0.1%)
Fish
No Fish
Classify environmental conditions
Gaussian Blur
CLAHE
Frame differencing
Scanlines
Binary classifier
Fish
No Fish

Fig. 2. Overview of the proposed four-stage underwater video classification pipeline. First, the top three environmental conditions are classified. Next, depending on the environmental class, Gaussian blur or Contrast Limited Adaptive Histogram Equalization (CLAHE) is applied to enhance the frames. In the frame classification phase, two competing methods, frame differencing and scanlines are run, producing binary frame data for “fish” (1) or “no fish” (0) for each frame of the video. Finally, the frame-wise classifications are applied to assess whether the video contains fish or not.

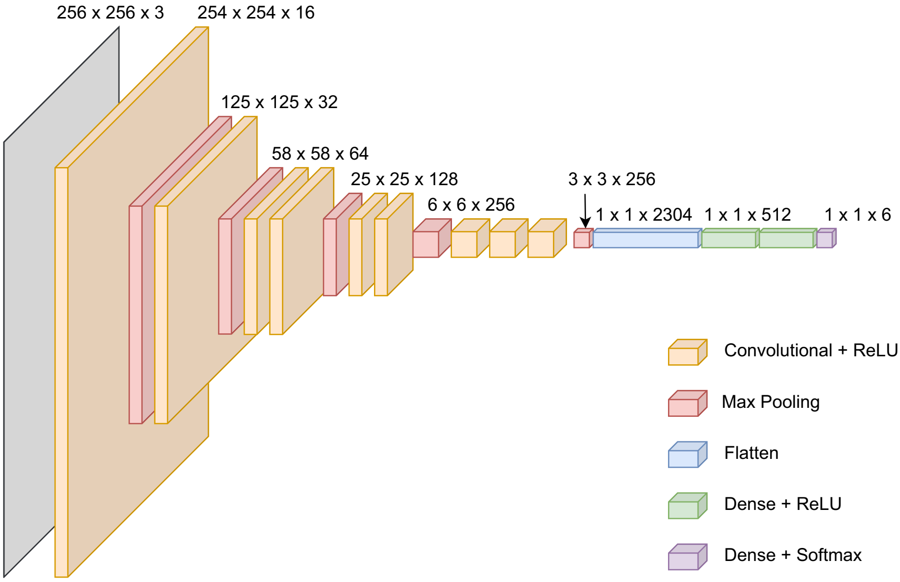
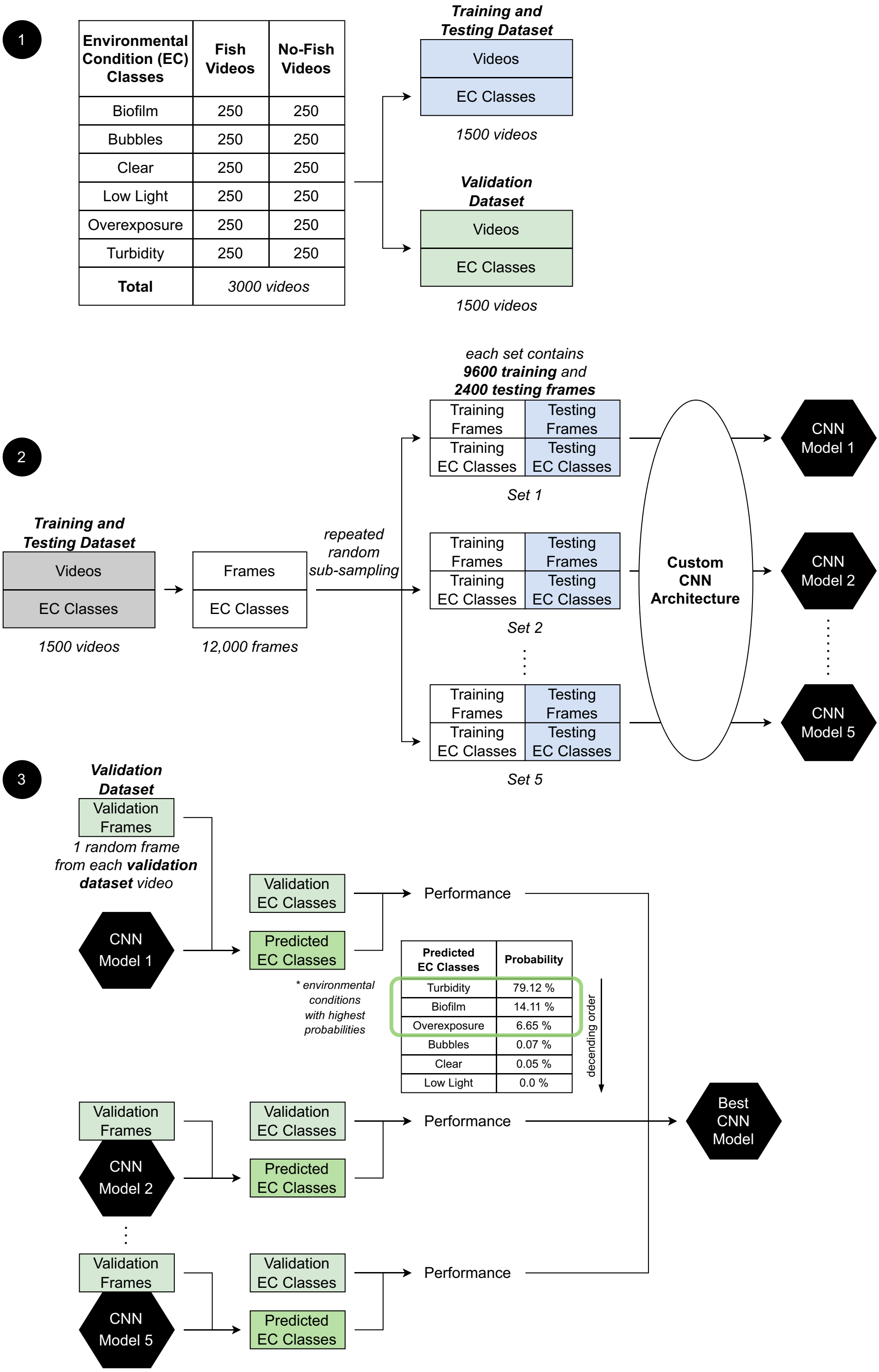


Fig. 3. Structure of the custom CNN model for environmental condition classification.

improve accuracy based on the corresponding environmental condition. Depending on the environmental conditions, two different image enhancement techniques are applied: Gaussian Blur or CLAHE. The parameters values used for each environment condition are provided in Table 3. Gaussian Blur is a commonly used method to smooth the image (Gedraite and Hadad, 2011). The method smooths the pixel intensities using a two-dimensional Gaussian kernel. This reduces unwanted noise, but can also results in the reduction of image features including edges and textures. The filter was applied in this work primarily to reduce false detections caused by bubbles, leaves and other small debris (Marcos et al., 2005; Rathi et al., 2017). The visibility of fish counter videos is often limited by biofilm and turbidity, making it difficult for vision-based methods to clearly differentiate objects from the background. To partially mitigate this issue, we apply Contrast Limited Adaptive Histogram Equalization (CLAHE) to selected video frames (Konovalov et al., 2019; Pengying et al., 2019). The idea behind CLAHE is to



Environmental Condition (EC) Classes	Fish Videos	No-Fish Videos
Biofilm	250	250
Bubbles	250	250
Clear	250	250
Low Light	250	250
Overexposure	250	250
Turbidity	250	250
Total	*3000 videos*	

Predicted EC Classes	Probability
Turbidity	79.12 %
Biofilm	14.11 %
Overexposure	6.65 %
Bubbles	0.07 %
Clear	0.05 %
Low Light	0.0 %

(caption on next page)

Fig. 4. 1) Illustration of the hold-out procedure used for training, testing, and validation of the environmental condition classification model. 2) Repeated random sub-sampling was applied for testing and training and resulted in five CNN models with identical model architecture. 3) The best-performing CNN model in terms of accuracy was used in the proposed fish or no-fish video classification method. The top three environmental conditions, ranked by their probabilities were used to evaluate the model accuracy. Adapted from (Raschka, 2018)

Table 3
Frame differencing and scanline parameters. Kernel size and sigma values are associated only with Gaussian blur, and grid size and the clipping limit apply only to CLAHE.

Frame Differencing						
Environmental condition	Preprocess	Kernel size/Grid size	Sigma/Clipping limit	Min. object area	Size ratio threshold	Threshold
Clear condition	Gaussian blur	15	0	500	0.5	20
Low Light condition	CLAHE	8	1			
Air bubbles	Gaussian blur	15	0			
Turbidity	CLAHE	8	1			
Periphytic biofilm	Gaussian blur	15	0			
Light overexposure	Gaussian blur	15	0			
Scanlines						
Environmental condition	Preprocessing	Kernel Size/ Grid Size	Sigma/ Clipping limit	Variance threshold	Min. detected scanlines	Scanline step value
Clear condition	Gaussian Blur	15	0	750	3	25
Low Light condition	CLAHE	8	1	50		
Air bubbles	Gaussian Blur	15	0	1000		
Turbidity	CLAHE	8	1	20		
Periphytic biofilm	Gaussian Blur	15	0	400		
Light overexposure	Gaussian Blur	15	0	400		

distribute an image's pixel intensities evenly across the entire image. In contrast to standard histogram equalization, which applies the same equalization across the entire image, CLAHE divides the image into tiles and performs equalization separately in each. This also comes with some drawbacks. The first drawback is noise amplification, which is most pronounced with small tile sizes. To address this, contrast limiting is applied, meaning that if any histogram bin is above the set limit, the pixel is clipped and uniformly distributed to other nearby containers. An additional drawback is that the method is considered computationally expensive due to the calculation of different neighborhood histograms and the need to apply a transformation function for each pixel.

2.6. *Main processing*

2.6.1. *Frame differencing*

The frame differencing method performs a background estimation by subtracting the current frame pixel intensities from those of the previous frame (Algethami and Redfern, 2018; Ellenfeld et al., 2021). In the second stage, the absolute value of the pixel-wise differences are taken and filtered based on a single threshold value over the entire image. During the development, multiple values were tested. The high threshold value removed too many moving details, resulting in decreased performance across numerous environmental conditions. Lowering the threshold resulted in an increased number of a false positive detection.

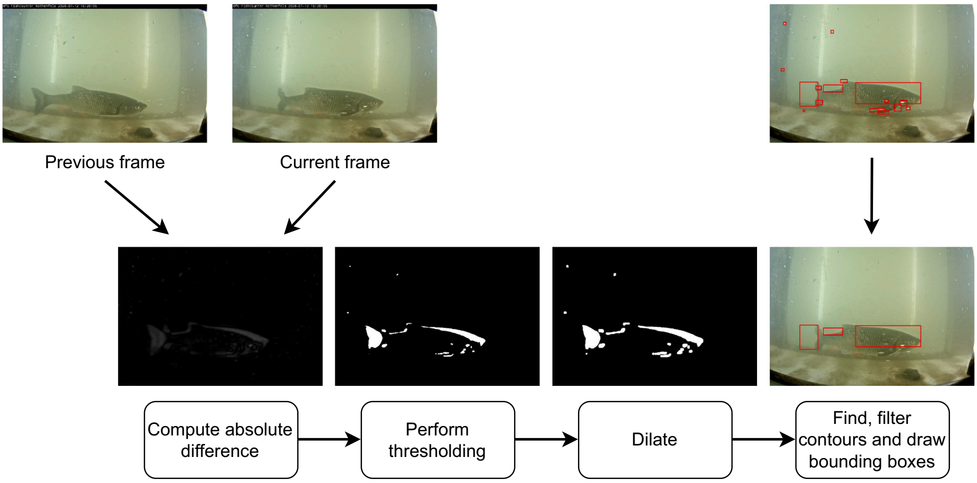

Previous frame
Current frame
Compute absolute difference
Perform thresholding
Dilate
Find, filter contours and draw bounding boxes

Fig. 5. Frame difference processing: After extracting a frame, the absolute difference between the pixel intensities of the current and previous frame is computed. Thresholds are then applied, creating a binary image. Dilation is used to fill small gaps and enhance the contours. Contours are filtered based on their dimensions to remove small objects.

These operations are expressed mathematically as:

$$B(x,y,t) = I(x,y,t-1) \tag{1}$$

$$|I(x,y,t) - I(x,y,t-1)| > Th \tag{2}$$

where B is the background image, x and y are the pixel coordinates, t is time, I is the previous frame background and Th is the threshold. Fig. 5 illustrates the processing pipeline using frame differencing methods. Dilation is applied to the thresholded binary image to fill small holes. We applied a simple chain approximation was used to determine the contours. Finally, contours were filtered by their x, y coordinates, width, and height. Contours that did not meet the set minimum size and area threshold were excluded from further processing, as described in Table 3.

2.6.2. Scanlines

A classic and robust method for motion detection are scanlines, which can be applied (Lin et al., 2003) in both vertical and horizontal directions (Zhang and Jin, 2019). The simple method relies on monitoring and detecting changes the pixel variance over multiple frames, which changes over time and space, providing an adaptive motion detection method. As illustrated in Fig. 6, scanlines are by default computationally lightweight and require sparse calculations on selected pixel columns instead of processing the entire frame. After extracting the pixel columns, RGB intensities are separated into three channels. The variance is computed for each scanline, and compared to a user-defined threshold. Compared to frame differencing, scanlines have the following advantages:

- Adaptive variance: Scanlines dynamically change as the imagery intensities vary over time and space.
- Dynamic regions of interest: Sensitivity can be adjusted along both the x and y-axes. This feature is especially helpful considering the six different environmental conditions, where regions with bubbles can be masked out to reduce false positives.

2.7. Binary classification

The final phase of the video classification pipeline is running the binary classifier to determine if the selected video is labeled as “fish” or “no fish”. The binary classifier exploited temporal patterns found in the frame-wise binary classification. Positive frame-wise classification most frequently occurred when fish, bubbles or floating debris and leaves were present. In most cases, it was observed that if a fish appears in the video unaffected by the environmental condition, the number of sequential positive frames was consistently larger than the number of negative frames. In addition, based on the detected environmental conditions, the number of positive and negative frames over a threshold count were evaluated. The threshold values for each environmental condition were retrieved using Design of Experiments (DOE). The principle of DOE is finding out factors that influence the outcome by manipulating input values simultaneously to identify critical interactions. The method also mitigates the possibility of missing input's influence when testing one at a time.

The process is initiated by searching for a positively marked frame, and the counter increases by one each time a new positive frame is found. If the frame is negative, the counter increases the sum of negative frames. The evaluation stops and assigns “no fish” if the negative frame threshold is reached. If a positive frame threshold is reached, the scanning stops and the video was classified as “fish”. If the end of the video frames are processed without exceeding the positive or negative frame count threshold for a given video, then it was labeled as “no fish”. From time complexity standpoint the algorithm is regarded as linear i.e., O(n).

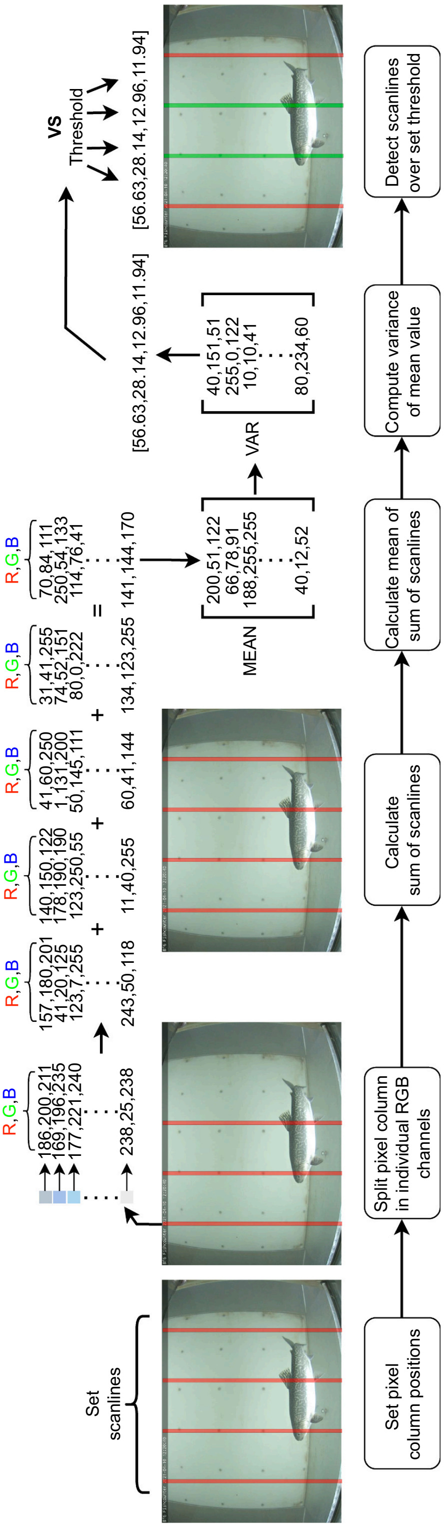


Fig. 6. Scanlines for motion detection: Each vertical pixel column has its RGB values split. The mean and variance of the pixel intensities over a scanline are computed across multiple frames. If the variance exceeds the set threshold, the scanline is labeled as a positive fish detection (shown in green). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Algorithm 1. Binary classifier algorithm

```
$D$ represents the frame data array, which is given as the input, $f_p$ and $f_n$ - number of positive
and negative frames, i - current frame index, $n_{fl}$ and $p_{fl}$ represent threshold
    procedure ANALYZE()
        while $i < size(D)$ do
            if $D[i] == 1$ then
                $f_p = f_p + 1$
                $i = i + 1$
            else
                if $f_p > 0$ then
                    $f_n = f_n + 1$
                    if $f_n == n_{fl}$ then
                        if $f_p == p_{fl}$ then return True
                else
                    $f_n = 0$
                    $f_p = 0$
                $i = i + 1$
        return False
```

2.8. Evaluation metrics

The performance of the fish / no-fish binary classification methods was evaluated using accuracy, precision, recall sensitivity, recall specificity, and the F1 Score. Videos classified as “fish” or “no fish” were compared to human labels which served as the ground truth. If a video was automatically classified as “fish” and the ground truth was also “fish”, this was considered as a True Positive (TP). If the classified video and ground truth were both classified as “no fish,” the automated classification represented a True Negative (TN). If a video was classified as “fish” while the ground truth was “no fish,” this was counted as a False Positive (FP). Finally, automated classifications of “no fish” which should have been classified as “fish”, were assigned as False Negatives (FN).

2.8.1. Accuracy

The evaluation process includes the accuracy, which returns the percentage of correct predictions with respect to the total number of videos.

$$\text{Accuracy}(\%) = \frac{\text{True positives} + \text{True negatives}}{\text{Total samples}}^{*}100 \tag{3}$$

While the accuracy metric estimates the classification performance, we also calculated the precision, recall specificity, recall sensitivity, and the F1 score. A combination of these metrics thus provides a thorough overview of the classification performance. The precision evaluates how many videos were true positives from all videos with positive predictions.

$$\text{Precision}(\%) = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}^{*}100 \tag{4}$$

Recall calculates the amount of correctly predicted out of all possible positives. The opposite of this measure is called the “recall specificity”, which provides the number of false predictions from all possible negatives. The recall is calculated as:

$$\text{Recall}(\%) = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}^{*}100 \tag{5}$$

Our final evaluation parameter is the F1 score, which provides a measure between precision and recall, calculated as the harmonic mean of the two metrics. The F1 score was evaluated as:

$$\text{F1score}(\%) = 2^{*}\frac{\text{Precision}^{*}\text{Recall}}{\text{Precision} + \text{Recall}}^{*}100 \tag{6}$$

2.8.2. Efficiency

In addition to measuring and evaluating the classification accuracy, this work aimed to provide a thorough overview of the proposed classification pipeline's efficiency and feasibility. The efficiency was evaluated by calculating the number of frames the system is able to process per second, which is described with the following equation:

$$\text{Frames processed per second (FPPS)} = \frac{\text{Number of frames}}{\text{Processing time}} \tag{7}$$

The proposed approach processed multiple videos simultaneously. Therefore, calculating only the number of processed frames per second provides a rough estimation of processing efficiency. Each of the tested hardware platforms differs in terms of available computational resources and architecture. We therefore computed the frames processed per second per thread/core, to provide a more normalized measure of the processing rate efficiency.

2.8.3. Feasibility

The feasibility of the entire system is evaluated based on the cost and energy consumption. The unit cost must be considered, especially for large-scale deployments. Considering the HDP hardware, only the CPU

Table 4
Performance metrics of the five CNN models over the validation dataset.

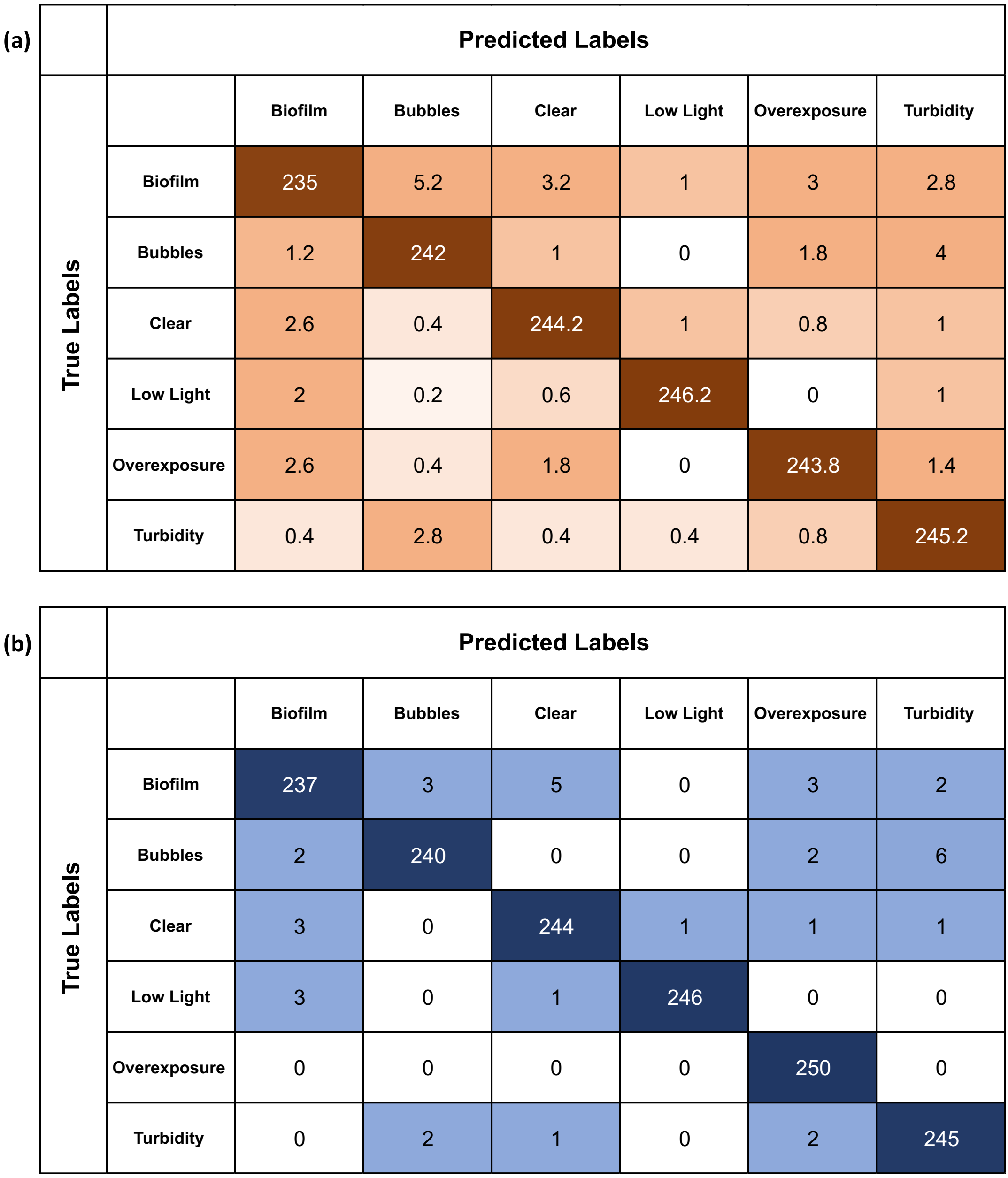
Metric	Model 1	Model 2	Model 3	Model 4 *	Model 5
Accuracy	99.1%	99.1%	98.8%	99.2%	98.9%
Precision	97.3%	97.4%	96.5%	97.5%	96.8%
Recall Sensitivity	97.3%	97.4%	96.5%	97.5%	96.8%
Recall Specificity	99.5%	99.5%	99.3%	99.5%	99.4%
F1 Score	97.3%	97.4%	96.5%	97.5%	96.8%
Avg. True Positives	243.2 (16.2%)	243.5 (16.2%)	241.3 (16.0%)	243.7 (16.2%)	242 (16.2%)
Avg. False Positives	6.8 (0.5%)	6.5 (0.4%)	8.7 (0.6%)	6.3 (0.4%)	8 (0.5%)
Avg. True Negatives	1243.2 (82.8%)	1243.5 (83.0%)	1241.3 (82.8%)	1243.7 (83.0%)	1242 (82.8%)
Avg. False Negatives	6.8 (0.5%)	6.5 (0.4%)	8.7 (0.6%)	6.3 (0.4%)	8 (0.5%)

[*] Best CNN model based on Accuracy.

price at the Manufacturer's Suggested Retail Price (MSRP) was considered, as the total system cost can vary significantly due to the selection of other components. The system's power consumption was measured during video processing. It plays a crucial role when deploying a battery-powered embedded system if access to a continuous power source is unavailable. The feasibility was described as the ratio of the processing efficiency and the power consumption:

$$\text{Frames processed per Watt (FPPW)} = \frac{\text{Frames processed per second}}{\text{Watts}} \tag{8}$$

Power consumption measurements were performed using software-based monitoring, when the system supported it. Only MediaTek Pumpkin i500 power consumption was measured using a high-precision external power meter.



(a)

Predicted Labels / True Labels

	Biofilm	Bubbles	Clear	Low Light	Overexposure	Turbidity
Biofilm	235	5.2	3.2	1	3	2.8
Bubbles	1.2	242	1	0	1.8	4
Clear	2.6	0.4	244.2	1	0.8	1
Low Light	2	0.2	0.6	246.2	0	1
Overexposure	2.6	0.4	1.8	0	243.8	1.4
Turbidity	0.4	2.8	0.4	0.4	0.8	245.2

(b)

Predicted Labels / True Labels

	Biofilm	Bubbles	Clear	Low Light	Overexposure	Turbidity
Biofilm	237	3	5	0	3	2
Bubbles	2	240	0	0	2	6
Clear	3	0	244	1	1	1
Low Light	3	0	1	246	0	0
Overexposure	0	0	0	0	250	0
Turbidity	0	2	1	0	2	245

Fig. 7. Confusion matrices generated during the cross-validation phase of the designed CNN models. (a) The average of confusion matrices of the five CNN models. (b) Confusion matrix of the best CNN model that was subsequently used in the next phase of the proposed pipeline.

Table 5
Results of individual conditions tested with frame difference and scanlines.

Metric	Condition: Clear condition		Condition: Low Light condition		Condition: Air bubbles	
	Frame Difference	Scanlines	Frame Difference	Scanlines	Frame Difference	Scanlines
Accuracy	91.9%	88.8%	94.8%	91.2%	84.7%	77.2%
Precision	99.0%	100%	90.6%	88.7%	78.2%	75.4%
Recall Sensitivity	84.4%	77.6%	97.9%	94.4%	94.9%	80.8%
Recall Specificity	99.2%	100%	91.4%	88.0%	75.2%	73.6%
F1 Score	91.1%	87.4%	95.1%	91.5%	85.7%	78.0%
True Positives	103 (41.7%)	97 (38.8%)	125 (50.0%)	118 (47.2%)	111 (45.9%)	101 (40.4%)
False Positives	1 (0.4%)	0 (0.0%)	13 (5.2%)	15 (6.0%)	31 (12.8%)	33 (13.2%)
True Negatives	124 (50.2%)	125 (50.0%)	112 (44.8%)	110 (44.0%)	94 (38.8%)	92 (36.8%)
False Negatives	19 (7.69%	28 (11.2%)	0 (0.0%)	7 (2.8%)	6 (2.5%)	24 (9.6%)

Metric	Condition: Periphytic biofilm		Condition: Turbidity		Condition: Light overexposure	
	Frame Difference	Scanlines	Frame Difference	Scanlines	Frame Difference	Scanlines
Accuracy	87.6%	82.8%	91.6%	72.4%	80.4%	70.8%
Precision	89.2%	77.3%	88.1%	70.3%	72.9%	64.9%
Recall Sensitivity	85.6%	92.8%	96.0%	77.6%	96.8%	90.4%
Recall Specificity	89.6%	72.8%	87.2%	67.2%	64.0%	51.2%
F1 Score	87.4%	84.3%	91.9%	73.8%	83.2%	75.6%
True Positives	107 (42.8%)	116 (46.4%)	119 (47.8%)	97 (38.8%)	121 (48.4%)	113 (45.2%)
False Positives	13 (5.2%)	34 (13.6%)	16 (6.4%)	41 (16.4%)	45 (18.0%)	61 (24.4%)
True Negatives	112 (44.8%)	91 (36.4%)	109 (43.8%)	84 (33.6%)	80 (32.0%)	64 (25.6%)
False Negatives	18 (7.2%)	9 (3.6%)	5 (2.0%)	28 (11.2%)	4 (1.6%)	12 (4.8%)

3. Results

3.1. Environmental conditions

To ensure a robust evaluation of the CNN model for environment condition classification, the five CNN models presented in Section 2.4 were cross-validated using the hold-out validation dataset containing 1500 videos. As described in Section 2.4.1, one random frame was extracted from each video in the dataset, giving us a total of 1500 frames for validation. These frames were then processed through the five CNN models, and performance metrics for each model were generated. Table 4 summarizes the acccuracy, precision, recall sensitivity, recall specificity and the F1 scores of the five models from the validation phase. Fig. 7 (a) provides an overview of the averaged confusion matrices from all five CNN models. CNN Model 4 had the best overall performance with an accuracy of 99.2% and an F1 score of 97.5%., and thus only CNN Model 4 was used in the subsequent fish or no-fish classification pipeline. The validation performance of CNN Model 4 with respect to each of the six environmental conditions is shown in the confusion matrix of Fig. 7 (b).

3.2. Fish and no-fish video classification

Overall, frame differencing and scanline-based methods classified videos with high accuracy. (Table 5), where the "clear condition" environmental condition using frame differencing achieved an average classification of 91.9%, and an F1 score of 91.1%. Scanlines performed 3.1% worse, with a score of 88.8% and an F1 score of 87.2%. Both methods showed similar accuracy in the "low light condition" environment; both retained an accuracy and F1 score over 91.0%. The environmental condition "air bubbles", exhibited a decrease in accuracy, where frame differencing had 84.7%, whereas scanlines dropped to 77.2%. Similar results were achieved in "turbidity", where frame differencing performed ca. 5% better compared to scanlines. Scanlines performed even lower under "periphytic biofilm" conditions, where the accuracy dropped to 72.4%, whereas frame differencing performed the best, with an accuracy of 91.6%. Videos classified as having "light overexposure" achieved results of 80.4% and 70.8% respectively. Final testing was performed with the randomized mixed dataset, where the results were averaged across five iterations (Table 6). The average accuracy for frame differencing was 88.5%. Scanlines performed around 6% percent worse than frame differencing, achieving an accuracy of 82.1%.

Table 6
Averaged fish / no-fish classification results after five iterations of the balanced dataset using 1500 videos using frame differencing and scanlines. During each iteration, 500 videos were randomly selected across six environmental conditions.

Metric	Frame Difference	Scanlines
Accuracy	88.7%	81.2%
Precision	85.6%	78.3%
Recall sensitivity	92.9%	86.2%
Recall specificity	84.6%	76.1%
F1 Score	89.1%	82.1%
True Positives	228.4 (46.0%)	215.6 (43.1%)
False Positives	38.4 (7.7%)	59.8 (11.9%)
True Negatives	211.6 (42.7%)	190.2 (38.0%)
False Negatives	17.4 (3.5%)	34.4 (6.8%)

3.3. Hardware comparison

Based on the results shown in Table 7, we compared the two High-Performance Desktop (HPD) machines. Both systems achieved a minimum speed of 60 frames per second, using all available CPU cores/threads. The Intel i7 12700K with frame differencing had a mean speed of 150 frames per second (fps). Scanlines performed worse, achieving an average frame rate of 103 fps. The Ryzen 5900× performed the best, achieving 200 fps with frame differencing and 160 fps with scanlines. Identical tests were also conducted on three different embedded hardware. Both Jetson Nano and MediaTek Pumpkin i500 achieved frame rates of greater than 40 fps, with both frame differencing and scanlines. The Raspberry Pi 4 was not able to exceed 30 frames per second. Detailed tables comparing the hardware are provided in the in Table 7.

4. Discussion

Estimates of the environmental conditions can be used to automatically and adaptively adjust the fish / no-fish classification model hyperparameters, which in turn improves the classification accuracy. In addition to being accurate, the proposed classification process is computationally efficient, as it relies only on a few extracted frames from each video, achieving an F1 score of up to 98%. These results are

Table 7
High Performance Desktop (HPD) hardware configurations, MediaTek Pumpkin i500, Raspberry Pi 4, Nvidia Jetson Nano Technical specifications and performance metrics. Frames processed per second (FPPS) is reported per core/thread.

Parameters	HPD 1	HPD 2	Nvidia Jetson Nano	Raspberry Pi 4	MediaTek Pumpkin i500
CPU/Microprocessor	AMD Ryzen 95,900×	Intel Core i7-12700K	Cortex-A57	Cortex-A72	Cortex-A73 Cortex-A53
Cores/Threads	12/24	12/20	4/4	4/4	8/8
Core Clock (GHz)	3.7	3.8	1.43	1.5	2.0
Memory	64 GB DDR4	32 GB DDR5	4 GB DDR4	4 GB DDR4	2 GB DDR4
Storage	1 TB	1 TB	32 GB	32 GB	16 GB
Operating System	Ubuntu 20.04	Ubuntu 20.04	Ubuntu 20.04	Raspberry Pi OS	Yocto Linux
Power consumption (W)	183	221	3.75	4.2	3.8
Cost ()	549	419	110	60	199
Performance metrics	HPD 1	HPD 2	Nvidia Jetson Nano	Raspberry Pi 4	MediaTek Pumpkin i500
Geekbench 5 Single-Core	1668	2075	228	231	299
Geekbench 5 Multi-Core	15,404	15,617	819	674	969
FPPS (Frame Diff.)	153	160	46	18	44
FPPS (Scanlines)	220	153	43	24	45
FPPW (Frame Diff.)	0.3	0.2	12.1	4.3	7.4
FPPW (Scanlines)	0.8	0.6	11.5	5.7	12.0

comparable to the underwater deep learning-based and unsupervised object detector systems tested by Coro and Bjerregaard Walsh (2021).

However, it was found that frame image enhancement can also lead to incorrect environmental condition classification. As an example, the Gaussian Blur filter can smooth out colors or textures used by the CNN classifier. Environmental conditions such as biofilm or overexposure frequently occurred in different image regions. This can result in a non-isotropic, patchy distortion of the pixel intensities, making it difficult for many deep learning models to detect environmental conditions as objects. However, as demonstrated by Ibrahim et al. (Ibrahim et al., 2019), and others (Xia et al., 2020; Xiao et al., 2021; Xie et al., 2021), the VGG (Simonyan and Zisserman, 2014) and ResNet (He et al., 2016) architecture models are able to detect non-object features for weather conditions (e.g. fog, glare) in terrestrial images. Therefore, the proposed model was designed based on the VGG architecture. Unlike conventional VGG16 or VGG19 models (the numeric value corresponds to the number of layers), our CNN model contained only 12 layers (see Fig. 3).

Lowering the number of layers reduces the model size, improving the system's computational performance. Standard VGG and ResNet models use an input image of size 224 × 224 pixels, whereas our custom model used images of 256 × 256 pixels. The choice of input image size was based on our experimentation, where it was found that overall classification performance could be improved with a minor increase in the image resolution.

In addition to problems caused by incorrect image enhancement, videos with low frame rates or resolutions can also have an impact on fish detection accuracy. Methods such as frame differencing depend on high frame rates. Otherwise, the technique has difficulties separating moving objects from the background. In addition, with low resolution, the system also has problems separating small fish from random floating debris.

The main focus of this work is on the correct classification of videos with and without fish in them. Thus our work serves as a precursor to automated fish counters which must also classify the fish species and count the number of individuals. Similar to our objective, (Konovalov et al., 2019) developed a highly accurate (FP = 0.17%, FN = 0.6%) wild, marine fish and no-fish video classification method. Underwater video obtained in rivers presents its own sets of challenges, such as detritus (e.g. leaves, small branches, garbage), turbidity, air bubbles, reflections, and biofilm growth. For example, biofilm and turbidity both result in hazed and blurred imagery, making it difficult to robustly detect fish moving in front of the camera. Detritus and air bubbles may lead to false-positives as fish identification algorithms may confuse these artifacts with moving fish. At sites where these problems are addressed, it may be possible to estimate fish biomass directly from estimates of fish size (Li et al., 2021). Our proposed method could also be combined with cloud-based hourly fish counts by combining acoustic fish tracking with underwater video fish counter data to follow individual fish through hydropower plants as they migrate up and downstream (Tuhtan et al., 2020; Yang et al., 2022).

Comparing the two fish detection methods proposed in this work, frame differencing performed the best overall, achieving a mean accuracy of 88.4%, while scanlines achieved a mean accuracy of 82.1%. These results are encouraging, but there remains room for improvement as (Hernández-Ontiveros et al. (2018) showed that an embedded fish counter running on a Raspberry Pi with controlled illumination and monotone background could achieve an individual fish counting accuracy of up to 98%. However, it is also important to note that in wild, unstructured environments, the F1-score can drop below 50% (Labao and Naval, 2019). Frame differencing implements a filter to exclude objects which can cause false positives based on the object dimensions and shape. The method was also found to create false negatives if the fish's head only briefly enters the video and for fast-moving, small fish. In order to reduce the number of false negatives using frame differencing, we recommend that the lowest video frame rate is set at a minimum of 15 fps.

Considering the video processing speed, both HPDs tested in this work processed videos at more than 150 fps, whereas the embedded hardware ran at speeds of 18 to 46 fps. Based on our recommendation of a minimum of 15 fps, all tested hardware were deemed to be adequate. At sites where power consumption is a limiting factor, HPDs are poor candidates due to their high energy consumption of more than 180 W, whereas the Raspberry Pi 4, which was the slowest tested embedded hardware, consumed only 4.2 W. In cases where the power consumption is critical, the MediaTek i500 and Jetson Nano were found to be similar, having a power consumption of 3.75 and 3.8 W, respectively. Although power consumption is often the main limitation, autonomous underwater cameras may run out of memory long before batteries are depleted, as discussed in Mouy et al. (2020).

5. Conclusions

We propose an environmentally adaptable and computationally lightweight solution for classifying underwater videos based on the presence or absence of fish. The results indicate that the proposed method can provide accurate binary video classification (fish = 1, no-fish = 0) under six different environmental conditions commonly occurring at underwater cameras installed in rivers. This was accomplished using a bespoke multi-stage video processing pipeline which included a CNN-based environmental classifier in conjunction with frame differencing, scanlines, and a binary classifier.

The model was trained, tested, and validated on 3000 balanced (i.e., uniformly distributed six environmental conditions) videos with fish (1500) and without fish (1500). Training and testing of the CNN-based environmental conditions classifier were performed using repeated random sub-sampling of 12,000 frames extracted from 1500 videos, while validation was carried out on the remaining 1500 videos, which were withheld for validation and not used for training or testing. The best performing CNN model for environmental conditions classification was then chosen to be used for the automated fish / no-fish sorting using frame differencing and scanlines.

The underwater video processing pipeline was also shown to be suitable for low-cost embedded hardware, which may allow for real-time fish counters. In addition, the proposed methods are largely platform-independent. This allows for their deployment in edge computer vision systems, high-performance PCs, or as a cloud-based solution.

Future research will explore improving the computational performance of the proposed pipeline using GPUs, including their optimization on low-cost embedded hardware. We are optimistic that the inclusion of optimized GPUs may allow for further improvements in the computational performance and classification accuracy when compared with the CPU-based methods presented in this work. In addition, we will explore the use of the proposed environmentally-adaptive outdoor video monitoring system for biodiversity monitoring to automatically sort continuously recorded videos including terrestrial animals and birds.

Declaration of Competing Interest

The authors declare there are no competing interests.

Data availability

The authors do not have permission to share data.

Acknowledgments

This work in the project “ICT programme” was supported by the European Union through European Social Fund. The research was supported by the “Smart Fish Counter” project funded by the German Federal Institute of Hydrology (BfG), by providing videos from Koblenz and Mallliß Vaki River Watcher installations (see Table 1). In addition, BfG staff members Bernd Mockenhaupt, Nicole Scheifhacken, Cornelia Schuetz and Julia Droste provided valuable insight, feedback, and suggestions to improve this manuscript. We are also thankful to Christian Haas and Philipp Thumser from IAHMHYDRO GmbH for providing underwater videos from commerical Vaki River Watcher installations (see Table 1). The authors wish to thank the anonymous reviewers for their insightful comments to improve this manuscript.

References

Algethami, N., Redfern, S., 2018. Combining accumulated frame differencing and corner detection for motion detection. In: Computer Graphics and Visual Computing (CGVC). https://doi.org/10.2312/CGVC.20181202.

Aridoss, M., Dhasarathan, C., Dumka, A., Loganathan, J., 2020. DUICM deep underwater image classification mobdel using convolutional neural networks. Int. J. Grid High Perform. Comput. 12, 88–100. https://doi.org/10.4018/ijghpc.2020070106.

Boom, B.J., He, J., Palazzo, S., Huang, P.X., Beyan, C., Chou, H.M., Lin, F.P., Spampinato, C., Fisher, R.B., 2014. A research tool for long-term and continuous analysis of fish assemblage in coral-reefs using underwater camera footage. Ecol. Inform. 23, 83–97. https://doi.org/10.1016/j.ecoinf.2013.10.006.

Ceballos, G., Ehrlich, P.R., Dirzo, R., 2017. Biological annihilation via the ongoing sixth mass extinction signaled by vertebrate population losses and declines. Proc. Natl. Acad. Sci. 114, E6089–E6096. https://doi.org/10.1073/pnas.1704949114.

Chu, W.T., Zheng, X.Y., Ding, D.S., 2017. Camera as weather sensor: estimating weather information from single images. J. Vis. Commun. Image Represent. 46, 233–249. https://doi.org/10.1016/j.jvcir.2017.04.002.

Coro, G., Bjerregaard Walsh, M., 2021. An intelligent and cost-effective remote underwater video device for fish size monitoring. Ecol. Inform. 63, 101311 https://doi.org/10.1016/j.ecoinf.2021.101311.

Deinet, S., Scott-Gatty, K., Rotton, H., Twardek, W., Marconi, V., McRae, L., Baumgartner, L., Brink, K., Claussen, J., Cooke, S., Darwall, W., Eriksson, B., Garcia de Leaniz, C., Hogan, Z., Royte, J., Silva, L., Thieme, M., Tickner, D., Waldman, J., Wanningen, H., Weyl, O., Berkhuysen, A., 2020. The Living Planet Index (LPI) for Migratory Freshwater Fish: Technical Report.

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. https://doi.org/10.1109/CVPR.2009.5206848.

Dwivedi, A.K., 2021. Role of digital technology in freshwater biodiversity monitoring through citizen science during covid-19 pandemic. River Res. Appl. 37, 1025–1031. https://doi.org/10.1002/rra.3820.

Ellenfeld, M., Moosbauer, S., Cardenes, R., Klauck, U., Teutsch, M., 2021. Deep fusion of appearance and frame differencing for motion segmentation. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 4334–4344. https://doi.org/10.1109/CVPRW53098.2021.00490.

Fabic, J., Turla, I., Capacillo, J., David, L., Naval, P.C., 2013. Fish population estimation and species classification from underwater video sequences using blob counting and shape analysis. In: 2013 IEEE International Underwater Technology Symposium (UT), pp. 1–6. https://doi.org/10.1109/UT.2013.6519876.

Feio, M.J., Hughes, R.M., Callisto, M., Nichols, S.J., Odume, O.N., Quintella, B.R., Kuemmerlen, M., Aguiar, F.C., Almeida, S.F., Alonso-EguíaLis, P., Arimoro, F.O., Dyer, F.J., Harding, J.S., Jang, S., Kaufmann, P.R., Lee, S., Li, J., Macedo, D.R., Mendes, A., Mercado-Silva, N., Monk, W., Nakamura, K., Ndiritu, G.G., Ogden, R., Peat, M., Reynoldson, T.B., Rios-Touma, B., Segurado, P., Yates, A.G., 2021. The biological assessment and rehabilitation of the world’s rivers: an overview. Water 13, 371. https://doi.org/10.3390/w13030371.

Fjeldstad, H.P., Pulg, U., Forseth, T., 2018. Safe two-way migration for salmonids and eel past hydropower structures in europe: a review and recommendations for best-practice solutions. Mar. Freshw. Res. 69, 1834. https://doi.org/10.1071/mf18120.

Fuentes-Pérez, J.F., García-Vega, A., Bravo-Córdoba, F.J., Sanz-Ronda, F.J., 2021. A step to smart fishways: an autonomous obstruction detection system using hydraulic modeling and sensor networks. Sensors 21. https://doi.org/10.3390/s21206909.

Gedraite, E.S., Hadad, M., 2011. Investigation on the effect of a gaussian blur in image filtering and segmentation. In: Proceedings ELMAR-2011, pp. 393–396.

Haas, C., Thumser, P., Mockenhaupt, B., Schletterer, M., 2018. The system vaki riverwatcher as a tool for long-term monitoring of fish migration in fishways. WASSERWIRTSCHAFT 108, 41–48.

Harvey, E., Shortis, M., 1995. A system for stereo-video measurement of sub-tidal organisms. Mar. Technol. Soc. J. 29, 10–22.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778. https://doi.org/10.1109/CVPR.2016.90.

Hernández-Ontiveros, J., Inzunza-González, E., García-Guerrero, E., López-Bonilla, O., Infante-Prieto, S., Cárdenas-Valdez, J., Tlelo-Cuautle, E., 2018. Development and implementation of a fish counter by using an embedded system. Comput. Electron. Agric. 145, 53–62. https://doi.org/10.1016/j.compag.2017.12.023.

Hu, J., Zhao, D., Zhang, Y., Zhou, C., Chen, W., 2021. Real-time nondestructive fish behavior detecting in mixed polyculture system using deep-learning and low-cost devices. Expert Syst. Appl. 178, 115051 https://doi.org/10.1016/j.eswa.2021.115051.

Ibrahim, M.R., Haworth, J., Cheng, T., 2019. Weathernet: recognising weather and visual conditions from street-level images using deep residual learning. ISPRS Int. J. Geo Inf. 8, 549. https://doi.org/10.3390/ijgi8120549.

Konovalov, D.A., Saleh, A., Bradley, M., Sankupellay, M., Marini, S., Sheaves, M., 2019. Underwater fish detection with weak multi-domain supervision. In: 2019 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. https://doi.org/10.1109/IJCNN.2019.8851907.

Labao, A.B., Naval, P.C., 2019. Cascaded deep network systems with linked ensemble components for underwater fish detection in the wild. Ecol. Inform. 52, 103–121. https://doi.org/10.1016/j.ecoinf.2019.05.004.

Lennox, R.J., Paukert, C.P., Aarestrup, K., Auger-Méthé, M., Baumgartner, L., Birnie-Gauvin, K., Bøe, K., Brink, K., Brownscombe, J.W., Chen, Y., Davidsen, J.G., Eliason, E.J., Filous, A., Gillanders, B.M., Helland, I.P., Horodysky, A.Z., Januchowski-Hartley, S.R., Lowerre-Barbieri, S.K., Lucas, M.C., Martins, E.G., Murchie, K.J., Pompeu, P.S., Power, M., Raghavan, R., Rahel, F.J., Secor, D., Thiem, J.D., Thorstad, E.B., Ueda, H., Whoriskey, F.G., Cooke, S.J., 2019. One hundred pressing questions on the future of global fish migration science, conservation, and policy. Front. Ecol. Evol. 7 https://doi.org/10.3389/fevo.2019.00286.

Li, D., Hao, Y., Duan, Y., 2020. Nonintrusive methods for biomass estimation in aquaculture with emphasis on fish: a review. Rev. Aquac. 12, 1390–1411. https://doi.org/10.1111/raq.12388.

Li, D., Miao, Z., Peng, F., Wang, L., Hao, Y., Wang, Z., Chen, T., Li, H., Zheng, Y., 2021. Automatic counting methods in aquaculture: a review. J. World Aquacult. Soc. 52, 269–283. https://doi.org/10.1111/jwas.12745.

Lin, S.F., Chang, Y.L., Chen, L.G., 2003. Motion adaptive interpolation with horizontal motion detection for deinterlacing. IEEE Trans. Consum. Electron. 49, 1256–1265. https://doi.org/10.1109/TCE.2003.1261227.

Lu, H., Li, Y., Zhang, Y., Chen, M., Serikawa, S., Kim, H., 2017. Underwater optical image processing: a comprehensive review. Mob. Network Appl. 22, 1204–1211. https://doi.org/10.1007/s11036-017-0863-4.

Mader, H., Brandl, A., Käfer, S., 2020. Design and function monitoring of an enature® vertical slot fish pass in a large potamal river in Carinthia/Austria. Water 12. https://doi.org/10.3390/w12020551.

Mallet, D., Pelletier, D., 2014. Underwater video techniques for observing coastal marine biodiversity: a review of sixty years of publications (1952–2012). Fish. Res. 154, 44–62. https://doi.org/10.1016/j.fishres.2014.01.019.

Marcos, M.S.A.C., Soriano, M.N., Saloma, C.A., 2005. Classification of coral reef images from underwater video using neural networks. Opt. Express 13, 8766–8771. https://doi.org/10.1364/OPEX.13.008766.

Mouy, X., Black, M., Cox, K., Qualley, J., Mireault, C., Dosso, S., Juanes, F., 2020. Fishcam: a low-cost open source autonomous camera for aquatic research. HardwareX 8, e00110. https://doi.org/10.1016/j.ohx.2020.e00110.

Pengying, T., Pedersen, M., Hardeberg, J.Y., Museth, J., 2019. Underwater fish classification of trout and grayling. In: 2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), pp. 268–273. https://doi.org/10.1109/SITIS.2019.00052.

Raschka, S., 2018. Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning. CoRR abs/1811.12808. URL: http://arxiv.org/abs/1811.12808, arXiv:1811.12808.

Rathi, D., Jain, S., Indu, S., 2017. Underwater fish species classification using convolutional neural network and deep learning. In: 2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR), pp. 1–6. https://doi.org/10.1109/ICAPR.2017.8593044.

Salman, A., Maqbool, S., Khan, A.H., Jalal, A., Shafait, F., 2019. Real-time fish detection in complex backgrounds using probabilistic background modelling. Ecol. Inform. 51, 44–51. https://doi.org/10.1016/j.ecoinf.2019.02.011.

Schettini, R., Corchs, S., 2010. Underwater image processing: state of the art of restoration and image enhancement methods. EURASIP J. Adv. Signal Process. 2010 https://doi.org/10.1155/2010/746052.

Sharma, S., Shakya, A., Panday, S.P., 2016. Fish counting from underwater video sequences by using color and texture. Int. J. Sci. Eng. Res. 7, 1243–1249.

Simonyan, K., Zisserman, A., 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. https://doi.org/10.48550/ARXIV.1409.1556.

Strachan, N., 1993. Recognition of fish species by colour and shape. Image Vis. Comput. 11, 2–10. https://doi.org/10.1016/0262-8856(93)90027-E.

Tickner, D., Opperman, J.J., Abell, R., Acreman, M., Arthington, A.H., Bunn, S.E., Cooke, S.J., Dalton, J., Darwall, W., Edwards, G., Harrison, I., Hughes, K., Jones, T., Leclère, D., Lynch, A.J., Leonard, P., McClain, M.E., Muruven, D., Olden, J.D., Ormerod, S.J., Robinson, J., Tharme, R.E., Thieme, M., Tockner, K., Wright, M., Young, L., 2020. Bending the curve of global freshwater biodiversity loss: an emergency recovery plan. BioScience 70, 330–342. https://doi.org/10.1093/biosci/biaa002.

Tuhtan, J.A., Nag, S., Kruusmaa, M., 2020. Underwater bioinspired sensing: new opportunities to improve environmental monitoring. IEEE Instrum. Meas. Mag. 23, 30–36. https://doi.org/10.1109/MIM.2020.9062685.

Xia, J., Xuan, D., Tan, L., Xing, L., 2020. ResNet15: weather recognition on traffic road with deep convolutional neural network. Adv. Meteorol. 2020, 1–11. https://doi.org/10.1155/2020/6972826.

Xiao, H., Zhang, F., Shen, Z., Wu, K., Zhang, J., 2021. Classification of weather phenomenon from images by using deep convolutional neural network. Earth Space Sci. 8 https://doi.org/10.1029/2020ea001604.

Xie, K., Wei, Z., Huang, L., Qin, Q., Zhang, W., 2021. Graph convolutional networks with attention for multi-label weather recognition. Neural Comput. & Applic. https://doi.org/10.1007/s00521-020-05650-8.

Yang, Y., Elsinghorst, R., Martinez, J.J., Hou, H., Lu, J., Deng, Z.D., 2022. A real-time underwater acoustic telemetry receiver with edge computing for studying fish behavior and environmental sensing. IEEE Internet Things J. 1–1 https://doi.org/10.1109/JIOT.2022.3164092.

Zhang, T., Jin, P.J., 2019. A longitudinal scanline based vehicle trajectory reconstruction method for high-angle traffic video. Transp. Res. Part C Emerg. Technol. 103, 104–128. https://doi.org/10.1016/j.trc.2019.03.015.

Zhang, L., Li, W., Liu, C., Zhou, X., Duan, Q., 2020a. Automatic fish counting method using image density grading and local regression. Comput. Electron. Agric. 179 https://doi.org/10.1016/j.compag.2020.105844.

Zhang, S., Yang, X., Wang, Y., Zhao, Z., Liu, J., Liu, Y., Sun, C., Zhou, C., 2020b. Automatic fish population counting by machine vision and a hybrid deep neural network model. Animals 10, 364. https://doi.org/10.3390/ani10020364.

Zhao, S., Zhang, S., Liu, J., Wang, H., Zhu, J., Li, D., Zhao, R., 2021. Application of machine learning in intelligent fish aquaculture: a review. Aquaculture 540, 736724. https://doi.org/10.1016/j.aquaculture.2021.736724.

Appendix 2

II

J. Soom, M. Leier, K. Janson, and J. A. Tuhtan. Open urban mmwave radar and camera vehicle classification dataset for traffic monitoring. *IEEE Access*, 12:65128–65140, 2024

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 April 2024, accepted 22 April 2024, date of publication 6 May 2024, date of current version 14 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3397013

RESEARCH ARTICLE

Open Urban mmWave Radar and Camera Vehicle Classification Dataset for Traffic Monitoring

JÜRGEN SOOM[1,2], **MAIRO LEIER**[1], **KARL JANSON**[1],
AND JEFFREY A. TUHTAN[2], **(Member, IEEE)**
[1]Embedded AI Research Laboratory, Tallinn University of Technology, Harju County, 12618 Tallinn, Estonia
[2]Department of Computer Systems, Tallinn University of Technology, Harju County, 12618 Tallinn, Estonia

Corresponding author: Jürgen Soom (jurgen.soom@taltech.ee)

This work was supported by the European Union through European Social Fund through the Project IT Academy (IT Akadeemia in Estonian) (ICT) Program.

ABSTRACT Traffic monitoring systems featuring robust, multi-sensor fusion capabilities are rapidly growing in demand to observe traffic flow, reduce congestion and to detect and report traffic accidents. However, monitoring outdoor environments using cameras remains challenging due to complex weather conditions, including fog, rain, snow and variable lighting conditions. The presence of these weather conditions can significantly reduce vehicle detection and classification performance using machine learning methods. Unfortunately, openly available datasets for multi-sensor traffic monitoring development and testing remain limited, especially those featuring infrastructure-based cameras and millimeter wave (mmWave) radar. To address these challenges, we evaluate open camera and mmWave radar data using vehicle classification models for cars, trucks, vans and buses on embedded hardware. We also provide an open multi-sensor traffic monitoring dataset with more than 8,000 manually annotated frames as well as mmWave radar point clouds recorded in an urban environment under sunny, partially cloudy, cloudy, rainy and night conditions.

INDEX TERMS Object detection, edge computing, machine learning, camera, millimeter wave radar, traffic video.

I. INTRODUCTION

In 2020, the International Council on Clean Transportation (ICCT) reported 11.7 million new vehicle registrations in the 27 member states of the European Union and the United Kingdom. [1]. According to the report published by the European Automobile Manufacturers Association (ACEA), China alone had more than 25.5 million newly registered vehicles in 2022. From 2021 to 2022, the number of newly registered vehicles in India increased by 24.1% [2]. The Bureau of Transportation Statistics (BTS) 2021 survey shows that the USA currently has 275.9 million registered light-duty vehicles, motorcycles, trucks, and buses [3]. Analysts expect that the number of newly registered vehicles will continue to grow for the foreseeable future, reaching 2.21 billion worldwide by 2050 [4], [5]. With the growing number of vehicles around the world, developing and managing a city's transportation infrastructure has become a substantial and persistent challenge. Frequent problems include traffic jams, congestion, and environmental and noise pollution. To address these challenges, traffic monitoring systems are deployed to collect data about the traffic flow, including velocity, volume, Peak Hour Factor (PHF), density, headway, spacing, gap, and clearance [8]. This data is essential to improve the planning and development of transportation infrastructure. However, collecting the required information in outdoor environments still poses a major technology challenge, especially in the presence of fog, rain, snow, and during the night.

II. RELATED WORK

Sensors used for traffic monitoring applications are divided into three subcategories: in-roadway, side-roadway, and over-roadway [9]. The earliest traffic monitoring systems used in-roadway sensors and were primarily adapted for vehicle counting applications. Inductive loop detectors (ILD) monitor

The associate editor coordinating the review of this manuscript and approving it for publication was Xinyu Du.

2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

passing vehicles by detecting changes in inductance. ILD sensors have excellent detection accuracy, ranging from 92% [10], [11], [12], [13] up to 99% accuracy with magnetic sensors [14], [15]. However, in-road sensor systems have several drawbacks: high installation and maintenance costs, pavement damage during installation, and limited lane coverage. Furthermore, installation and maintenance can cause traffic disruptions and congestion, since the road section must be closed during these activities [9].

Non-intrusive, side- and over-roadway traffic monitoring systems are typically comprised of acoustic sensors, light detection and range (LiDAR), or radio detection and range (radar). A key advantage over in-roadway sensing is their ability to monitor multiple lanes simultaneously [16], [17], [18]. In addition, the sensor can be installed over the road or on the side of the road without damaging the pavement or requiring closing any traffic lanes during installation and maintenance. Furthermore, these sensors are less dependent on the intensity and variability of illumination. However, reflective surfaces can cause large local changes in illumination intensity, reducing the detection accuracy, and in some cases, false detections. Nonideal weather conditions including fog, rain, or snow can also scatter or absorb radio waves, reducing both the operating range and the accuracy [19].

The first attempts at camera-based vehicle counting date back to 1978 by the Jet Propulsion Laboratory [21]. Cameras have since become the most widely used type of sensor for traffic monitoring, mainly because they offer several advantages: they can cover multiple lanes simultaneously, have flexible mounting options and require minimal maintenance [9]. The ability to survey the surroundings in high spatial and temporal detail provides cameras with a significant advantage over in-roadway sensors and they can detect and classify vehicles with over 95% accuracy [22], [23], [24]. Camera observations can also be applied for additional traffic monitoring use cases, such as assessing road conditions, detecting collisions and assisting rescue services. Despite their numerous advantages, camera-based vehicle detection accuracy remains highly dependent on weather and local illumination conditions. Fog can reduce visibility by scattering light and reducing contrast, making objects appear hazy and lacking details. Rain and snow can cause droplets to accumulate on the lens or sensor, adversely impacting the image quality. Low light conditions may produce unwanted levels of noise and blurring. Conversely, cameras may provide overexposed imagery during localized bright lighting conditions caused by reflection glare and headlights [9], [19]. Table 1 provides a comparative overview of the influence of weather conditions by sensor type. Recently, multi-sensor-based traffic monitoring systems have begun to gain more traction [28], [29], [33], [34], [35]. The primary motivation behind this trend is to improve detection accuracy and mitigate the shortcomings in complex, changing environments and weather conditions [30], [31], [32]. One major obstacle in developing, testing, and validating multi-sensor-based traffic monitoring systems is the limited number of openly available traffic monitoring datasets.

TABLE 1. Impacts of different weather conditions on cameras, radar and LiDAR sensors used for traffic monitoring. A score for each of the weather conditions and its impact on the traffic monitoring sensor type ranges from 0: negligible effect, to 5: severe impact, and was adopted for each combination in the table based on the method used in [19].

Sensor type	Light rain[1]	Heavy rain[2]	Mist[3]	Fog[4]	Snow	Strong light
Camera	3	4	5	4	2/3	5
Radar (24, 77 and 122 GHz)	0	1	2	0	2	0
LiDAR (850-950nm)	2	3	5	4	5	2

[1] < 4mm/hr
[2] < 25mm/hr
[3] Visibility < 0.1km
[4] Visibility < 0.5km

The effect level each phenomenon causes to sensors:
0 - negligible: influences that can almost be ignored
1 - minor: rarely cause detection error
2 - slight: occasionally cause minor errors
3 - moderate: cause perception error up to 30% of the time
4 - serious: cause perception error more than 30% but lower than 50% of the time
5 - severe: noise or blockage that causes false detection or detection failure

A. OVERVIEW OF MULTI-SENSOR INFRASTRUCTURE-BASED TRAFFIC MONITORING DATASETS

DAIR-V2X-I contains over 10000 annotated frames collected using a high-resolution camera and LiDAR. The dataset features ten classes, focusing on diverse weather and lighting variations [36].

A9 dataset consists of footage using a high-resolution camera and LiDAR, covering a variety of traffic situations. The anonymized and precision-timestamped footage was recorded at the three km-long Providentia++ testfield near Munich, Germany. The dataset features a total of four weather conditions and six vehicle classes [37].

LUMPI features over 200k frames, collected over several days during different weather and light conditions at a large junction in Hanover, Germany. The dataset includes 2D image information (videos) and 3D point clouds with labels of the traffic participants in the scene [38].

AAU RainSnow dataset was collected using conventional RGB and thermal infrared cameras. It features scenes with rain and snowfall, captured from 22 five-minute videos from seven different traffic intersections. The illumination of the scenes varies from broad daylight to twilight and night. Scenes show glare from car headlights, reflections from puddles, and raindrop blur to the camera lens. In total, the data contains 2200 annotated frames [43].

Radar LAB created an automatic radar-camera dataset generation toolkit for sensor-fusion applications to minimize labor costs for recording and processing camera and radar data simultaneously. However, the dataset is not openly available [45].

UTIMR applies an urban traffic imaging using millimeter-wave radar system. Information about the features of the dataset is very limited and is not openly available [46].

TJRS TS focuses on trajectory tracking using millimeter-wave radar sensors. The verification data was captured with cameras attached to UAV [44]. The weather conditions and amount of data collected are not specified. The dataset is not open access, but is available upon request.

CIM (this work) complements existing open datasets by providing novel camera and mmWave radar data, covering multiple weather conditions, locations, and vehicle classes common to urban traffic monitoring locations. The specifics of the dataset are discussed in more detail in Section IV-A. A comparison of the CIM dataset with current openly available infrastructure-based multi-sensor traffic monitoring datasets is summarized in Table 2.

B. OBJECTIVES

The objectives of this work are two-fold: First, we provide a camera and radar-based vehicle detection and classification pipeline and evaluate its performance using embedded hardware. Second, we provide a new open infrastructure-based multi-sensor traffic dataset featuring nearly 8400 manually annotated frames, including five weather conditions and four vehicle classes. This article is organized as follows: In **Section III**, we provide a detailed overview of the selected embedded hardware options, camera sensor, and mmWave radar. The methods applied for camera and mmWave radar vehicle detection and classification are provided in **Section IV**. The results are provided in **Section V**. Finally, in **Sections VI and VII**, the advantages, limitations, and future research directions are discussed.

III. HARDWARE

A. EMBEDDED HARDWARE

Two embedded hardware configurations were chosen to evaluate the proposed vehicle detection and classification pipeline, and a summary of the hardware specifications is provided in Table 3. The author's first choice was the Nvidia Jetson Nano, which represents a typical platform for AI and machine learning applications. The Jetson Nano hardware features a Quad-core Cortex-A57 microprocessor control unit (MCU), 128-core Maxwell architecture-based graphics processing unit (GPU), 4GB of LPDDR4 and can support MIPI CSI-2 cameras. Although the board lacks built-in FLASH memory, the hardware supports flash storage devices. The second choice was the Nvidia Jetson Orin Nano 4GB version, emulated using the Jetson AGX Orin Developer Kit, chosen as it is the successor to the Nvidia Xavier platform. The board features a Hexa-core MCU based on Cortex-A78AE architecture running at 1.5 GHz. In addition, the Orin Nano board has a 512-core GPU based on the Ampere architecture with 16 dedicated Tensor Cores. Both systems support software-based power consumption monitoring, which was used in this work to evaluate their efficiency. The selection of the Jetson platform by the authors can be attributed to two primary factors. First, the Nvidia Jetson platform is a widely recognized and extensively documented commercial device that has gained widespread adoption in both academic and commercial research communities. Secondly, the worldwide chip shortage has put alternative platforms, such as Raspberry Pi, in short supply for many users, making the Jetson platform the most viable option for the authors during the time this work was carried out.

B. RADAR

Unlike optical sensors, such as cameras or LiDAR, radar sensors use radio waves to detect objects. By measuring the differences between arrival times and the phase shift of the radio wave signals reflected from an object's surfaces, radar sensors estimate the distance and speed of the target. Automotive millimeter-wave (mmWave) radars do not produce a high-resolution three-dimensional scan of the environment like LiDAR. Instead, they have a more limited output of some 200 measurement points. In this work, the Texas Instruments AWR1843BOOST mmWave radar development board [53] was implemented. The kits are based on the Texas Instruments AWR1843 automotive 77GHz radar sensor. The development board also includes a signal processor to translate the analog radar data into digital radar point clouds.

1) RADAR CONFIGURATION

The radar used in this work can be configured using multiple parameters to meet the needs of the selected application. Finding the best combination of these parameters depends highly on the application, and the parameters used in this work are provided in Table 4. These parameters in Table 4 were optimized from field data testing as part of the Advanced Traffic Sensor Pilot project conducted by the Embedded AI Research Lab at Tallinn University of Technology in cooperation with Thinnect OÜ in 2021 and 2022. The configured range resolution of the radar is 0.59 m, and the velocity resolution is 1.3 km/h, with a maximum measurable velocity of 83 km/h. This resolution was achieved for objects at a maximal distance of 30 m. However, the pilot project experiments indicated that the radar can feasibly recognize objects at distances up to 100 m. The radar uses a relatively narrow beam azimuth of 15°, transmitting measurement points 15 times per second. Measurements from stationary objects were filtered to reduce clutter and simplify processing. The filter subtracts the mean value of the input samples after applying a two-dimensional Fast Fourier Transform [20].

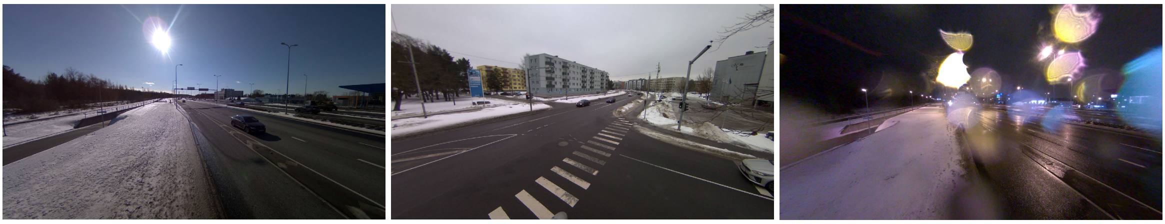
C. CAMERA

Compared to radar and LiDAR, camera sensors are able to record multiple features such as color, shape, and luminance, which is advantageous for object detection and classification applications. However, camera performance can be negatively impacted by commonly-occurring weather and environmental conditions such as rain, fog, snow and low lighting at night [19]. In this work, a Sony IMX-219-120

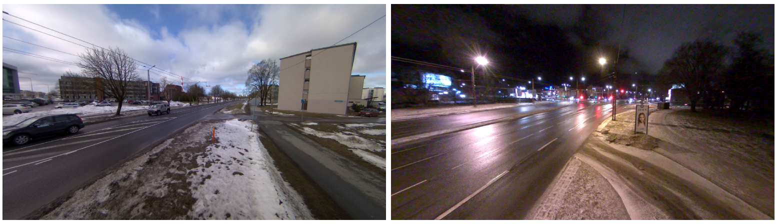
TABLE 2. Comparison of open multi-sensor infrastructure-based datasets for vehicle detection and classification. CIM (this work) provides the largest open dataset for camera and mmWave radar to-date.

Name	Content	Resolution	Weather conditions	Vehicle classes	Sensors	Availability
DAIR-V2X-I [36]	10084	1920x1080	Sunny Cloudy Nighttime Rain	Passenger car Truck Bus Van Motorcycle	Camera LiDAR	Open
A9 [37]	1098	1920x1200	Cloudy Snow Fog Sunny	Passenger car Truck Van Bus Motorcycle Trailer	Camera LiDAR	Open
LUMPI [38]	200k	1640x1232 1920x1080	Sunny Cloudy Night Rain	Passenger car Truck Van Bus Motorcycle Trailer	Camera LiDAR	Open
Rope3D [39]	50k	1920x1080	Clear Rain Night Dawn/Dusk	Passenger car Motorcycle Van Bus Truck Bicycle Tricycle Barrow	Camera LiDAR	Open
IPS300+ [42]	14198	1920x1080	N/A	Passenger car Bicycle Tricycle Bus Truck Engineer Car	Camera LiDAR	Open
RainSnow [43]	2200	640x480	Snow Rain Night Blizzard	Passenger car Bus Truck Van	Camera Thermal Camera	Open
TJRD TS [44]	N/A	N/A	N/A	Passenger car Bus Truck Van	Camera mmWave Radar	On request
Radar LAB [45]	8035	N/A	Clear Partially-cloudy	Passenger car	Camera Radar	Not available
UTIMR [46]	N/A	N/A	N/A	Small car[1] Medium car[1] Large car[1]	Camera Radar	Not available
CIM (this work)	8393	1920x1080	Sunny Partially cloudy Rain/Sleet Cloudy Night	Passenger car Van Truck Bus	Camera mmWave Radar	Open

[1] Vehicles are classified by length: small car (L < 4.3 m), medium-sized car (4.3 m < L < 7 m), and large bus (L > 8 m).



(a) Clear - Järvevana tee (b) Cloudy - Akadeemia tee (Raja junction) (c) Rain - Järvevana tee



(d) Partially cloudy - (Fujitsy sign) (e) Night - Kristiine (Intersection)

FIGURE 1. Examples from recording locations with different weather conditions. (a) Clear, ideal conditions of the roadway and vehicles. (b) Cloudy conditions, where some regions of the roadway have poor illumination at a distance. (c) Rain and other non-ideal conditions in which the camera lens may have water droplets and where sections of the roadway may have blurred imagery. (d) Partially cloudy, dynamic changes in near and far-field illumination occur on the roadway due to variations in cloud cover. (e) Night, considerable variability in the roadway illumination levels due to static street lighting in conjunction with automobile head and tail lights. Computer vision approaches for detection and classification remain challenging because at some locations different illumination and weather conditions may co-occur and vary substantially between frames.

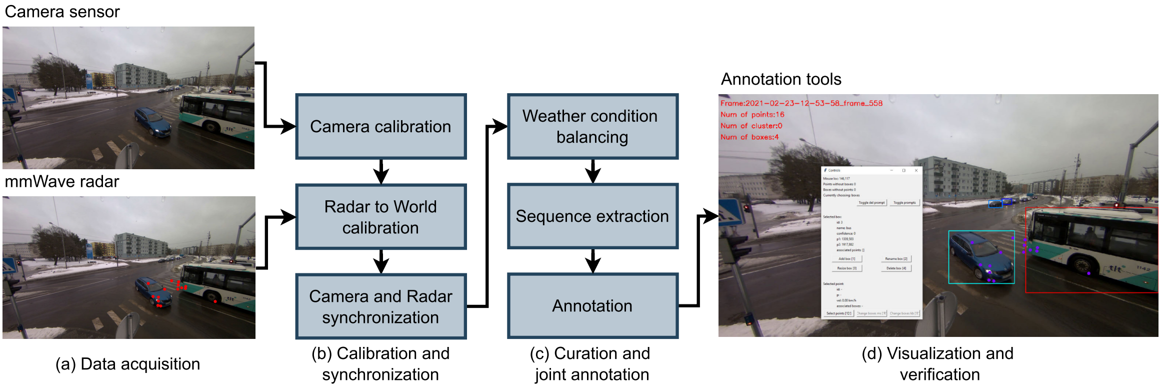

Camera sensor
mmWave radar
(a) Data acquisition
Camera calibration
Radar to World calibration
Camera and Radar synchronization
(b) Calibration and synchronization
Weather condition balancing
Sequence extraction
Annotation
(c) Curation and joint annotation
Annotation tools
Frame:2021-02-23-12-53-58_frame_558
Num of points:16
Num of cluster:0
Num of boxes:4
(d) Visualization and verification

FIGURE 2. General overview of the CIM dataset curation workflow: First, the curator selects and undistorts the camera imagery. The point cloud data is then converted from 3D to the camera's 2D reference frame, as discussed in Section IV-C. Afterwards, the camera footage and radar data were synchronized. Next, the videos are sorted and balanced based on the weather and local illumination conditions following the National Oceanic and Atmospheric Administration (NOAA) guidelines [49]. After sorting, cleaning, and adjusting the recorded footage, the curator randomly selects and extracts a 100-frame sequence from each video. The sequence is then manually annotated for each vehicle within a frame using rectangular bounding boxes.

camera [52] was chosen primarily due to its compatibility with existing embedded hardware platforms and its wide angle of view. Table 5 provides an overview of the camera sensor specifications.

IV. MATERIALS AND METHODS

A. DATASET

Critical Infrastructure Monitoring (CIM, this work) is a new dataset collected in the Tallinn urban environment during the late winter and early spring, covering most weather and environmental conditions common to temperate and sub-polar regions. Fig. 1 provides an overview and examples of the weather conditions and recording locations. The original recordings feature over 41 hours of footage at a resolution of 3264 × 2464 px at 15 fps, which was fixed due to radar data acquisition limitations. The footage was undistorted and cropped to a resolution to 1920 × 1080 px. The CIM data curation workflow is depicted in Fig. 2. The final curated dataset contains 8393 manually annotated frames, including

TABLE 3. **Overview of the Jetson Orin Nano and Jetson Nano hardware specifications.**

Component/Feature	Nvidia Jetson Orin Nano [50]	Nvidia Jetson Nano [51]
Microprocessor	Hexa-core ARM A78AE @ 1.5 GHz	Quad-core ARM A57 @ 1.43GHz
Graphics processing unit	512-core Ampere with 16 Tensor cores	128-core Maxwell
Memory	4 GB LPDDR5	4 GB LPDDR4
Storage	MicroSD / NVMe	MicroSD
Camera	2x MIPI CSI-2 connectors	2x MIPI CSI-2 connectors
Network	Wi-Fi (802.11ac), GbE	GbE

TABLE 4. **Summary of the AWR1843BOOST mWave radar hardware configuration used in this work, including brief descriptions of each parameter.**

Parameter	Value	Description
Frame-rate	15	Number of measurements per second
Azimuth	15°	Horizontal angle of the radar beam
Range resolution	0.586 m	Maximum range measurement error
Max unambiguous range	30 m	Maximum range that radar can accurately measure
Velocity resolution	1.33 km/h	Maximum velocity measurement error
Max velocity	82.98 km/h	Maximum velocity radar can accurately measure
Clutter removal	Enabled	Removes radar reflections from stationary objects

TABLE 5. **Summary of the Sony IMX-219-120 camera sensor specifications.**

Parameter	Value
Resolution	3280 x 2464
Aperture (F)	2.2
Focal Length	1.79 mm
Angle of View (diagonal)	120°
Distortion	< 13.6%

TABLE 6. **Label counts per vehicle class in each recording location. The largest number of samples belong to Fujitsy sign, featuring 14294 samples for passenger car, 382 for bus, 685 for van, and 289 for truck class.**

Location	Car	Bus	Van	Truck
Akadeemia tee (Raja junction)	6438	170	50	187
Fujitsy sign	14294	382	685	289
Järvevana tee	2729	14	750	198
Kristiine (Intersection)	7471	416	571	383
Total	**30932**	**982**	**2056**	**1057**

radar point cloud data (Table 2). The distribution of sample count across all the vehicle classes and the number of samples for each vehicle class captured at each location is shown in Table 6. Each frame in the CIM dataset is represented using XML and JSON files. The XML file contains information about the vehicle class, followed by bounding box data, using (X_{min},Y_{min},X_{max},Y_{max}) formatting. The contents of the JSON and the fields in each measurement point object are documented in Table 7. CIM is a openly available under Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License [48]. The dataset is hosted via Zenodo: 10.5281/zenodo.8301276

TABLE 7. **Description of the JSON fields for radar point objects included in the open CIM dataset.**

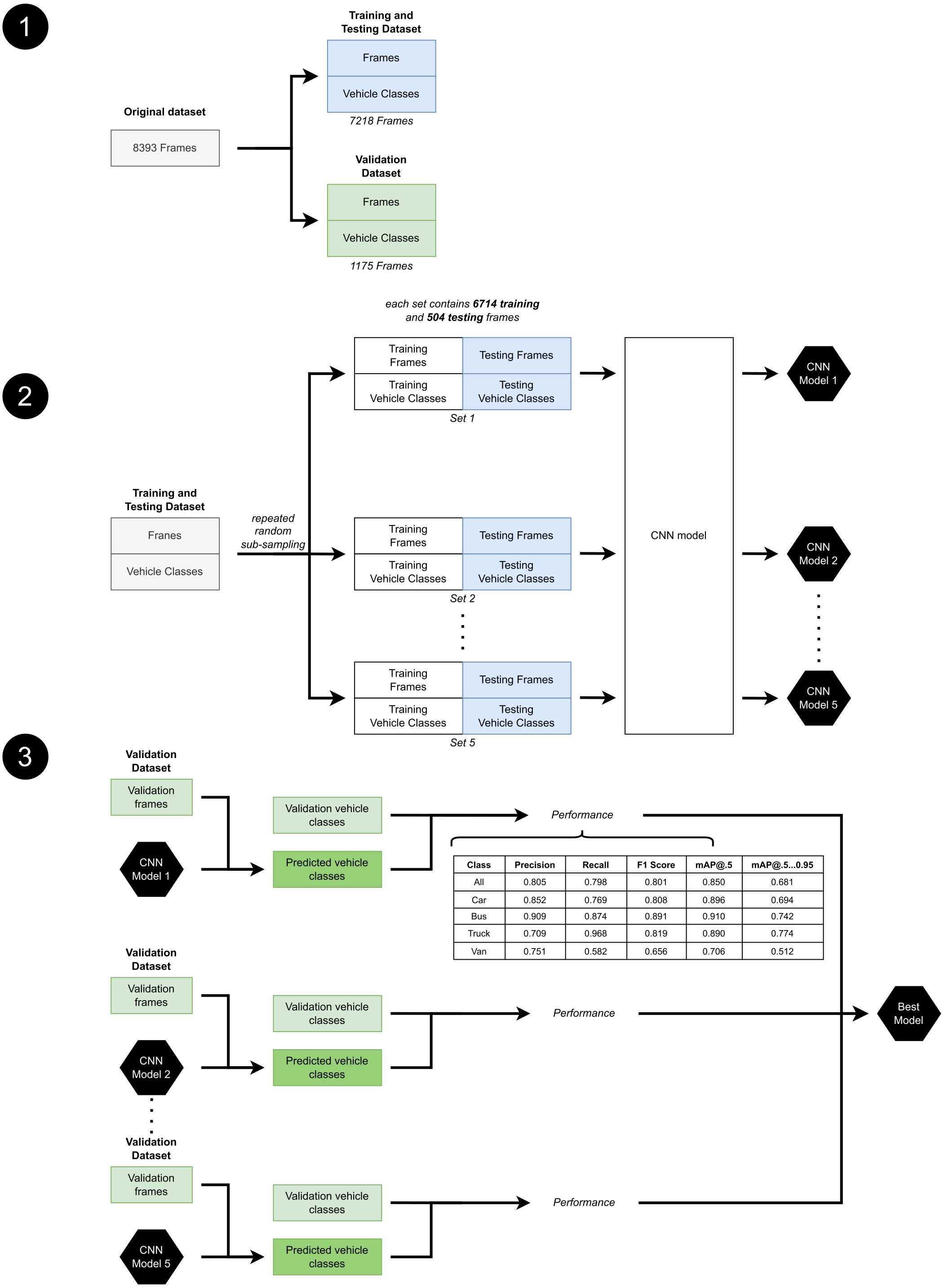
Field Name	Description
x	Measurement point coordinate in 3D space, relative to the radar (X component)
y	Measurement point coordinate in 3D space, relative to the radar (Y component)
z	Measurement point coordinate in 3D space, relative to the radar (Z component)
velocity	Speed at the measurement point. The value is given in km/h.
screen_coords	Pixel coordinates radar measurement point in camera's coordinate system (location of the point in camera image)
is_line	Specifies if the measurement point's coordinates are given as a line (always set to False in this work)
corresponding_box	ID of the object in the image the point is associated with. Defaults to -1 (no object)

B. CAMERA-BASED VEHICLE DETECTION

In order to detect and classify vehicles using the camera system, we trained an object detection model based on YOLOv7 architecture [55]. Several CNN architectures were considered, such as SSDLite [25] and ResNet [26]. Ultimately, the YOLOv7 architecture was chosen based on the Average precision (AP) and inference time ratio. The model was trained to recognize four vehicle classes: passenger car, bus, truck, and van. The vehicle classification model was developed using a two-step process. As illustrated in Fig. 3, five randomly subsampled models were trained, tested and validated. The CIM dataset contains a total of 8393 frames, and was split into separate datasets for training and validation. The training dataset used for training and testing included 7218 labeled frames, leaving 1175 labeled frames for the validation dataset. The 7218 frames were further divided into five datasets for training and testing using random subsampling (see Fig. 3 (2)). Each of the five datasets contained 6714 frames for training and 504 for testing. All five models were then trained over a total of 50 epochs. In order to select the best-performing model, the five trained models were validated using the same hold-out data in the final stage (see Fig. 3 (3)). The results of the validation phase are further discussed in Section V-A.

C. RADAR-BASED VEHICLE CLASSIFICATION

A radar-based vehicle classifications model was developed using the mmWave radar point cloud coordinates converted from the sensor frame of reference using the extrinsic and intrinsic parameters of the camera. The extrinsic parameters



Class	Precision	Recall	F1 Score	mAP@.5	mAP@.5...0.95
All	0.805	0.798	0.801	0.850	0.681
Car	0.852	0.769	0.808	0.896	0.694
Bus	0.909	0.874	0.891	0.910	0.742
Truck	0.709	0.968	0.819	0.890	0.774
Van	0.751	0.582	0.656	0.706	0.512

FIGURE 3. **1) Illustration of the hold-out procedure for training, testing, and validating the vehicle detection model. 2) Usage of repeated sub-sampling was applied for testing and training and was used to train models with identical CNN architecture. 3) Finding the best-performing model using the validation dataset, adapted from [27].**

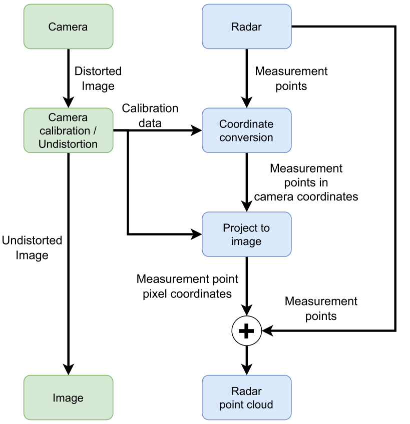


FIGURE 4. Workflow used to calibrate and align the camera pixel locations with the mmWave radar point cloud.

allow conversion between the camera and the world coordinate system. Extrinsic parameters are used to rotate and translate the radar point cloud coordinates to match the camera coordinate system. Intrinsic parameters represent the internal properties of the camera sensor including lens distortion. An overview of the coordinate conversion process is provided in Fig. 4, and can be broken down into the following four processing steps:

1) Calibration is carried out to retrieve the camera's intrinsic and extrinsic parameters using tools from the robot operating system (ROS 2).
2) Coordinate conversion from the radar point cloud reference frame to the camera as the reference system by rotating and translating the radar point cloud into the camera's coordinate system.
3) Lens distortion effects are reduced using the intrinsic parameters obtained during the camera calibration stage.
4) The radar point cloud is projected from the 3D space into the camera's 2D image plane. This results in a common reference frame for both the camera pixel coordinates and the radar point cloud data.

Before the mmWave radar vehicle classification model could be trained, we first generated a labeled point-cloud dataset. As described in Table 7, the raw radar point cloud information includes a bounding box ID, which was used to match the points in each box to a corresponding vehicle label taken from the camera system. This was done manually for each frame by extracting the vehicle class and the bounding box coordinates $(x_{min}, y_{min}, x_{max}, y_{max})$. Afterwards, we stored the number of points for each radar bounding box with the corresponding vehicle label. The outcome of the point cloud annotation process resulted in a dataset featuring 423 unique vehicle bounding box samples, summarized in Table 8. Due to the low number of truck and bus samples obtained by the radar, the truck and bus classes were merged into a single class corresponding to large non-car vehicles. The radar dataset was divided into training and validation datasets using a ratio of 80:20, leaving 338 samples for training and 85 samples for validation. The validation dataset was checked to feature equal amount of samples labeled as *car* and *truck/bus*. Three distinct approaches were examined and assessed in an effort to classify vehicle types, relying only on the radar point cloud information: Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Feedforward Neural Network (FNN).

TABLE 8. Example datasets used for the mmWave radar based classification. Each sample contains the number of radar points, the coordinates, the point velocity and the corresponding vehicle class label.

#	Number of points	Radar point 2D coordinate pairs	Velocity (km/h)	Vehicle Class Label
1	6	[(867, 519) ... (910, 514)]	8.952	Car
2	3	[(970, 553), (1010, 556), (949, 554)]	6.445	Car
3	8	[(1219, 589) ... (1378, 574)]	0.716	Truck/Bus
.....				
423	23	[(1244, 561) ... (1244, 552)]	2.865	Truck/Bus

SVM is one of the most commonly used supervised machine learning algorithms for classifying, regressing, or detecting outliers. The algorithm works by finding a hyperplane, separating the data points of one class from the other. Maximizing the distance between classes in a multidimensional space. These characteristics have enabled researchers to accurately distinguish vehicles, pedestrians, and other objects from point cloud information obtained using mmWave radar [17], [58].

The proposed SVM model utilizes a radial basis function (RBF) kernel. The RBF kernel measures the similarity between two data points in infinite dimensions. The kernel function is defined as:

$$K(x_1, x_2) = \exp(-\gamma \cdot \|x_1 - x_2\|^2) \tag{1}$$

where γ controls the 'spread' of the kernel. The closer the kernel function is to zero, the larger the Euclidean distance between two points $\|x_1 - x_2\|^2$. The larger the distance between the two points, the more likely they are dissimilar [61].

Neural networks are a category of machine learning algorithms renowned for their capacity to discern intricate patterns and relationships within data, solidifying their position as one of the most widely employed methods for addressing classification and regression tasks. These networks have been effectively deployed to perform object classification using exclusively point cloud data, enabling more streamlined and precise object categorization without reliance on conventional image-based information [59], [60].

The designed FNN model incorporates two hidden layers. The first hidden layer contains eight units, while the second only features four. Using dense layers, each individual neuron is connected to every neuron in the subsequent layer, creating a densely connected network structure. With the exception of the output layer, which employs a softmax activation

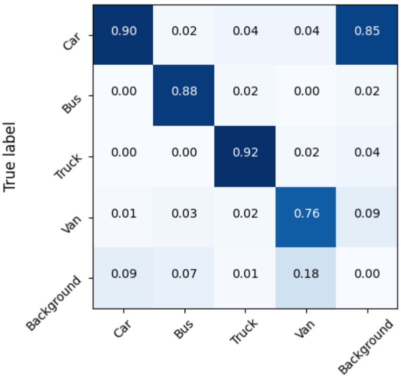


FIGURE 5. Confusion matrix of best performing camera classification model (Model 3).

function, all layers utilize the ReLu activation function. Additionally, dropout regularization was implemented to avoid overfitting the network.

KNN is a simple yet highly effective algorithm, which operating on the principle that similar data points are in close proximity to each other in the feature space. It looks at the K closest data points and assigns the new point the majority class (for classification). To find the optimal K value, we utilized the elbow method [56], [57]. The elbow method involves plotting the error rate of the KNN model as a function of different K values.

The results of the radar-based vehicle classification models are further discussed in Section V-A.

V. RESULTS

A. VEHICLE DETECTION AND CLASSIFICATION USING CAMERA

To ensure a robust evaluation of the CNN model for camera-based vehicle classification, a hold-out validation dataset containing 1175 frames was used. As described in Section IV-B and visualized in Fig 3, the curated dataset was split using a ratio of 70:30. The first portion of the split dataset was used to train and test the model, using a randomizer to generate a new training and testing data for each run. The remaining portion of the camera dataset was set aside to validate the performance and to determine the best performing model. Table 9 summarizes the CNN model performance across all vehicle classes, using the evaluation parameters described in the previous section. Of the five trained models, Model 3 performed best, albeit by a small margin, achieving the highest mAP@0.5…0.95 score of 0.681 across all vehicle classes. The confusion matrix of Model 3 is depicted in Fig. 5.

B. VEHICLE DETECTION AND CLASSIFICATION USING RADAR

Three separate approaches were tested to classify vehicle types using only information collected by the mmWave radar. The results were validated using a hold-out dataset featuring 50 samples, of which 25 belong to passenger cars and 25 to bus/truck class, to validate the performance of the radar-based classification model. The evaluation outcomes are summarized in Table 10.

TABLE 9. Performance summary after validation for each of the five trained models for all vehicle classes, and for all classes combined. Model 3 was chosen as the overall best model based on the mAP@0.5…0.95 evaluation parameter.

Model	Class	Precision	Recall	F1 score	mAP@0.5	mAP@0.5...0.95
Model 1	All	0.793	0.786	0.789	0.834	0.642
	Car	0.825	0.769	0.796	0.867	0.650
	Bus	0.855	0.851	0.853	0.890	0.703
	Truck	0.747	0.952	0.837	0.905	0.759
	Van	0.744	0.572	0.647	0.675	0.457
Model 2	All	0.783	0.708	0.748	0.78	0.581
	Car	0.846	0.724	0.780	0.861	0.641
	Bus	0.840	0.712	0.771	0.799	0.568
	Truck	0.735	0.895	0.807	0.868	0.717
	Van	0.711	0.501	0.588	0.591	0.400
Model 3	All	0.805	0.798	0.801	0.850	0.681
	Car	0.852	0.769	0.808	0.896	0.694
	Bus	0.909	0.874	0.891	0.910	0.742
	Truck	0.709	0.968	0.819	0.890	0.774
	Van	0.751	0.582	0.656	0.706	0.512
Model 4	All	0.74	0.81	0.773	0.823	0.637
	Car	0.792	0.812	0.802	0.874	0.667
	Bus	0.796	0.792	0.794	0.848	0.676
	Truck	0.733	0.975	0.837	0.917	0.765
	Van	0.637	0.659	0.648	0.651	0.441
Model 5	All	0.681	0.651	0.665	0.679	0.466
	Car	0.753	0.732	0.742	0.803	0.556
	Bus	0.829	0.560	0.668	0.696	0.466
	Truck	0.633	0.779	0.698	0.754	0.560
	Van	0.508	0.533	0.520	0.463	0.276

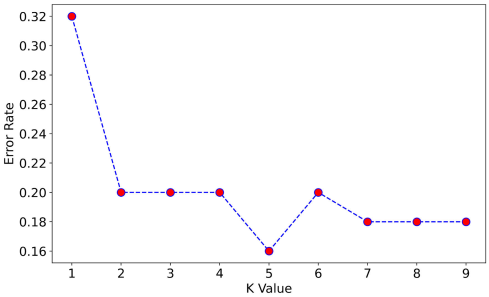


FIGURE 6. The optimal K value was determined using Elbow method. A K value of 5 resulted in the lowest error rate of 0.16.

The Feedforward Neural Network (FNN) exhibited the highest F1 Score for the *Car* category at 0.86. However, the method struggled to classify the *Bus/Truck* class, yielding an F1 score of 0.67. In contrast, the Support Vector Machine (SVM) demonstrated respectable performance, achieving F1 scores of 0.75 for *Car* and 0.7 for *Bus/Truck*. The K-Nearest Neighbors (KNN) performed the best of the three tested approaches. As described in Section IV-C, we used the elbow method to determine the optimal K value. As depicted in Fig. 6, the lowest error rate of 0.16 was achieved using a K value of 5. The KNN succeeded in attaining an F1 score exceeding 0.8 for both classes. The performance of the KNN model is visually depicted in the confusion matrix in Fig. 7.

TABLE 10. Classification result of the mmWave radar various methods. Out of the four tested method, KNN performed best, achieving an F1 score of 0.85 classifying cars and 0.83 for the bus/truck class.

Method	Vehicle class	Precision	Recall	F1 score
SVM	Car	0.76	0.74	0.75
	Bus/Truck	0.69	0.71	0.70
FNN	Car	0.83	0.90	0.86
	Bus/Truck	0.74	0.61	0.67
KNN	Car	0.79	0.92	0.85
	Bus/Truck	0.90	0.76	0.83

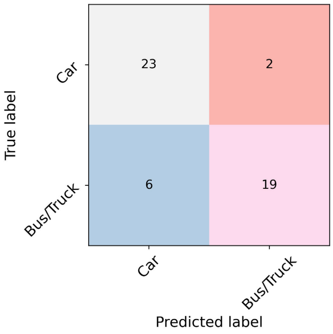


FIGURE 7. Radar point cloud classification model confusion matrix using the KNN. Out of 50 samples, 23 where correctly classified as Car and 19 as Bus/Truck, only 8 samples were incorrectly classified.

TABLE 11. Embedded hardware performance comparison when running the camera-based (YOLOv7) and mmWave radar (KNN) vehicle classification models.

Camera		
Evaluation parameter	Nvidia Jetson Orin Nano	Nvidia Jetson Nano
Frames per second	20	2 40
Power consumption (W)	8.9	4.9
Performance per Watt	2.25	0.41
mmWave Radar		
Evaluation parameter	Nvidia Jetson Orin Nano	Nvidia Jetson Nano
Frames per second	303	119
Power consumption (W)	5.8	3.6
Performance per Watt	52.24	39.67

C. HARDWARE COMPARISON–EFFICIENCY

As described in Section III-A, we evaluated the vehicle classification pipeline using two embedded hardware devices, the Jetson Nano and Jetson Orin Nano. A comparison of the results is provided in Table 11), where it was found that the Orin Nano performed the best overall, reaching up to 20 fps, while the Nano performed only 2 fps. Although the Nano (4.9 W) consumed around 4 watts less power than the Orin Nano (8.9 W), the Orin Nano (2.25 PPW) provided the highest overall performance-per-watt ratio relative to the Nano (0.41 PPW).

VI. DISCUSSION

The major contribution of this work is CIM, the open infrastructure-based multi-sensor traffic monitoring dataset. The primary motivation behind creating CIM was the limited availability of open annotated datasets featuring mmWave radar and camera imagery. Most open datasets, such as DAIR-V2X-I [36], A9 [37], LUMPI [38] and Rope3D [39] (summarized in Table 2) are based on LiDAR and camera systems. CIM, the dataset provided in this work exhibits several important features for real-world testing including multiple weather conditions and vehicle classes and features high image resolution and annotated, synchronized mmWave point clouds and corresponding vehicle labels. Datasets such as TJRD TS, Radar LAB, and UTIMR have limited availability, which CIM specifically addresses. As with all field datasets, CIM does have some shortcomings. The point cloud sample sizes are relatively sparse and are limited to cars and a limited number of buses and trucks due to the vehicle types passing during our field data collection campaign.

To mitigate this issue to the greatest extent, future studies using mmWave radar and infrastructure-mounted cameras could collect footage from a busy intersection during rush hour, allowing for the capture of more vehicle types within a single frame. Furthermore, a larger diversification of recording locations can help balance the sample count for each vehicle class. For example, the inclusion of industrial regions to collect more van and truck samples.

The second contribution of this work was to train and evaluate the performance of camera-based and mmWave radar classification in complex environments and weather conditions on embedded hardware. Weather conditions provided in the CIM dataset include overexposure from the headlights in a low-luminosity environment with low-contrast regions, which may lead to false negatives. Fog can reduce visibility by scattering light and reducing contrast, making objects appear hazy and lacking details. Rain and snow can cause droplets (see Fig. 1 (c)) to accumulate on the lens or sensor, affecting image clarity or suffering from clipping, and reducing detail in overexposed situations, increasing the amount of incorrect classifications, as well as the number of false negatives. Previous works have shown that combining multiple sensors can be effective [28], [29], [30], [31]. A recent work published by researchers at Guilin University of Electronic Technology [32] showed promising results. Using a camera and a mmWave radar, they achieved an average accuracy of 95.3% for vehicle detection. Expanding on this finding, we were able to show that our mmWave radar classification model with a few point cloud samples using KNN resulted in a minimum F1 score of 0.83 for two vehicle classes. Furthermore, the KNN managed to outperform both the SVM and FNN based classification models. However, the camera-based system was able to detect and classify cars, buses, trucks and vans with similar performance, with the best performing model obtaining an F1 score 0.805 for all classes combined.

Considering processing efficiency, one of the chosen embedded hardware options was able to process the camera data at a minimum rate of 20 fps, which far exceeded the recommended minimum of 15 fps. In situations and at locations where restricted access to a reliable power supply is a limiting factor, we also show that embedded hardware can provide viable real-world solutions for vehicle detection under non-ideal weather conditions. Future work based on the open CIM dataset can be conducted by developing, testing and validating novel vehicle classification pipelines fusing the mmWave radar and infrastructure-mounted camera systems.

VII. CONCLUSION

We show how synchronized camera and mmWave radar traffic monitoring sensors can be applied for complimentary vehicle detection and classification on embedded hardware. In addition, we provide CIM, an open infrastructure-based camera and mmWave radar traffic monitoring dataset featuring several weather conditions and vehicle classes. The results show promise for future work related to multi-sensor classification systems using stationary camera and mmWave radar for traffic monitoring. Both models were found to be suitable for low-cost embedded hardware running in real-time. Future work will focus on expanding the open CIM dataset by investigating the use of a cascaded mmWave radar system to improve data collection in urban settings with multiple weather and environmental conditions. Further research is needed when combining camera and mmWave radar traffic monitoring systems, considering that computational performance and vehicle classification accuracy may be increased by including an additional dedicated computing accelerator to the edge hardware used in this work.

VIII. DECLARATION OF COMPETING INTEREST

The authors declare that there are no competing interests.

ACKNOWLEDGMENT

The authors wish to thank Henry Juhanson and Uljana Reinsalu from the Embedded AI Research Laboratory, Tallinn University of Technology, for their assistance in collecting the CIM dataset.

REFERENCES

[1] S. Diaz, M. R. Bernard, Y. Bernard, G. Bieker, K. Lee, P. Mock, E. Mulholland, P. Ragon, F. Rodriguez, U. Tietge, and S. Wappelhorst, "European vehicle market statistics," ICCT, Washington, DC, USA, Tech. Rep., Dec. 2021, Accessed: Aug. 18, 2022. [Online]. Available: https://theicct.org/publication/european-vehicle-market-statistics-2021-2022/

[2] ACEA. (Jul. 18, 2022). *New Passenger Car Registrations and Annual GDP Growth in the EU*. Accessed: Oct. 3, 2023. [Online]. Available: https://www.acea.auto/figure/world-new-motor-vehicle-registrations-in-units/

[3] Bureau of Transportation Statistics. (Jan. 1, 2021). *National Transportation Statistics*. Accessed: Aug. 18, 2022. [Online]. Available: https://www.bts.gov/content/number-us-aircraft-vehicles-vessels-and-other-conveyances

[4] *US Energy Information Administration*. Accessed: Aug. 18, 2022. [Online]. Available: https://www.eia.gov/todayinenergy/detail.php?id=50096

[5] ACEA. (Jul. 18, 2022). *New Passenger Car Registrations and Annual GDP Growth in the EU*. Accessed: Aug. 19, 2022. [Online]. Available: https://www.acea.auto/figure/new-passenger-car-registrations-and-annual-gdp-growth-in-the-eu/

[6] J. Arts, W. Leendertse, and T. Tillema, "Road infrastructure: Planning, impact, and management," *Int. Encyclopedia Transp.*, vols. 1–7, pp. 360–372, May 2021, doi: 10.1016/b978-0-08-102671-7.10448-8.

[7] H. J. Shatz, K. E. Kitchens, S. Rosenbloom, and M. Wachs, "Highway infrastructure and the economy: Implications for federal policy," RAND Corp., Santa Monica, CA, USA, Tech. Rep., 2011. [Online]. Available: https://www.rand.org/pubs/monographs/MG1049.html

[8] *University of Memphis*. Accessed: Jun. 26, 2023. [Online]. Available: http://www.ce.memphis.edu/4162/L1_Traffic_Flow_Parameters.pdf

[9] M. Won, "Intelligent traffic monitoring systems for vehicle classification: A survey," *IEEE Access*, vol. 8, pp. 73340–73358, 2020, doi: 10.1109/ACCESS.2020.2987634.

[10] Y.-K. Ki and D.-K. Baik, "Vehicle-classification algorithm for single-loop detectors using neural networks," *IEEE Trans. Veh. Technol.*, vol. 55, no. 6, pp. 1704–1711, Nov. 2006, doi: 10.1109/TVT.2006.883726.

[11] S. Meta and M. G. Cinsdikici, "Vehicle-classification algorithm based on component analysis for single-loop inductive detector," *IEEE Trans. Veh. Technol.*, vol. 59, no. 6, pp. 2795–2805, Jul. 2010, doi: 10.1109/TVT.2010.2049756.

[12] H. A. Oliveira, F. R. Barbosa, O. M. Almeida, and A. P. S. Braga, "A vehicle classification based on inductive loop detectors using artificial neural networks," in *Proc. 9th IEEE/IAS Int. Conf. Ind. Appl. INDUSCON*, Sao Paulo, Brazil, Nov. 2010, pp. 1–6, doi: 10.1109/INDUSCON.2010.5740079.

[13] J. Gajda, R. Sroka, M. Stencel, A. Wajda, and T. Zeglen, "A vehicle classification based on inductive loop detectors," in *Proc. 18th IEEE Instrum. Meas. Technol. Conf. Rediscovering Meas. Age Informat.*, Budapest, Hungary, 2001, pp. 460–464, doi: 10.1109/IMTC.2001.928860.

[14] B. Yang and Y. Lei, "Vehicle detection and classification for low-speed congested traffic with anisotropic magnetoresistive sensor," *IEEE Sensors J.*, vol. 15, no. 2, pp. 1132–1138, Feb. 2015, doi: 10.1109/JSEN.2014.2359014.

[15] S. Y. Cheung, S. Coleri, B. Dundar, S. Ganesh, C.-W. Tan, and P. Varaiya, "Traffic measurement and vehicle classification with single magnetic sensor," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1917, no. 1, pp. 173–181, Jan. 2005, doi: 10.1177/0361198105191700119.

[16] H.-S. Lim, H.-M. Park, J.-E. Lee, Y.-H. Kim, and S. Lee, "Lane-by-lane traffic monitoring using 24.1 GHz FMCW radar system," *IEEE Access*, vol. 9, pp. 14677–14687, 2021, doi: 10.1109/ACCESS.2021.3052876.

[17] Z. Zhao, Y. Song, F. Cui, J. Zhu, C. Song, Z. Xu, and K. Ding, "Point cloud features-based kernel SVM for human-vehicle classification in millimeter wave radar," *IEEE Access*, vol. 8, pp. 26012–26021, 2020, doi: 10.1109/ACCESS.2020.2970533.

[18] J. Zhang, W. Xiao, B. Coifman, and J. P. Mills, "Vehicle tracking and speed estimation from roadside LiDAR," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 5597–5608, 2020, doi: 10.1109/JSTARS.2020.3024921.

[19] Y. Zhang, A. Carballo, H. Yang, and K. Takeda, "Perception and sensing for autonomous vehicles under adverse weather conditions: A survey," *ISPRS J. Photogramm. Remote Sens.*, vol. 196, pp. 146–177, Feb. 2023, doi: 10.1016/j.isprsjprs.2022.12.021.

[20] Texas Instruments. (Jan. 18, 2019). *MMwave SDK User Guide*. Accessed: Sep. 12, 2023. [Online]. Available: https://https://e2e.ti.com/cfs-file/__key/communityserver-discussions-components-files/1023/7801.mmwave_5F00_sdk_5F00_user_5F00_guide.pdf

[21] Y. Liu, B. Tian, S. Chen, F. Zhu, and K. Wang, "A survey of vision-based vehicle detection and tracking techniques in ITS," in *Proc. IEEE Int. Conf. Veh. Electron. Saf.*, Dongguan, China, Jul. 2013, pp. 72–77, doi: 10.1109/ICVES.2013.6619606.

[22] N. Buch, S. A. Velastin, and J. Orwell, "A review of computer vision techniques for the analysis of urban traffic," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 920–939, Sep. 2011, doi: 10.1109/TITS.2011.2119372.

[23] S. R. E. Datondji, Y. Dupuis, P. Subirats, and P. Vasseur, "A survey of vision-based traffic monitoring of road intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 10, pp. 2681–2698, Oct. 2016, doi: 10.1109/TITS.2016.2530146.

[24] C. Liu, D. Q. Huynh, Y. Sun, M. Reynolds, and S. Atkinson, "A vision-based pipeline for vehicle counting, speed estimation, and classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 12, pp. 7547–7560, Dec. 2021, doi: 10.1109/TITS.2020.3004066.

[25] W. Liu, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)* (Lecture Notes in Computer Science), Amsterdam, The Netherlands, 2016, p. 2137, doi: 10.1007/978-3-319-46448-0_2.

[26] L. Chen, S. Lin, X. Lu, D. Cao, H. Wu, C. Guo, C. Liu, and F.-Y. Wang, "Deep neural network based vehicle and pedestrian detection for autonomous driving: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3234–3246, Jun. 2021, doi: 10.1109/TITS.2020.2993926.

[27] S. Raschka, "Model evaluation, model selection, and algorithm selection in machine learning," 2018, *arXiv:1811.12808*.

[28] H. Bischof, "Autonomous multi-sensor vehicle classification for traffic monitoring," in *Data and Mobility* (Advances in Intelligent and Soft Computing), vol. 81, J. Dh, H. Hufnagl, E. Juritsch, R. Pfliegl, H. K. Schimany, H. Schnegger, Eds. Berlin, Germany: Springer, 2010, doi: 10.1007/978-3-642-15503-1_2.

[29] H. Cho, Y.-W. Seo, B. V. K. V. Kumar, and R. R. Rajkumar, "A multi-sensor fusion system for moving object detection and tracking in urban driving environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Hong Kong, May 2014, pp. 1836–1843, doi: 10.1109/ICRA.2014.6907100.

[30] I.-S. Weon, S.-G. Lee, and J.-K. Ryu, "Object recognition based interpolation with 3D LiDAR and vision for autonomous driving of an intelligent vehicle," *IEEE Access*, vol. 8, pp. 65599–65608, 2020, doi: 10.1109/ACCESS.2020.2982681.

[31] D. Roy, Y. Li, T. Jian, P. Tian, K. Chowdhury, and S. Ioannidis, "Multi-modality sensing and data fusion for multi-vehicle detection," *IEEE Trans. Multimedia*, vol. 25, pp. 2280–2295, 2023, doi: 10.1109/TMM.2022.3145663.

[32] W. Zhang, K. Liu, and H. Li, "Traffic vehicle detection by fusion of millimeter wave radar and camera," in *Proc. 3rd Int. Conf. Inf. Sci., Parallel Distrib. Syst. (ISPDS)*, Guangzhou, China, Jul. 2022, pp. 105–108, doi: 10.1109/ISPDS56360.2022.9874115.

[33] F. Cui, Q. Zhang, J. Wu, Y. Song, Z. Xie, C. Song, and Z. Xu, "Online multipedestrian tracking based on fused detections of millimeter wave radar and vision," *IEEE Sensors J.*, vol. 23, no. 14, pp. 15702–15712, 2023, doi: 10.1109/JSEN.2023.3255924.

[34] Y. Liu and Y. Liu, "A data fusion model for millimeter-wave radar and vision sensor in advanced driving assistance system," *Int. J. Automot. Technol.*, vol. 22, no. 6, pp. 1695–1709, Dec. 2021, doi: 10.1007/s12239-021-0146-8.

[35] Z. Wang, X. Miao, Z. Huang, and H. Luo, "Research of target detection and classification techniques using millimeter-wave radar and vision sensors," *Remote Sens.*, vol. 13, no. 6, p. 1064, Mar. 2021, doi: 10.3390/rs13061064.

[36] H. Yu, Y. Luo, M. Shu, Y. Huo, Z. Yang, Y. Shi, Z. Guo, H. Li, X. Hu, J. Yuan, and Z. Nie, "DAIR-V2X: A large-scale dataset for vehicle-infrastructure cooperative 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 21329–21338, doi: 10.1109/CVPR52688.2022.02067.

[37] C. Creß, W. Zimmer, L. Strand, M. Fortkord, S. Dai, V. Lakshminarasimhan, and A. Knoll, "A9-dataset: multi-sensor infrastructure-based dataset for mobility research," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Aachen, Germany, Jun. 2022, pp. 965–970, doi: 10.1109/IV51971.2022.9827401.

[38] S. Busch, C. Koetsier, and J. Axmann. (2022). *Dataset: LUMPI: The Leibniz University Multi-Perspective Intersection Dataset*. [Online]. Available: https://doi.org/10.25835/z54qcu1b

[39] X. Ye, M. Shu, H. Li, Y. Shi, Y. Li, G. Wang, X. Tan, and E. Ding, "Rope3D: The roadside perception dataset for autonomous driving and monocular 3D object detection task," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 21309–21318, doi: 10.1109/CVPR52688.2022.02065.

[40] J. Sochor, J. Špaňhel, and A. Herout, "BoxCars: Improving fine-grained recognition of vehicles using 3-D bounding boxes in traffic surveillance," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 1, pp. 97–108, Jan. 2019, doi: 10.1109/TITS.2018.2799228.

[41] E. Strigel, D. Meissner, F. Seeliger, B. Wilking, and K. Dietmayer, "The Ko-PER intersection laserscanner and video dataset," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Qingdao, China, Oct. 2014, pp. 1900–1901, doi: 10.1109/ITSC.2014.6957976.

[42] H. Wang, X. Zhang, Z. Li, J. Li, K. Wang, Z. Lei, and R. Haibing, "IPS300+: A challenging multi-modal data sets for intersection perception system," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 2539–2545.

[43] C. H. Bahnsen and T. B. Moeslund, "Rain removal in traffic surveillance: Does it matter?" *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 8, pp. 2802–2819, Aug. 2019, doi: 10.1109/TITS.2018.2872502.

[44] J. Wang, T. Fu, J. Xue, C. Li, H. Song, W. Xu, and Q. Shangguan, "Realtime wide-area vehicle trajectory tracking using millimeter-wave radar sensors and the open TJRD TS dataset," *Int. J. Transp. Sci. Technol.*, vol. 12, no. 1, pp. 273–290, Mar. 2023.

[45] F. Jin, A. Sengupta, S. Cao, and Y.-J. Wu, "Mmwave radar point cloud segmentation using GMM in multimodal traffic monitoring," in *Proc. IEEE Int. Radar Conf. (RADAR)*, Washington, DC, USA, Apr. 2020, pp. 732–737, doi: 10.1109/RADAR42522.2020.9114662.

[46] B. Yang, H. Zhang, Y. Chen, Y. Zhou, and Y. Peng, "Urban traffic imaging using millimeter-wave radar," *Remote Sens.*, vol. 14, no. 21, p. 5416, Oct. 2022, doi: 10.3390/rs14215416.

[47] L. Wen, D. Du, Z. Cai, Z. Lei, M.-C. Chang, H. Qi, J. Lim, M.-H. Yang, and S. Lyu, "UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking," *Comput. Vis. Image Understand.*, vol. 193, Apr. 2020, Art. no. 102907.

[48] Commons Creative. (2021). *Attribution-Noncommercial-Noderivatives 4.0 International (Cc by-Nc-Nd 4.0)*. Accessed: Jun. 26, 2023. [Online]. Available: https://creativecommons.org/licenses/by-nc-nd/4.0/

[49] (2021). *National Oceanic and Atmospheric Administration*. Accessed: Jun. 26, 2023. [Online]. Available: https://forecast.weather.gov/glossary.php?word=Sky%20Condition

[50] Nvidia. (Aug. 22, 2022). *Jetson Nano Developer Kit*. Accessed: May 2, 2023. [Online]. Available: https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/

[51] Nvidia. *Jetson Nano Developer Kit*. Accessed: Aug. 22, 2022. [Online]. Available: https://developer.nvidia.com/embedded/jetson-nano-developer-kit

[52] Sony. *IMX219*. Accessed: Aug. 22, 2022. [Online]. Available: https://www.gophotonics.com/products/CMOS-image-sensors/sony-corporation/21-209-imx219

[53] Texas Instruments. *AWR1843BOOST*. Accessed: Aug. 22, 2022. [Online]. Available: https://www.ti.com/tool/AWR1843BOOST

[54] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011, doi: 10.1109/TPAMI.2010.161.

[55] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022, *arXiv:2207.02696*.

[56] S. Raj and D. Ghosh, "Improved and optimal DBSCAN for embedded applications using high-resolution automotive radar," in *Proc. 21st Int. Radar Symp. (IRS)*, Warsaw, Poland, Oct. 2020, pp. 343–346, doi: 10.23919/IRS48640.2020.9253774.

[57] A. Manjunath, Y. Liu, B. Henriques, and A. Engstle, "Radar based object detection and tracking for autonomous driving," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Munich, Germany, Apr. 2018, pp. 1–4, doi: 10.1109/ICMIM.2018.8443497.

[58] T. D. Bufler and R. M. Narayanan, "Radar classification of indoor targets using support vector machines," *IET Radar, Sonar Navigat.*, vol. 10, no. 8, pp. 1468–1476, Oct. 2016, doi: 10.1049/iet-rsn.2015.0580.

[59] A. Diab, R. Kashef, and A. Shaker, "Deep learning for LiDAR point cloud classification in remote sensing," *Sensors*, vol. 22, no. 20, p. 7868, 2022, doi: 10.3390/s22207868.

[60] Y. Sun, H. Zhang, Z. Huang, and B. Liu, "R2P: A deep learning model from Mmwave radar to point cloud," 2022, *arXiv:2207.10690*.

[61] Scikit-Learn Developers. (Aug. 22, 2022). *Support Vector Machines*. Accessed: Apr. 2, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/SVM

[62] R. Shi, K. N. Ngan, and S. Li, "Jaccard index compensation for object segmentation evaluation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Paris, France, Oct. 2014, pp. 4457–4461, doi: 10.1109/ICIP.2014.7025904.

[63] R. Padilla, S. L. Netto, and E. A. B. da Silva, "A survey on performance metrics for object-detection algorithms," in *Proc. Int. Conf. Syst., Signals Image Process. (IWSSIP)*, Niteroi, Brazil, Jul. 2020, pp. 237–242, doi: 10.1109/IWSSIP48289.2020.9145130.

JÜRGEN SOOM received the B.Sc. degree in computer systems from Tallinn University of Technology, in 2017, and the M.Sc. degree, in 2020. He is currently an Early-Stage Researcher with the Embedded AI Research Laboratory. His current Ph.D. thesis is related to computer vision, machine learning, and embedded hardware.

MAIRO LEIER received the Ph.D. degree in computer systems from Tallinn University of Technology, in 2016. He was a Research Scientist with the Department of Computer Systems, Tallinn University of Technology, where he currently leads the Embedded AI Research Laboratory. His current research interests include machine learning on embedded systems, optimization techniques, and edge computing.

KARL JANSON received the Ph.D. degree in computer systems from Tallinn University of Technology, in 2021. He is currently with the Embedded AI Research Laboratory. His current research interests include machine learning, radars, embedded hardware, edge computing, and signal processing.

JEFFREY A. TUHTAN (Member, IEEE) received the B.Sc. degree in civil engineering from California Polytechnic University, San Luis Obispo, CA, USA, in 2004, and the M.Sc. and Dr.-Eng. degrees in water resources engineering and management from the Institute for Modelling Hydraulic and Environmental Systems, University of Stuttgart, Germany, in 2007 and 2011, respectively. Since 2016, he has been leading the Centre for Environmental Sensing and Intelligence, Department of Computer Systems, Tallinn University of Technology. His research interests include data-driven modeling and bio-inspired underwater sensing in extreme environments.

• • •

Appendix 3

III

J. Soom, I. Boavida, R. Leite, M. J. Costa, G. Toming, M. Leier, and J. A. Tuhtan. Open real-time, non-invasive fish detection and size estimation utilizing binocular camera system in a portuguese river affected by hydropeaking. *Ecological Informatics*, 90, 2025

Ecological Informatics 90 (2025) 103196

Contents lists available at ScienceDirect

Ecological Informatics

journal homepage: www.elsevier.com/locate/ecolinf

Open real-time, non-invasive fish detection and size estimation utilizing binocular camera system in a Portuguese river affected by hydropeaking

Jürgen Soom [a,b,*], Isabel Boavida [c], Renan Leite [c], Maria João Costa [d], Gert Toming [b,1], Mairo Leier [a], Jeffrey A. Tuhtan [b,1]

[a] Embedded AI Research Lab, Tallinn University of Technology, Akadeemia tee 15A, Tallinn, 12618, Harjumaa, Estonia
[b] Department of Computer Systems, Akadeemia tee 15A, Tallinn, 12618, Harjumaa, Estonia
[c] Civil Engineering Research and Innovation for Sustainability, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal
[d] Forest Research Centre, TERRA Associate Laboratory, School of Agriculture, University of Lisbon, Lisboa, Portugal

ARTICLE INFO

Dataset link: 10.5281/zenodo.14519903

Keywords:
Fish migration
Size estimation
Binocular vision
Hydropeaking
Edge computing

ABSTRACT

The need for efficient approaches to track and assess fish behavior in rivers impacted by hydropeaking is increasing. Nonetheless, employing an automated camera system for underwater monitoring requires that the algorithms function under highly variable environmental conditions, which affect the ability to detect and assess fish size. Additionally, there is a lack of openly accessible freshwater fish classification and size estimation datasets. To address these limitations, we propose a binocular underwater fish monitoring system capable of real-time fish detection and size estimation. The system was deployed and tested over one week in two Portuguese rivers affected by hydropeaking. The week-long analysis also provided new insights regarding wild fish behavior in rivers affected by hydropeaking. Results indicate that hydropeaking strongly influences how fish may use instream flow refuges during hydropeaking. Fish were less frequently detected in the flow refuge during peak flow events, suggesting that the flow conditions created habitat instability and difficulty accessing the flow refuge. In contrast, fish in the non-hydropeaking river consistently used refuge areas, reinforcing their importance as shelter during natural flow variations. This study demonstrates the potential of a computer vision-based pipeline for real-time, fully automated fish monitoring of hydropeaking's impacts on riverine fish. Additionally, we provide PTFish, an open dataset with 18,523 manually annotated frames featuring infrared and color video frames. These findings emphasize that automated, camera-based solutions for hydropeaking monitoring can be used to develop evidence-based mitigation strategies to sustain fish populations in rivers impacted by hydropeaking.

1. Introduction

The majority of global energy production is sourced from fossil fuels. However, renewable energy sources such as hydropower, solar, and wind provide an increasing share of the global energy mix, contributing to a gradual reduction in fossil fuel dependence. According to the International Energy Agency (IEA) report, by 2025, renewable energy is projected to surpass coal, becoming the largest source of electricity generation (IEA, 2024b). Despite the rapid expansion of solar and wind power, hydropower remains one of the largest renewable energy sources (IEA, 2024a; IHA, 2024). Hydroelectric power plants with reservoirs provide exceptional operational flexibility, enabling immediate responses to fluctuations in the electricity demand. Their inherent flexibility and storage capacity enhance efficiency and cost-effectiveness in supporting intermittent energy sources. Notably, many hydropower facilities can rapidly transition from zero output to maximum generation within minutes, offering vital backup power during significant electricity outages or disruptions (U.S. Geological Survey, 2025; U.S. Department of Energy, 2025).

Despite its numerous advantages, it is also essential to acknowledge the inherent limitations of hydropower. Hydropeaking results from rapid and frequent flow fluctuations caused by intermittent water releases through turbines to meet peak energy demand. These fluctuations alter flow patterns, affect the water temperature, impact sediment transport, and change the dissolved gas levels within ecosystems downstream of hydropower operations. These alterations affect various aspects of aquatic ecosystems, including fish growth, behavior, reproductive success, habitat, and migration patterns, among others (Bipa et al., 2024; He et al., 2024). Considering fish communities, these

---

[*] Corresponding author at: Embedded AI Research Lab, Tallinn University of Technology, Akadeemia tee 15A, Tallinn, 12618, Harjumaa, Estonia.
E-mail address: jurgen.soom@taltech.ee (J. Soom).

[1] Jeffrey Tuhtan and Gert Toming's contributions were funded by the Estonian Research Council Grant PRG 2198.

https://doi.org/10.1016/j.ecoinf.2025.103196
Received 26 December 2024; Received in revised form 8 May 2025; Accepted 8 May 2025
Available online 26 May 2025
1574-9541/© 2025 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC license (http://creativecommons.org/licenses/by-nc/4.0/).

fluctuations have been reported to cause both lateral and longitudinal displacements, leading to habitat shifts; reducing the survival rates due to stranding; and are known to disrupt key life-cycle events such as growth, reproductive migration and spawning. Furthermore, hydropeaking can also lead to habitat fragmentation, erosion, and loss of riparian vegetation, impacting terrestrial ecosystems that depend on the aquatic environment (Bejarano et al., 2018). However, there is limited understanding of the long-term ecological consequences of hydropeaking and its cumulative effects on aquatic ecosystems (Bipa et al., 2024; Schmutz and Sendzimir, 2018).

Fish size is an important parameter used to assess and analyze the impacts of hydropeaking. Size estimation can give an in-depth overview of population structure, including growth rates, age distribution, the juvenile-to-adult ratio, and overall weight (Pope et al., 2010; Froese et al., 2014). These metrics can provide significant insights into the ecosystem, offering a comprehensive understanding of its dynamics and overall well-being.

1.1. *Previous work*

Electrofishing requires direct fish handling and provides biometric data needed to estimate community composition and distribution. Moreover, the handling is known to cause stress and potentially injury to fish, which can have long-term effects on their health and well-being (Snyder, 2003). Additionally, fish size estimation using manual handling methods can be prone to inconsistencies and bias due to human error and expertise dependency (Bravata et al., 2020). To address these issues, fish size estimation using images and video can provide more consistent and reliable data (Bravata et al., 2020). Another key consideration is that smaller fish, particularly juveniles, are more difficult to catch and sample in the wild, and are more susceptible to the adverse impacts of hydropeaking. The rapid changes in water depth and velocity subject juvenile fish to an elevated risk of downstream displacement and stranding (Naudascher et al., 2024; Boavida et al., 2023), since they are less capable to cope with rapid changes in hydrodynamic conditions (Enders et al., 2017).

Performing automated size estimation *in situ*, and in real-time remains highly challenging for several reasons. A common cause of poor size estimation accuracy is the motion and orientation of a swimming fish's body. The presence of foreign objects or other fish can cause partial occlusion, making it difficult to obtain accurate size estimates. Environmental factors, such as low luminosity with low-contrast regions, air bubbles, turbidity, periphytic biofilm, and light overexposure, further exacerbate the issue (Soom et al., 2022). Consequently, the size estimation process becomes more challenging in the presence of ever-changing environmental conditions, leading to larger uncertainty.

Unfortunately, due to the difficulty of conducting research on fish in the wild, size estimation techniques are most commonly carried out in highly controlled environments, allowing for more precise measurements and systematic observations. However, the results of laboratory-based methods are unlikely to be matched in more complex natural habitats, where factors such as water quality, behavior, and species interactions can also significantly influence size assessments (Yu et al., 2024; Shi et al., 2022; Wang et al., 2024b; Gao et al., 2024; Muñoz-Benavent et al., 2018). Joint research conducted by the Institut Mediterrani d'Estudis Avançats (IMEDEA) and the Universitat de les Illes Balears in Spain focused on estimating body size based on head dimensions, addressing some of the challenges previously highlighted. Their proposed methodology yielded promising results, with a maximum deviation of only 4.0 cm between the estimated and measured mean body lengths. Notably, these experiments were performed on ex vivo specimens under controlled conditions (Álvarez-Ellacuría et al., 2019). A similar outcome was reported by research teams from the University of Girona in Spain and the Institute of Marine Research in Norway, who estimated specimen lengths using stereo cues. While their observations demonstrated a high degree of accuracy, the experiments were conducted in highly controlled environments, eliminating external factors and resulting in static conditions (Garcia et al., 2019). Although statistical methods can approximate and compensate results in controlled environments, their predictive accuracy diminishes significantly in uncontrolled settings, where external factors exert greater influence. This limitation reduces their reliability and applicability under adverse conditions (Álvarez-Ellacuría et al., 2019; Garcia et al., 2019; Tseng et al., 2020; Monkman et al., 2019).

The findings of these previous studies highlight the need for a non-invasive, in-situ camera-based monitoring system capable of estimating fish sizes. Such a method would enable a more comprehensive analysis of size-class specific responses to hydropeaking, and provide new and additional insights into habitat preferences and potential vulnerabilities which are lacking (Karlsson, 2024; Kevin M. Boswell and Jr., 2008).

1.1.1. *Overview of fish datasets for computer vision applications*

An examination of the available fish datasets for computer vision applications, as summarized in Table 1, highlights a critical research gap; namely, the limited availability of datasets which can be used to develop, test and validate fish size estimation methods. This limitation needs to be addressed by introducing open datasets. To this end, we propose **PTFish**, a dataset comprising multi-modal frames tailored for size estimation. The specifics of the dataset are further discussed in Section 2.2.

DeepFish — Collection of in situ samples from 20 different tropical marine habitats (Australia). Featuring approximately 40k high-resolution (1920 × 1080) images of tropical marine fish, out of which 3.2k frames have been annotated for object detection purposes (Saleh et al., 2020).

AFFiNe — Featuring 7k samples, covering 30 common freshwater fish species, collected ex vivo (Netherlands). The data contains photos taken by anglers of fish out of water and are organized by fish species, bounding box information and fish size measurements (Jorrit Venema, 2021).

Fish4Knowledge — The dataset was collected in situ (Taiwan). It has the most extensive collection of reef species detected and identified automatically, with some detections manually reviewed. Fisher et al. (2016)

QUT — Dataset containing 3960 images collected in Australia, containing 468 marine species, taken in different conditions (in situ, ex vivo, in vitro) (Anantharajah et al., 2014)

Brakish — collected in situ in brackish water (Denmark). The images were collected using artificial illumination, and include some non-fish objects. It is the largest dataset for brackish European fish species, containing around 14.5k annotated images for object detection and classification. The annotated images cover six classes fish species, as well as marine fauna including starfish or crabs. Pedersen et al. (2019).

RockFish — Collection of thirteen species of rockfish (Sebastes spp.). The dataset was collected in southern California, USA, using a remotely operated vehicle. In total, the dataset contains 4307 samples (Cutter et al., 2015).

1.2. *Objectives*

The primary objective of this work was to design and develop a computer vision processing pipeline for riverine fish monitoring. The pipeline integrates fish detection with depth and size estimation to facilitate the monitoring and analysis of hydropeaking impacts in freshwater rivers. A significant contribution of this work was also a novel open-access dataset aimed at addressing the lack of suitable datasets for fish size estimation. Additionally, the work focuses on deploying and evaluating the complete processing pipeline on cost-effective and low-power embedded hardware. The main contributions of this work are three-fold:

Table 1
Comparison of open access fish datasets for computer vision tasks. In situ: on site, in vitro: in lab, ex vivo: on dead specimens. ObD: object detection, FiC: classification (fish/ no fish), SpC: species classification, SzE: size estimation, Seg: segmentation. The number of frames corresponds to the number of available images before augmentation.

Dataset	Environment	Task	Number of frames	Species	Resolution	Mono/Stereo
DeepFish	in situ/marine	ObD Seg	39 766	20	1920 × 1080	Mono
Rockfish	in situ/marine	ObD	4307	13	1280 × 720	Mono
Fish4Knowledge	in situ/marine	ObD	27 370	23	352 × 240	Mono
QUT	in vitro ex vivo/marine	SpC	3960	468	480 × 360	Mono
Brakish	in situ/marine	ObD SzE	14 518	6	1920 × 1080	Mono
AFFiNe	ex vivo/freshwater	ObD SpC SzE	7000	30	710 × 852	Mono
PTF (Our work)	in situ/freshwater	ObD SzE	18 523	3	2560 × 960	**Stereo**

Table 2
Summary of the Jetson Orin Nano hardware specifications.

Component/Feature	Nvidia Jetson Orin Nano (Nvidia, 2024)
CPU	Hexa-core ARM A78AE @ 1.5 GHz
GPU	1024-core Ampere with 32 dedicated Tensor Cores
Memory	8 GB LPDDR5
Storage	MicroSD/NVMe PCIe 3.0 x4
Camera	2x MIPI CSI-2 connectors
Network	Wi-Fi (802.11ac), GbE

- Design a computationally lightweight size estimation pipeline, running on low-power, commercially available embedded hardware.
- Test the pipeline's ability to detect fish and classify them as either adult or juvenile, and in doing so, provide a preliminary analysis of hydropeaking at a high temporal resolution in two Portuguese rivers.
- Provide an open dataset dedicated to and supporting the development of fish detection and size estimation applications.

The rest of the article is structured as follows: Section 2.1 provides a comprehensive overview of the used hardware. Next, the open dataset and methods for object detection, depth, size estimation, and the evaluation methods are provided in Section 2. The findings are subsequently detailed in Section 3. Section 4 provides the primary outcomes and limitations of this work. Finally, Section 5 explores potential directions for future research.

2. Materials and methods

2.1. Hardware

2.1.1. Embedded hardware

The chosen embedded hardware was the Nvidia Jetson Orin Nano. The board features a Hexa-core on Cortex-A78AE architecture running at 1.5 GHz with 8 GB of dedicated system memory. Additionally, the selected hardware includes a 1024-core GPU based on NVIDIA Ampere architecture with 32 dedicated Tensor Cores. Table 2 summarizes the hardware specifications.

2.1.2. Camera sensor

The binocular camera system incorporates two identical camera sensors: infrared (IR) and RGB. Both sensors share common characteristics, featuring a resolution of up to 3072 × 2048 pixels. The lens has an aperture of f/1.8 and a focal length of 4.1 mm. In addition, both sensors have 90° angle of view horizontally and 67° vertically. Table 3 summarizes the camera sensor specifications.

Table 3
Overview of the camera sensor specifications used in this work.

Feature	Mobotix Mx-O-SMA-S-6N016 (Mobotix, 2023)
Resolution	1280 × 960
Aperture (F)	f/1.8
Focal length (mm)	4.1
Angle of view (Horizontal)	92°
Angle of view (Vertical)	67°

Table 4
Technical specifications for different system configurations, including power consumption, recommended installed solar panel power, and necessary battery capacity.

Feature	Standalone	Equipped with NAS	Equipped with Jetson Orin Nano
Power consumption (W)	14	25	30
Installed power (W)	400–500	>800	>800
Necessary battery capacity (Ah)	240	480	600

2.1.3. Power

The system implemented in the Portugal installation is designed to operate with a consistent power source; however, it is also tailored for deployment in regions where infrastructure may be limited or insufficient. This system is capable of functioning independently using either battery power or solar panel energy. An overview of the operational requirements for the device when powered by battery or solar energy is presented in Table 4.

The standalone camera system has a power consumption of 14 W, which necessitates the use of solar panels rated at 400–500 W and a 240 Ah battery. When integrated with a Network Attached Storage (NAS), the power consumption increases to 25 W. This configuration requires the installation of more than 800 W of solar panels as well as a 480 Ah battery. When equipped with the Jetson module, the power consumption increases further to 30 W. This configuration can use the same 800 W solar panels, however the battery capacity must be increased to 600 Ah to maintain continuous operation.

2.2. Dataset

The data used in the PTFish dataset was obtained as part of the research project EcoPeak4Fish (Boavida et al., 2022). The main goal was of the project was to contribute to sustainable fish populations downstream of hydropower plants by using an integrated approach. Accordingly, this approach included studying fish microhabitat use at hydropeaking-affected and undisturbed sites, and the selection, and implementation of the most effective flow-refuge designs downstream

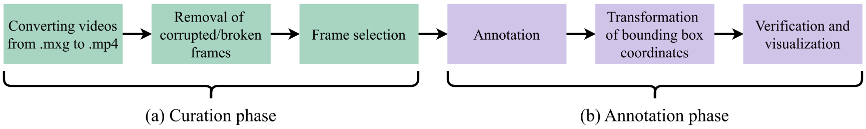

Converting videos from .mxg to .mp4
Removal of corrupted/broken frames
Frame selection
Annotation
Transformation of bounding box coordinates
Verification and visualization
(a) Curation phase
(b) Annotation phase

Fig. 1. The dataset creation workflow is comprised of two separate phases. (a) The videos are converted from propriety .mxg format to .mp4 during the initial curation phase. Any corrupted or broken frames are then eliminated. Subsequently, the frames are selected at specific intervals to avoid having similar samples. (b) In the annotation phase, the selected frames undergo the annotation process. The annotations are then transformed and exported into suitable formats, such as YOLO. Finally, verification and visualization are carried out to validate the annotations.

Table 5
Summary of annotation statistics for the PTFish dataset, detailing key annotation metrics for the Bragado and Covas Do Barroso sites, including total annotated frames, frame counts segmented by time of day (morning, afternoon, evening, and night), total number of fish bounding boxes, and average fish instances per frame. It also includes statistics on bounding box sizes, reporting the minimum, maximum, mean, and standard deviation in pixel dimensions. The Bragado site contains a significantly larger volume of annotated data, both in terms of frames and bounding boxes, and exhibits greater variability in object size. In contrast, the Covas Do Barroso site has a more limited data volume with larger average bounding boxes and fewer fish per frame.

Metric	Bragado	Covas Do Barroso
Annotated frames	18,292	231
Frame count (Morning, 06:00–12:00)	3076	24
Frame count (Afternoon, 12:00–18:00)	11 594	78
Frame count (Evening, 18:00–21:00)	2878	38
Frame count (Night, 21:00–06:00)	744	91
Total bounding boxes (fish samples)	126,299	580
Average fish per frame	6.90	2.51
Min bbox size (pixels)	16 × 16	5 × 1
Max bbox size (pixels)	679 × 659	734 × 599
Mean bbox size (pixels)	127 × 81	176 × 129
Std. dev. bbox size	81 × 49	110 × 74

of hydropower plants (Leite et al., 2024). The fish refuge usage was monitored using the binocular camera system. The recordings were collected during spring and late summer periods from two separate sites: Bragado (Fig. 2(a)), located in Avelames River, and Covas Do Barroso in Couto River (Fig. 2(b)), both tributaries of the Tamega River (Douro River basin), Portugal. The recordings feature three species: the Northern Iberian chub (*Squalius carolitertii*), the Northern straight-mouth nase (*Pseudochondrostoma duriense*), and the brown trout (*Salmo trutta*). An overview of the curation and annotation workflow is provided in Fig. 1.

The curated dataset contains 18,523 manually annotated frames from infrared and color cameras. From 18,523 frames, 18,292 originate from Bragado and only 231 were taken from Covas Do Barroso. The annotations were conducted by a team of two experienced annotators, both possessing prior expertise in the field. Notably, one of the annotators has substantial expertise in projects pertaining to freshwater ecosystems, fish migration, and the ecological flow regimes related to hydroelectric plants. The annotations were primarily performed using the MATLAB VideoLabeler, a specialized tool designed for both manual and semi-automated annotation processes. Additionally, custom Python scripts were utilized during the initial curation and review phases to facilitate data filtering, ensure annotation validation, and maintain consistency throughout the dataset. An overview of the conditions and a link to the dataset is provided in Section 8. Table 5 summarizes the annotation statistics for the PTFish dataset, including frame counts, fish detection metrics, bounding box size distributions, and temporal sampling across both recording sites.

2.3. *Fish detection model architecture*

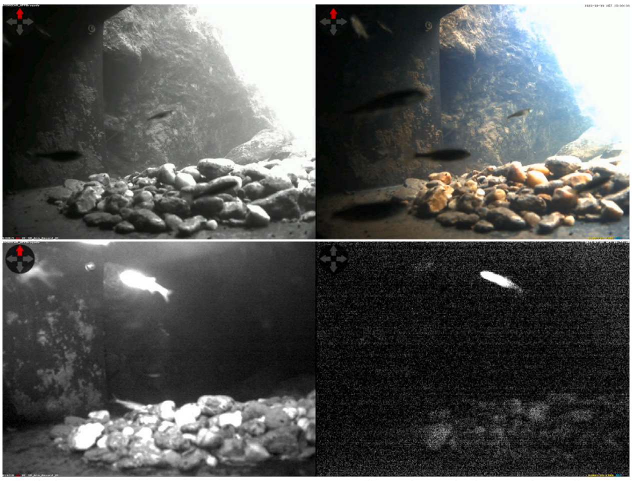
The object detection model architecture choice depends on three critical criteria: performance, computational efficiency, and the target hardware (Lazarevich et al., 2023). As described in Section 2.1.1, the model will be deployed and evaluated on the Nvidia Orin Nano, which must be capable of running the proposed approach, which includes object detection, depth, and size estimation at least 5 frames per second (fps) minimum, at 2560 × 960 resolution. The object detection model must achieve an F1 score of at least 0.80 with the hold-out validation dataset. The specified requirements considerably narrowed the pool of potential candidates. Based on the outcomes of the preliminary analysis, two potential candidates were selected for further evaluation:

YOLOv8 — The YOLO (You Only Look Once) is the state-of-the-art (SOTA) model designed to be computationally lightweight, making it suitable for deploying on low-power hardware. YOLOv8 is a direct successor to YOLOv5 (Jocher et al., 2022), and includes significant changes to the core architecture. A new anchor-free detection system directly predicts the center of an object instead of the offset from a known anchor box. It reduces the number of box predictions and improves Non-Maximum Suppression (NMS), a complex post-processing step that sifts through candidate detections after inference. Another significant change has been implemented to the training routine. Mosaic augmentation involves stitching together four images, forcing the model to learn objects in new locations, in partial occlusion, and against different surrounding pixels. However, this augmentation is empirically shown to degrade performance if performed through the whole training routine, and mosaic augmentation was disabled during the last ten epochs. For the preliminary analysis, YOLOv8n and YOLOv8s models were selected for further evaluation (Jocher et al., 2023).

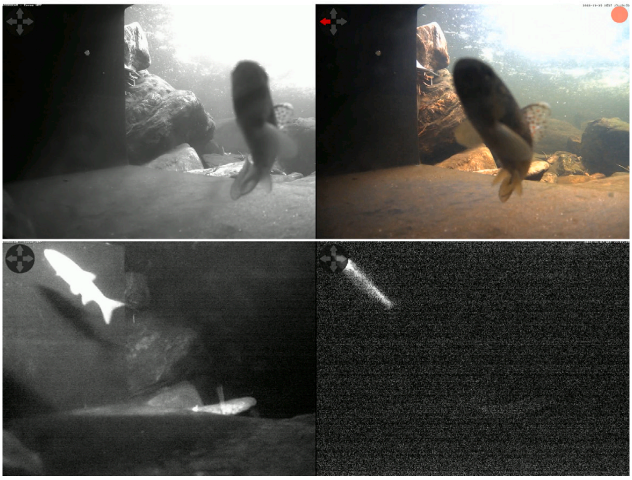
YOLOv10 — Compared to YOLOv8, the most significant change to the architecture is the elimination of NMS. Instead, the architecture utilizes consistent dual assignments, reducing the inference latency. Additional improvements are achieved in both the model's accuracy and efficiency by using a holistic model design, featuring lightweight classification heads, spatial-channel decoupled downsampling and rank-guided block design. Lastly, the architecture incorporates large-kernel convolutions and partial self-attention modules, further improving the performance at minimal or no computational cost (Wang et al., 2024a). Two models were selected to evaluate the performance and efficiency: YOLOv10-N, the smallest of the models designed for resource-constrained hardware, and YOLOv10-S, which balances speed and accuracy.

RT-DETR — This YOLO series is widely recognized as the preferred option for applications requiring real-time object detection. The primary limitation of YOLO lies in its use of Non-Maximum Suppression (NMS), which can decrease the speed and the accuracy. In contrast, end-to-end transformer-based detectors, such as DETR, present a viable alternative by eliminating the NMS. Unfortunately, their high computational demands restrict their applicability across multiple domains. The RT-DETR model seeks to address this challenge by utilizing an efficient hybrid encoder that processes multi-scale features with greater speed through the decoupling of intra-scale interactions and cross-scale fusion. It employs uncertainty-minimal query selection to generate high-quality initial queries for the decoder, thereby enhancing accuracy. For preliminary testing, we selected the RT-DETR-l model (Zhao et al., 2024).

DEYO — DEYO (DETR with YOLO) represents advancements in the realm of object detection by integrating the strengths of DEtection TRansformers (DETR) and You Only Look Once (YOLO) architectures.



(a) Bragado during the day and at night. Left images are taken from the infrared camera, right images are from the color camera.



(b) Covas Do Barroso during the day and at night.

Fig. 2. Examples of recording locations situated in tributaries of the Tamega River (Douro River basin), Portugal: (a) Bragado, located in Avelames River. (b) Covas Do Barroso located in Couto River. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The hybrid model is designed to improve both the speed and the accuracy by employing a systematic two-stage training methodology. In the first stage, the model utilizes a classic detector with a one-to-many matching strategy to pre-train the backbone and neck of the model, subsequently freezing these pre-trained components in the second stage to train the decoder from scratch. By synthesizing YOLO's efficient feature extraction techniques with DETR's comprehensive end-to-end detection capabilities, DEYO achieves remarkable improvements in both speed and accuracy when compared to existing real-time object detectors, notably excelling without the need for additional training data. To evaluate the DEYO capabilities in this work, we chose the DEYO-N model (Ouyang, 2024).

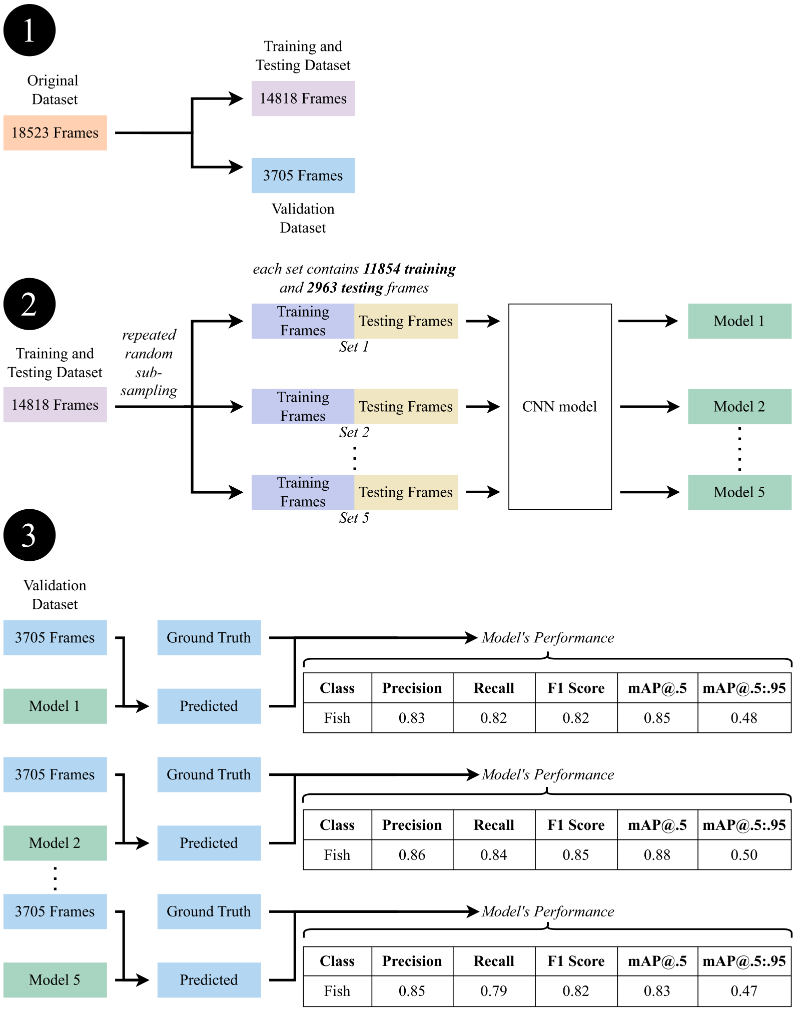
A small, manually annotated dataset consisting of 800 images was initially used to train the candidate models. The results were then evaluated using a hold-out dataset of 200 images while deployed on the target hardware to find the best-performing candidate model. Table 6 describes the selected architecture parameters and characteristics as well as the initial evaluation outcomes.

2.3.1. *Training and validation*

The object detection model used in the approach was trained using a three-step process. As illustrated in Fig. 3, five models using k-fold cross-validation (scikit-learn developers, 2024) were trained, tested, and validated. The dataset was split into separate datasets for training and validation using a ratio of 80:20. As a result, the training dataset used for training and testing comprises 14,818 frames, leaving 3705 images for validation. Using random sub-sampling, the training dataset was further split into five datasets (see Fig. 3 (2)). All five models were then trained with 20 epochs. In the final stage (see Fig. 3 (3)), the best-performing model was found by evaluating the trained models using the hold-out dataset. The results of the validation phase are further discussed in Section 3.1.

Table 6
Comparison and summary of the six selected object detection models. All models were trained using 20 epochs and later evaluated on the Nvidia Jetson Orin Developer Kit. Based on the initial outcomes, YOLOv8s was the best-performing option, with YOLOv10s close behind. Both models showed nearly identical performance; however, YOLOv8s was chosen due its 0.9ms quicker inference time. RT-DETR showed comparable performance against other YOLO models, but had the highest inference time of 40.3 ms. An outlier was DEYO-N, which performed significantly worse compared all other models.

Metric	YOLOv8n	YOLOv8s	YOLOv10n	YOLOv10s	RT-DETR	DEYO-N
Parameter count (M)	3.2	11.2	2.3	7.2	32	6.0
FLOPs (G)	8.7	28.6	6.7	21.6	110	8.9
Size (pixels)	640	640	640	640	640	640
Inference time (ms)	0.9	2.0	1.3	2.9	40.3	4.5
Precision	0.85	0.82	0.82	0.82	0.79	0.67
Recall	0.77	0.83	0.77	0.80	0.80	0.59
F1 Score	0.81	0.82	0.79	0.81	0.79	0.63
mAP@.5	0.84	0.85	0.82	0.84	0.84	0.61
mAP@.5...0.95	0.47	0.48	0.45	0.48	0.46	0.28



Class	Precision	Recall	F1 Score	mAP@.5	mAP@.5:.95
Fish	0.83	0.82	0.82	0.85	0.48

Class	Precision	Recall	F1 Score	mAP@.5	mAP@.5:.95
Fish	0.86	0.84	0.85	0.88	0.50

Class	Precision	Recall	F1 Score	mAP@.5	mAP@.5:.95
Fish	0.85	0.79	0.82	0.83	0.47

Fig. 3. (1) Illustration of the hold-out procedure for training, testing, and validating the fish detection model. (2) Usage of repeated sub-sampling was applied for testing and training and was used to train five models with best performing architecture, found in Section 2.3. (3) The best-performing model is found using the hold-out validation dataset of 3705 samples.

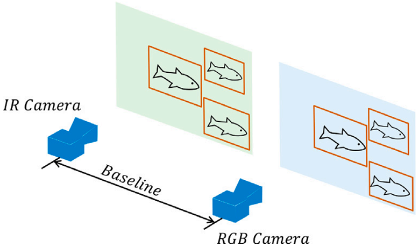


Fig. 4. Binocular vision system setup consisting of IR and RGB camera. First, the proposed solution detects and locates fish from a single frame, using both the IR and RGB cameras.

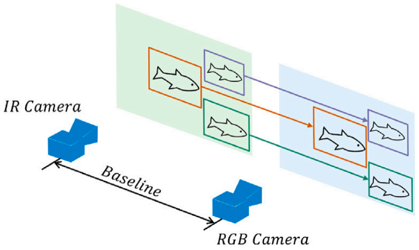


Fig. 5. After detecting fish on both images, the algorithm determines the quality of the bounding box pairs by analyzing their geometric and spatial properties.

2.4. Size estimation

To perform non-invasive fish size estimation in situ, we propose a solution combining both machine learning and classical stereo vision methods in order to remain computationally lightweight, enabling it to be deployed on low-power embedded hardware (see Section 2.1.1). As described in Fig. 4, at first detect fish from the multi-modal frame using an object detection model, described in Section 2.3.1.

Next, as illustrated in Fig. 5, we try to match pairs of detected objects between the IR and RGB frames. The process evaluates the bounding boxes properties: dimensions, width-to-height ratio, and the overall spatial positioning via the epipolar line. Before the pipeline can estimate the object size, it must determine the depth between the object and the camera. Disparity, which refers to the horizontal shift of pixels between images, arises from the parallax effect. The differences between corresponding pixels in two or more images capturing the same scene from distinct viewpoints. As depth is inversely proportional to disparity, and by knowing the baseline and the focal length, it is possible to calculate the depth using the following equation (Hamzah and Ibrahim, 2016):

$$\text{Distance (z)} = \frac{\text{Focal length} \times \text{Baseline}}{\text{Disparity}} \tag{1}$$

This method is commonly referred to as triangulation. As shown in Fig. 6, the process looks at the center pixel and the surrounding ones to calculate the depth. To avoid inaccuracies and extreme values, the median value is then calculated. Once the depth is determined, the process estimates the three-dimensional coordinates by transforming the depth data into 3D space, as illustrated in Fig. 7. The methodology then utilizes the coordinates of the two bounding box corners to compute the Euclidean distances corresponding to each bounding box. These distances are subsequently averaged to estimate the size of the fish:

$$\text{Euclidean distance (d)} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{2}$$

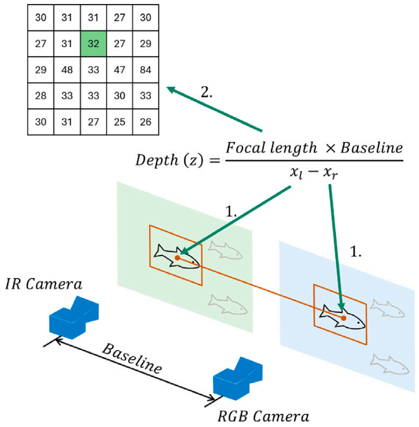


Fig. 6. The disparity is first computed and subsequently utilized to estimate depth, leveraging the focal length and baseline as scene parameters. Following the depth estimation for each point, the median value (e.g. 32 in this example) is selected from a neighborhood of depth estimates at each pixel is extracted to mitigate the influence of outliers and ensure robust depth measurement.

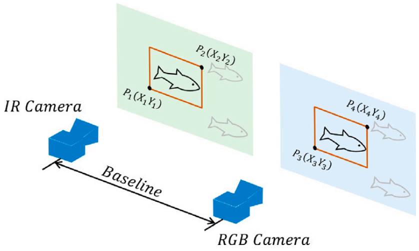


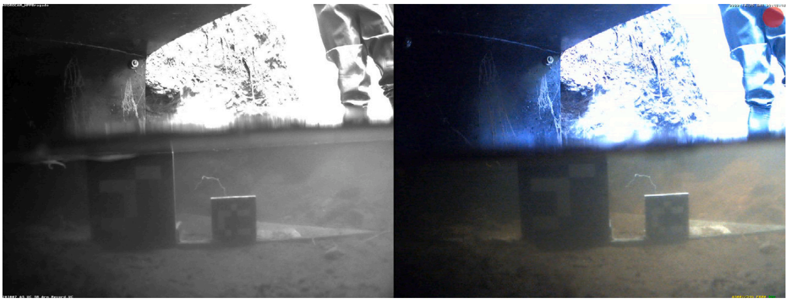
Fig. 7. The fish size is determined by calculating the Euclidean distance utilizing the coordinates of the bottom left and top right corners (x and y) for both bounding boxes, which are then averaged to estimate the fish's size (total body length).

and the mean value of the IR and RGB distance is calculated. Finally, a size threshold is applied to classify if the fish is juvenile or an adult. All fish under 10 cm are considered juvenile.

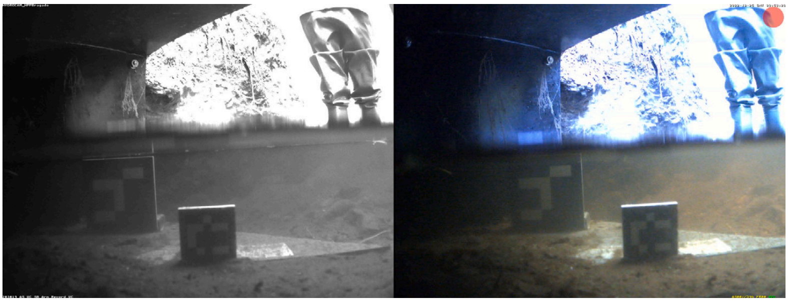
To validate the proposed depth and size estimation approach in the field, experiments were conducted using three distinct scenes, each featuring two ArUco markers of different sizes: a 10 × 10 cm marker (large) and a 5 × 5 cm marker (small). The arrangement and distances of the markers relative to the camera were varied across the scenes, as detailed in Table 7. In *Scene 1*, both ArUco markers were positioned parallel to each other at a uniform distance of 32.1 cm from the camera. In *Scene 2*, the smaller marker was placed closer to the camera at 25.2 cm, while the larger marker was positioned further away at 40.3 cm. In *Scene 3*, the larger marker was placed closer to the camera at 30.0 cm, and the smaller marker was situated farther away at 46.2 cm. All three scenes are also depicted in Fig. 8. The outcome of depth and size evaluation is discussed further in Section 3.2.

2.5. Hydropeaking

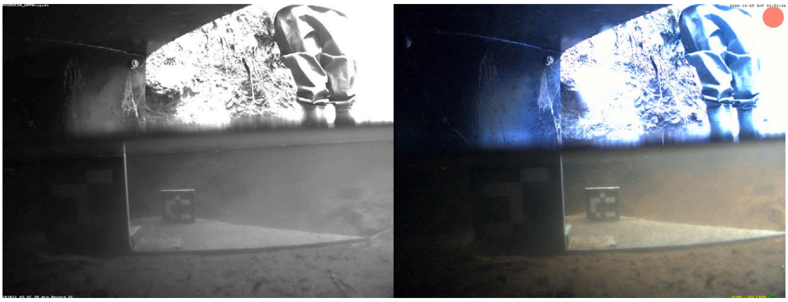
The analysis of hydropeaking effects was conducted in Covas do Barroso and Bragado to assess and validate the effectiveness of the designed pipeline. The footage used in the analysis was recorded over the period from April 26 to May 3 in 2023. Three one-hour periods were selected from each day, at specific times of the day: morning (08:00 to 09:00), afternoon (13:00 to 14:00), and evening (18:00 to



(a) Depth and size estimation evaluation Scene 1. Both markers are place parallel to the camera at distance of 32.1 cm.



(b) Depth and size estimation evaluation Scene 2. The smaller (5×5 cm) ArUco marker is 25.2 cm away from the camera. The larger ArUco marker (10×10 cm) is further away, around 40.3 cm.



(c) Depth and size estimation evaluation Scene 3. The smaller ArUco marker (5×5 cm) is 46.2 cm away from the camera. The second marker (10×10 cm) is placed closer, 30.0 cm away from the camera.

Fig. 8. Comparison of three distinct scenes featuring ArUco markers of different sizes (10 × 10 cm and 5 × 5 cm) positioned at varying distances from the camera.

Table 7
Overview of evaluation scenes and ArUco marker distances from the camera (cm).

Scene	Marker	Distance (cm)
Scene 1	10 × 10 cm	31.2
	5 × 5 cm	31.2
Scene 2	10 × 10 cm	25.2
	5 × 5 cm	40.3
Scene 3	10 × 10 cm	30.0
	5 × 5 cm	46.2

19:00). To reduce the processing of videos with no fish in them, all videos were captured with the motion detection feature turned on. Figs. 9 and 10 depict amount of footage that was extracted during each day, classified into morning (08:00–09:00), afternoon (12:00–13:00) and evening (18:00–19:00) periods. In Bragado, the majority of the footage is derived from the afternoon and evening periods, with the least amount of footage was extracted during the morning hours. A similar outcome was observed at Covas do Barroso, where afternoons yielded the most footage and mornings the least. Overall Bragado encompasses substantially more footage than Covas do Barroso by a considerable margin.

The evaluation of the fish population involved analyzing the size of each individual fish in every frame of the video. Fish sizes were first estimated and subsequently processed using a thresholding filter to classify them as either adult or juvenile. Fish measuring less than 10 cm in length were categorized as juveniles. Following this classification, the number of juvenile and adult fish was counted and averaged per second of the video. The outcomes of the hydropeaking analysis are presented and discussed in Section 3.4.

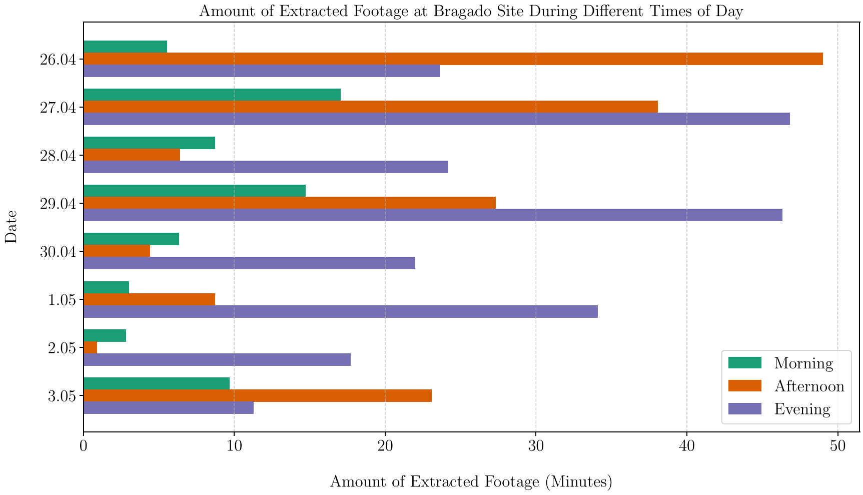


Fig. 9. Daily duration of Morning, Afternoon, and Evening sessions at Bragado site, illustrating the fluctuation of activity times across each day from April 26 to May 3. Afternoon sessions generally show higher durations compared to Morning and Evening sessions.

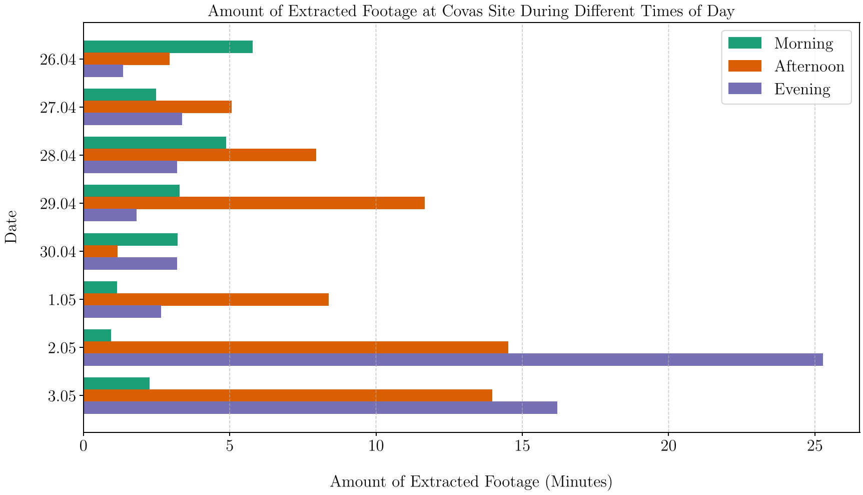


Fig. 10. Comparison of time spent in Morning, Afternoon, and Evening sessions at Covas do Barroso site over eight days from April 26 to May 3. Afternoon sessions tend to be longer, while both Morning and Evening times show variability, particularly with a notable increase in Evening activity on May 2 and May 3.

2.6. Evaluation metrics

2.6.1. Object detection

The Intersection over Union (IoU), or Jaccard index quantifies the percent overlap between the target mask and the prediction output. The IoU in this work represents the number of overlapping pixels between the target and prediction masks divided by the total number of pixels across both masks and is computed as:

$$\mathrm{IoU} = \frac{target \cap prediction}{target \cup prediction} \tag{3}$$

The intersection ($target \cap prediction$) consists of all pixels in the prediction and ground truth masks. In contrast, the union ($target \cup prediction$) includes all pixels in the prediction or target mask. The global mean IoU score is calculated for each class individually. The result is then averaged over all classes. Standard methods to evaluate object detection model accuracy also include the precision and recall, which are calculated based on the comparison of the following three possible outcomes:

- True positive (TP) — Correct detection, where the predicted bounding box matches with the ground truth
- False positive (FP) — Incorrect detection, where the predicted bounding box does not match with the ground truth
- False negative (FN) — The bounding box was not detected

The precision describes the relationship between the number of TP against the sum of the TP and the FP. It is used to describe the ability of a model to identify only the relevant objects. The precision was calculated as:

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{4}$$

Recall describes the relation between the number of TP and the sum of TP and FN, which is the model's ability to find all the ground

Table 8
Performance summary after validation for each of the five trained models.

Metric	Model 1	Model 2	Model 3	Model 4	Model 5
Precision	0.83	0.86	0.85	0.86	0.84
Recall	0.82	0.84	0.79	0.81	0.84
F1 Score	0.82	0.85	0.82	0.83	0.84
mAP@.5	0.85	0.88	0.83	0.85	0.87
mAP@.5...0.95	0.48	0.50	0.47	0.48	0.49

truth bounding boxes. It is the ratio of true positives from all ground truth bounding boxes. The following equation was used to compute the recall:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5}$$

The F1 score is a measure of a classification model's accuracy, calculated as the harmonic mean of precision and recall:

$$\text{F1 Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

Precision and recall are useful parameters for assessing class imbalanced datasets, which are especially common when evaluating field applications in computer vision. The precision provides a measure of result relevancy, whereas the recall reports how many relevant results are returned. This can be visualized using a precision–recall curve which shows the trade-off between the two over a range of threshold values. A high area under the curve represents both high recall and high precision, where high precision relates to a low false positive rate, and high recall relates to a low false negative rate. High scores for both show that the classifier returns accurate results (high precision) for the majority of all positive results (high recall). The average precision (AP) further summarizes the precision–recall curve into a single value calculated as:

$$\text{AP} = \sum_{n} (R_n - R_{n-1}) P_n \tag{7}$$

where R_n and P_n are the precision and recall at the nth threshold. The AP is calculated for each class individually across all of the IoU thresholds. The mean average precision (mAP) is a parameter used to summarize the performance across all vehicle classes using the following equation:

$$\text{mAP} = \frac{1}{\text{N}} \sum_{i=1}^{N} AP_i \tag{8}$$

where N is the number of all classes and AP_i is the average precision of a given class, i. The mAP metric is usually reported as mAP@0.5:0.95, which corresponds to an IoU threshold ranging from 0.5 to 0.95 with 0.05 as the increment size.

3. Results

3.1. Fish detection

The evaluation of the object detection model for fish detection was performed using a hold-out validation dataset composed of 3705 frames. As described in Fig. 3(b), the curated dataset was split using a ratio of 80:20. The first portion of the split dataset was used to train and test the model, using a randomizer to generate a new training and testing data for each run. The remaining portion of the dataset was set aside to validate the performance and to determine the best-performing model (see Fig. 3(c)). Table 8 summarizes the performance metrics of five trained models based on validation results. Among these, Model 2 demonstrated the highest performance, achieving an F1 score of 0.85 and an mAP@0.5 of 0.88.

3.2. Depth and size estimation

As shown in Tables 9 and 10, the results demonstrate accurate depth estimation across all scenes. The findings indicate that the estimation error is generally low. Notably, the 5 × 5 cm marker achieves a lower relative error in certain scenarios (e.g., Scene 1 and Scene 3), whereas the 10 × 10 cm marker exhibits higher relative accuracy in other cases (e.g., Scene 2). Overall, the 10 × 10 cm marker achieves a mean absolute error of 0.83 cm with a relative error of 2.74%. In comparison, the 5 × 5 cm marker offers slightly better performance, achieving a mean absolute error of 0.63 cm and a relative error of 1.54%.

Table 11 provides a comprehensive comparison of the estimated width and height of large and small ArUco markers across three distinct scenes, highlighting the corresponding absolute and relative errors. In Scene 1, the 10 × 10 cm marker demonstrates a notable absolute error in width estimation (0.97 cm, 9.70% relative error) while achieving higher accuracy in height estimation (0.08 cm, 0.80%). In contrast, the 5 × 5 cm marker exhibits lower errors for width (0.10 cm, 2.00%) and slightly elevated errors for height (0.29 cm, 5.80%). Across all scenes, the 5 × 5 cm markers consistently achieve superior performance with reduced relative errors compared to the 10 × 10 cm markers. Notably, in Scene 3, the 5 × 5 cm marker achieves the lowest relative errors for width and height, at 3.00% and 2.20%, respectively.

The 10 × 10 cm marker exhibits greater inaccuracies in width estimation, with a mean absolute error of 0.78 cm and a relative error of 7.10%, compared to height estimation, which achieves a mean absolute error of 0.31 cm and a relative error of 2.40%. In contrast, the 5 × 5 cm marker demonstrates superior precision in width estimation, with a mean absolute error of 0.12 cm and a relative error of 2.33%, while height estimation records a slightly higher mean absolute error of 0.27 cm and a relative error of 5.47%.

3.3. Hardware evaluation

As outlined in Section 2.1.1, we implemented and assessed the automated non-invasive fish detection and size estimation approach on low-power embedded hardware. The results presented in Table 13 demonstrate that the chosen hardware can process the information at a rate of 10 fps.

3.4. Hydropeaking analysis

The analysis of hydropeaking effects on fish populations yielded significant insights. Initially the absence of fish in Covas raised concerns about the impact of hydropeaking in that area. In contrast, analyzed footage from Bragado displayed a notable presence of fish as depicted in Fig. 11, indicating the utility of the refuge system into which the camera was installed. According to the initial study conducted from EcoPeak4Fish (Boavida et al., 2022), the authors expected the opposite result. Upon further examination of the Covas footage, it became apparent that the fish were present, but were frequently not located within the refuge. This finding emphasizes the importance of considering fish behavior and environmental context when assessing the impacts of hydropeaking on aquatic life.

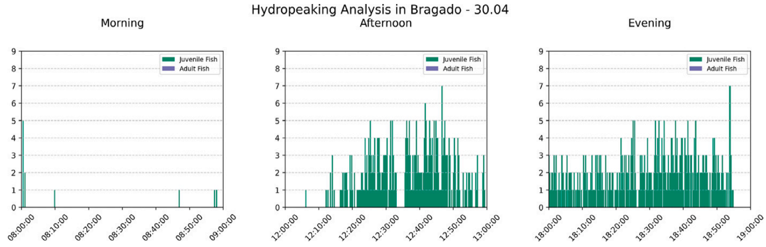


Fig. 11. Hydropeaking analysis in Bragado on 30th of April. The figure illustrates the distribution of juvenile and adult fish activity during morning (08:00–09:00), afternoon (12:00–13:00), and evening (18:00–19:00) period. Increased activity, particularly of juvenile fish, is predominantly observed during afternoon and evening.

Table 9
Comparison of real distances, estimated distances, and associated errors for two marker sizes (10 × 10 cm and 5 × 5 cm) across three different scenes.

Scene	Marker	Real distance (cm)	Estimated distance (cm)	Absolute error (cm)	Relative error (%)
Scene 1	10 × 10 cm	31.2	32.0	0.8	2.56
	5 × 5 cm	31.2	31.3	0.1	0.32
Scene 2	10 × 10 cm	25.2	25.2	0.0	0.00
	5 × 5 cm	40.3	39.0	1.3	3.22
Scene 3	10 × 10 cm	30.0	28.3	1.7	5.67
	5 × 5 cm	46.2	45.7	0.5	1.08

Table 10
Summary of mean absolute and relative errors for ArUco markers.

Marker	Mean absolute error (cm)	Mean relative error (%)
10 × 10 cm	0.83	2.74
5 × 5 cm	0.63	1.54

4. Discussion

The primary contribution of this work is a real-time, non-invasive fish detection and size estimation computer vision pipeline running on low-power embedded hardware. The system was implemented and evaluated to study the effects of hydropeaking at two sites: Bragado, located in Avelames River, and Covas Do Barroso in Couto River, both tributaries of the Tamega River (Douro River basin), Portugal. Previous works (Silva et al., 2024; da Silva Vale et al., 2020; Shi et al., 2020; Ubina et al., 2022; Li et al., 2024) have shown that performing size estimation with a stereo camera configuration can produce accurate results, but did not consider the complexities of real-time analysis in a river undergoing rapid changes in flow conditions. A recent work published by researchers at National Taiwan Ocean University (Ubina et al., 2022) showed how stereo cameras can accurately estimate the fish size, with the maximum error being only 5.52%. These results are comparable with our findings. In an uncontrolled environment, across three different scenes, our approach for size estimation had a mean absolute error below 1 cm (see Table 12). It should be noted that in most situations, having a sub cm accuracy is not necessary. For freshwater species, the size class in 5 cm increments is typically used (Tuhtan et al., 2022). In this work, we also illustrate that the majority of published research regarding fish size estimation is carried out in controlled environments, which can restrict the applicability of the findings to more dynamic natural settings (Yu et al., 2024; Shi et al., 2022; Wang et al., 2024b; Gao et al., 2024; Muñoz-Benavent et al., 2018), where our proposed work focused solely on real-time, in-situ applications.

The proposed pipeline was shown to be accurate as well as computationally lightweight by combining machine learning methods with data-driven methods, allowing it to be deployed on low-power embedded hardware. While more sophisticated modeling approaches exist for creating a disparity map, a significant downside is their computational complexity, which stems from the need to carry out a pixel-by-pixel comparison (Tosi et al., 2024; Ming et al., 2021). This makes them nonviable for systems with limited computational capabilities. Our proposed approach was able to process video at a frame rate of 10 fps, which is substantially faster than a trained biologist (Boavida et al., 2023). As detailed in Section 2.1.3, the system can be deployed in environments with limited access to infrastructure (connectivity and power supply), as the system can be powered via battery or solar power, extending the deployments to remote locations.

The developed system was utilized to investigate the effects of hydropeaking. As previously pointed out, fluctuations caused by hydropeaking can significantly alter flow patterns, water temperature, sediment transport, and dissolved gas levels within downstream ecosystems (Bipa et al., 2024; He et al., 2024). In fish communities, it can have a range of negative effects on fish populations. These include displacement (Alexandre et al., 2016; Rocaspana et al., 2019; Auer et al., 2023), restricted access to low-flow refuges (Moreira et al., 2019), and increased energetic costs that in the longer term may impair growth and reproductive fitness (Bipa et al., 2024; Kelly et al., 2017). Thus reducing survival rates due to stranding and disrupt critical life-cycle events such as growth, reproductive migrations, and spawning. Furthermore, hydropeaking contributes to habitat homogenization (Boavida et al., 2015; Jelovica et al., 2023), reducing the availability of microhabitats. Indirectly, it can alter trophic dynamics by reducing benthic invertebrate abundance (Bruno et al., 2016; Sabo et al., 2018), thus affecting fish foraging behavior. In our study, these hydropeaking-related dynamics likely contributed to lower detection rates during high-flow periods, highlighting the importance of interpreting monitoring outcomes in the context of altered flow regimes.

The week-long analysis conducted in Covas Do Barrosso and Bragado revealed that hydropeaking is likely to affect the utilization of flow refuges by fish. During peak flow events, fish were less frequently detected in these refuges, indicating that the resulting flow conditions created habitat instability and hindered access to these essential areas. However, it should be noted that a week long review might be insufficient to make definitive conclusions.

Juvenile fish appeared particularly vulnerable, as evidenced by reduced detection rates, suggesting potential risks to their recruitment and survival. In contrast, fish residing in the non-hydropeaking river consistently utilized refuge areas, underscoring their potential significance as shelter during natural flow variations. This outcome study

Table 11
Comparison of estimated width and height for ArUco markers with errors.

Scene	Marker	Width			Height		
		Estimated (cm)	Abs. Error (cm)	Rel. Error (%)	Estimated (cm)	Abs. Error (cm)	Rel. Error (%)
Scene 1	10 × 10 cm	9.03	0.97	9.70	9.92	0.08	0.80
	5 × 5 cm	4.90	0.10	2.00	4.71	0.29	5.80
Scene 2	10 × 10 cm	9.71	0.29	2.90	9.91	0.09	0.90
	5 × 5 cm	4.90	0.10	2.00	4.58	0.42	8.40
Scene 3	10 × 10 cm	8.93	1.07	10.70	9.25	0.75	7.50
	5 × 5 cm	5.15	0.15	3.00	4.89	0.11	2.20

Table 12
Summary of mean absolute and relative errors for ArUco markers.

Marker	Dimension	Mean absolute error (cm)	Mean relative error (%)
10 × 10 cm	Width	0.78	7.10
	Height	0.31	2.40
5 × 5 cm	Width	0.12	2.33
	Height	0.27	5.47

Table 13
Overview of performance and efficiency on selected low-power embedded hardware.

Metric	Nvidia Jetson Orin Nano
Frames per second	10
Power consumption (W)	10.7
Performance per Watt	0.93

highlights the value of automated size assessment systems for fish monitoring while providing new ecological insights into the impacts of hydropeaking (Boavida et al., 2023). This is because without automated methods, it is not feasible for human raters to process the videos at high temporal resolution, which can be as fine-grained as fish counts per minute. Nonetheless, it is also important to acknowledge that foreign objects or the presence of other fish can obstruct the view, causing partial occlusion and complicating precise length measurements. Environmental factors such as low luminosity, low-contrast regions, air bubbles, turbidity, periphytic biofilm, and light overexposure may further exacerbate these challenges.

Although further gains in precision are always preferable, the achieved performance was sufficient to meet the ecological objectives of the study. Specifically, the detection and size estimation results allowed us to interpret patterns of fish use of flow refuges during hydropeaking. In this context, the observed precision and low size estimation error support ecological insights, such as whether smaller individuals are more vulnerable or more likely to occupy refuge habitats. Nevertheless, we acknowledge that improving detection robustness would enhance the system's transferability to other river types and fish species. However, the model comparison already points to future developments to expand the scope and reliability of automated fish monitoring under highly fluctuating flow regimes. The system presented in this study also offers the advantage of unbiased sampling, capturing natural fish behavior. While this may result in lower performance, the observed detections could be closely linked to flow variability associated with hydropeaking. To address this, deploying multiple devices could help compensate for the low fish densities and improve data robustness.

The second contribution of this work is the PTFish dataset, an openly accessible multi-modal resource featuring both infrared (IR) and RGB imagery. The primary motivation for developing the PTFish dataset stems from the evident scarcity of multi-modal datasets specifically designed for size estimation tasks. Existing open-access datasets such as DeepFish (Saleh et al., 2020), Fish4Knowledge (Fisher et al., 2016), and AffiNe (Jorrit Venema, 2021) (as summarized in Table 1) exhibit notable limitations, particularly regarding their lack of infrared and color imagery, a gap that PTFish aims to partially address. Furthermore, many datasets focus solely on marine species, rendering them unsuitable for freshwater environments. Some datasets include samples collected in laboratory settings (in vitro) (Anantharajah et al., 2014), while others comprise ex vivo samples from deceased species (Pedersen et al., 2019; Jorrit Venema, 2021).

Despite its novelty and advantages over existing datasets, PTFish does have certain drawbacks. The most prominent limitations include a restricted number of species and an imbalance in the samples collected from the Bragado and Covas Do Barroso regions as shown in Table 5. Nevertheless, the dataset holds significant value and it is planned to increase it over time as additional sites are incorporated, thereby increasing both the overall sample count and the diversity of the represented species.

5. Conclusion

The results of this study demonstrate how an underwater stereo camera system can be successfully applied as a non-invasive method to fish detection and size estimation in complex hydrodynamic conditions characteristic of hydropeaking rivers. These systems can provide new insights into potential size-specific vulnerabilities and habitat preferences in these highly variable river ecosystems.

Another important contribution of this study is the public availability of the PTFish dataset, which contains 18,523 manually annotated frames, supporting the development of fish detection, identification, and size estimation methods.

Future research will focus on testing the binocular cameras in a broader range of river sites, extending the openly available datasets to foster research into methods for fish species identification, habitat use, and even swimming patterns, improving monitoring efficiency.

CRediT authorship contribution statement

Jürgen Soom: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Isabel Boavida:** Validation, Project administration, Conceptualization. **Renan Leite:** Data curation. **Maria João Costa:** Writing – review & editing, Validation, Conceptualization. **Gert Toming:** Conceptualization. **Mairo Leier:** Supervision. **Jeffrey A. Tuhtan:** Supervision, Project administration, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Isabel Boavida reports financial support was provided by Fundacao para a Ciencia e a Tecnologia (FCT). Isabel Boavida reports financial support was provided by Fundacao para a Ciencia e a Tecnologia (FCT). Renan Leite reports financial support was provided by Fundacao para a Ciencia e a Tecnologia (FCT). If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This study received funding from Fundacao para a Ciencia e a Tecnologia (FCT), Portugal through the EcoPeak4Fish project (DOI: 10.54499/PTDC/EAM-AMB/4531/2020). The authors are also grateful to FCT for its support through funding (DOI: 10.54499/UIDB/04625/2020) from the research unit Civil Engineering Research and Innovation for Sustainability (CERIS). Jeffrey Tuhtan and Gert Toming's contributions were funded by the Estonian Research Council Grant PRG 2198.

Data availability

Dataset used in this proposed work can be found at Zenodo: 10.5281/zenodo.14519903, published under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (CC BY-NC 4.0) (Commons, 2024).

References

Alexandre, C.M., Almeida, P.R., Neves, T., Mateus, C.S., Costa, J.L., Quintella, B.R., 2016. Effects of flow regulation on the movement patterns and habitat use of a potamodromous cyprinid species. Ecohydrology 9 (2), 326–340. http://dx.doi.org/10.1002/eco.1638, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/eco.1638 URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/eco.1638.

Álvarez-Ellacuría, A., Palmer, M., Catalán, I.A., Lisani, J.-L., 2019. Image-based, unsupervised estimation of fish size from commercial landings using deep learning. ICES J. Mar. Sci. 77 (4), 1330–1339. http://dx.doi.org/10.1093/icesjms/fsz216.

Anantharajah, K., Ge, Z.Y., McCool, C., Denman, S., Fookes, C., Corke, P., Tjondronegoro, D., Sridharan, S., 2014. Local inter-session variability modelling for object classification. pp. 309–316. http://dx.doi.org/10.1109/WACV.2014.6836084, URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84904704882&doi=10.1109%2fWACV.2014.6836084&partnerID=40&md5=0e3972f3e52c84c6df4aed4b0a89298cCited by: 61; All Open Access, Green Open Access,

Auer, S., Hayes, D.S., Führer, S., Zeiringer, B., Schmutz, S., 2023. Effects of cold and warm thermopeaking on drift and stranding of juvenile European grayling (thymallus thymallus). River Res. Appl. 39 (3), 401–411. http://dx.doi.org/10.1002/rra.4077, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/rra.4077 URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rra.4077.

Bejarano, M.D., Jansson, R., Nilsson, C., 2018. The effects of hydropeaking on riverine plants: a review. Biological Rev. 93 (1), 658–673. http://dx.doi.org/10.1111/brv.12362, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/brv.12362 URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/brv.12362.

Bipa, N.J., Stradiotti, G., Righetti, M., Pisaturo, G.R., 2024. Impacts of hydropeaking: A systematic review. Sci. Total Environ. 912, 169251. http://dx.doi.org/10.1016/j.scitotenv.2023.169251, URL: https://www.sciencedirect.com/science/article/pii/S0048969723078816.

Boavida, I., Costa, M.J., Portela, M.M., Godinho, F., Tuhtan, J., Pinheiro, A., 2023. Do cyprinid fish use lateral flow-refuges during hydropeaking? River Res. Appl. 39 (3), 554–560. http://dx.doi.org/10.1002/rra.3863, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/rra.3863 URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rra.3863.

Boavida, I., Santos, J., Costa, M.J., Leite, R., Portela, M., Godinho, F., Leitão, P., Mota, R., Tuhtan, J., Pinheiro, A., 2022. The EcoPeak4Fish project: an integrated approach to support self-sustaining fish populations downstream hydropower plants. 39th IAHR World Congr. http://dx.doi.org/10.3850/IAHR-39WC2521716X20221160.

Boavida, I., Santos, J.M., Ferreira, T., Pinheiro, A., 2015. Barbel habitat alterations due to hydropeaking. J. Hydro- Environ. Res. 9 (2), 237–247. http://dx.doi.org/10.1016/j.jher.2014.07.009, URL: https://www.sciencedirect.com/science/article/pii/S1570644314000926 Special Issue on Environmental Hydraulics.

Bravata, N., Kelly, D., Eickholt, J., Bryan, J., Miehls, S., Zielinski, D., 2020. Applications of deep convolutional neural networks to predict length, circumference, and weight from mostly dewatered images of fish. Ecol. Evol. 10 (17), 9313–9325. http://dx.doi.org/10.1002/ece3.6618.

Bruno, M.C., Cashman, M.J., Maiolini, B., Biffi, S., Zolezzi, G., 2016. Responses of benthic invertebrates to repeated hydropeaking in semi-natural flume simulations. Ecohydrology 9 (1), 68–82. http://dx.doi.org/10.1002/eco.1611, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/eco.1611 URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/eco.1611.

Commons, C., 2024. CC by 4.0 deed | attribution 4.0 international | creative commons — creativecommons.org. (Accessed 08 June 2024) https://creativecommons.org/licenses/by/4.0/deed.en.

Cutter, G., Stierhoff, K., Zeng, J., 2015. Automated detection of rockfish in unconstrained underwater videos using haar cascades and a new image dataset: Labeled fishes in the wild. In: 2015 IEEE Winter Applications and Computer Vision Workshops. pp. 57–62. http://dx.doi.org/10.1109/WACVW.2015.11.

da Silva Vale, R.T., Ueda, E.K., Takimoto, R.Y., de Castro Martins, T., 2020. Fish volume monitoring using stereo vision for fish farms. IFAC- Pap. 53 (2), 15824–15828. http://dx.doi.org/10.1016/j.ifacol.2020.12.232, URL: https://www.sciencedirect.com/science/article/pii/S240589632030505X21st IFAC World Congress.

Enders, E.C., Watkinson, D.A., Ghamry, H., Mills, K.H., Franzin, W.G., 2017. Fish age and size distributions and species composition in a large, hydropeaking prairie river. River Res. Appl. 33 (8), 1246–1256. http://dx.doi.org/10.1002/rra.3173, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/rra.3173 URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rra.3173.

Fisher, R.B., Chen-Burger, Y.-H., Giordano, D., Hardman, L., Lin, F.-P., et al., 2016. Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data. vol. 104, Springer.

Froese, R., Thorson, J.T., Reyes Jr., R.B., 2014. A Bayesian approach for estimating length-weight relationships in fishes. J. Appl. Ichthyol. 30 (1), 78–85. http://dx.doi.org/10.1111/jai.12299, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/jai.12299 URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/jai.12299.

Gao, T., Xiong, Z., Li, Z., Huang, X., Liu, Y., Cai, K., 2024. Precise underwater fish measurement: A geometric approach leveraging medium regression. Comput. Electron. Agric. 221, 108932. http://dx.doi.org/10.1016/j.compag.2024.108932, URL: https://www.sciencedirect.com/science/article/pii/S0168169924003235.

Garcia, R., Prados, R., Quintana, J., Tempelaar, A., Gracias, N., Rosen, S., V̊agstøl, H., Løvall, K., 2019. Automatic segmentation of fish using deep learning with application to fish size measurement. ICES J. Mar. Sci. 77 (4), 1354–1366. http://dx.doi.org/10.1093/icesjms/fsz186.

Hamzah, R.A., Ibrahim, H., 2016. Literature survey on stereo vision disparity map algorithms. J. Sensors 2016 (1), 8742920. http://dx.doi.org/10.1155/2016/8742920, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1155/2016/8742920 URL: https://onlinelibrary.wiley.com/doi/abs/10.1155/2016/8742920.

He, F., Zarfl, C., Tockner, K., Olden, J.D., Campos, Z., Muniz, F., Svenning, J.-C., Jähnig, S.C., 2024. Hydropower impacts on riverine biodiversity. Nat. Rev. Earth Environ. URL: https://api.semanticscholar.org/CorpusID:273365616.

IEA, 2024a. Hydropower has a crucial role in accelerating clean energy transitions to achieve countries' climate ambitions securely - news - IEA — iea.org. (Accessed 08 May 2024) https://www.iea.org/news/hydropower-has-a-crucial-role-in-accelerating-clean-energy-transitions-to-achieve-countries-climate-ambitions-securely.

IEA, 2024b. Renewables - energy system - IEA — iea.org. (Accessed 08 May 2024) https://www.iea.org/energy-system/renewables.

IHA, 2024. Europe — hydropower.org. (Accessed 08 May 2024) https://www.hydropower.org/region-profiles/europe.

Jelovica, B., Marttila, H., Ashraf, F.B., Kløve, B., Torabi Haghighi, A., 2023. A probability-based model to quantify the impact of hydropeaking on habitat suitability in rivers. River Res. Appl. 39 (3), 490–500. http://dx.doi.org/10.1002/rra.4050, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/rra.4050 URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rra.4050.

Jocher, G., Chaurasia, A., Qiu, J., 2023. Ultralytics YOLO. URL: https://github.com/ultralytics/ultralytics.

Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., NanoCode012, Kwon, Y., Michael, K., TaoXie, Fang, J., imyhxy, Lorna, Yifu, Z., Wong, C., V, A., Montes, D., Wang, Z., Fati, C., Nadar, J., Laughing, UnglvKitDe, Sonck, V., tkianai, yxNONG, Skalski, P., Hogan, A., Nair, D., Strobel, M., Jain, M., 2022. ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation. http://dx.doi.org/10.5281/zenodo.7347926.

Jorrit Venema, 2021. Affine | angling freshwater fish netherlands. https://www.kaggle.com/datasets/jorritvenema/affine.

Karlsson, K., 2024. A hands-on guide to use network video recorders, internet protocol cameras, and deep learning models for dynamic monitoring of trout and salmon in small streams. Ecol. Evol. 14 (5), e11246. http://dx.doi.org/10.1002/ece3.11246, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/ece3.11246 URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/ece3.11246e11246 ECE-2024-01-00100.R1.

Kelly, B., Smokorowski, K.E., Power, M., 2017. Growth, condition and survival of three forage fish species exposed to two different experimental hydropeaking regimes in a regulated river. River Res. Appl. 33 (1), 50–62. http://dx.doi.org/10.1002/rra.3070, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/rra.3070 URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rra.3070.

Kevin M. Boswell, M.P.W., Jr., J.H.C., 2008. A semiautomated approach to estimating fish size, abundance, and behavior from dual-frequency identification sonar (DIDSON) data. North Am. J. Fish. Manag. 28 (3), 799–807. http://dx.doi.org/10.1577/M07-116.1, arXiv:https://doi.org/10.1577/M07-116.1.

Lazarevich, I., Grimaldi, M., Kumar, R., Mitra, S., Khan, S., Sah, S., 2023. YOLOBench: Benchmarking efficient object detectors on embedded systems. arXiv:2307.13901.

Leite, R., Costa, M.J., Mameri, D., Afonso, F., Pinheiro, A., Santos, J., Boavida, I., 2024. The hide-and-seek effect of pulsed-flows in a potamodromous cyprinid fish. Hydrobiologia 851, http://dx.doi.org/10.1007/s10750-024-05575-6.

Li, H., Zheng, R., Jiang, W., Man, X., Ma, X., 2024. Fish length estimation based on stereo vision and keypoint detection. In: 2024 36th Chinese Control and Decision Conference. CCDC, pp. 1747–1752. http://dx.doi.org/10.1109/CCDC62350.2024.10587541.

Ming, Y., Meng, X., Fan, C., Yu, H., 2021. Deep learning for monocular depth estimation: A review. Neurocomputing 438, 14–33. http://dx.doi.org/10.1016/j.neucom.2020.12.089, URL: https://www.sciencedirect.com/science/article/pii/S0925231220320014.

Mobotix, 2023. MOBOTIX S16b DualFlex technical specifications. (Accessed 08 June 2024) https://www.mobotix.com/sites/default/files/2023-08/Mx_TS_S16B_V1.09_EN.pdf.

Monkman, G., Hyder, K., Kaiserc, M., Vidal, F., 2019. Using machine vision to estimate fish length from images using regional convolutional neural networks. Methods Ecol. Evol. 10 (12), 2045–2056. http://dx.doi.org/10.1111/2041-210x.13282.

Moreira, M., Hayes, D.S., Boavida, I., Schletterer, M., Schmutz, S., Pinheiro, A., 2019. Ecologically-based criteria for hydropeaking mitigation: A review. Sci. Total Environ. 657, 1508–1522. http://dx.doi.org/10.1016/j.scitotenv.2018.12.107, URL: https://www.sciencedirect.com/science/article/pii/S0048969718349520.

Muñoz-Benavent, P., Andreu-García, G., Valiente-González, J.M., Atienza-Vanacloig, V., Puig-Pons, V., Espinosa, V., 2018. Enhanced fish bending model for automatic tuna sizing using computer vision. Comput. Electron. Agric. 150, 52–61. http://dx.doi.org/10.1016/j.compag.2018.04.005, URL: https://www.sciencedirect.com/science/article/pii/S0168169918300358.

Naudascher, R., Boes, R.M., Fernandez, V., Wittmann, J., Holzner, M., Vanzo, D., Silva, L.G., Stocker, R., 2024. Fine-scale movement response of juvenile brown trout to hydropeaking. Sci. Total Environ. 952, 175679. http://dx.doi.org/10.1016/j.scitotenv.2024.175679, URL: https://www.sciencedirect.com/science/article/pii/S0048969724058352.

Nvidia, 2024. NVIDIA jetson AGX orin — nvidia.com. (Accessed 29 July 2024) https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/.

Ouyang, H., 2024. DEYO: DETR with YOLO for end-to-end object detection. ArXiv, abs/2402.16370.

Pedersen, M., Haurum, J., Gade, R., Moeslund, T., Madsen, N., 2019. Detection of marine animals in a new underwater dataset with varying visibility.

Pope, K.L., Lochmann, S.E., Young, M.K., 2010. Methods for assessing fish populations. URL: https://api.semanticscholar.org/CorpusID:5655880.

Rocaspana, R., Aparicio, E., Palau-Ibars, A., Guillem, R., Alcaraz, C., 2019. Hydropeaking effects on movement patterns of brown trout (l.). River Res. Appl. 35 (6), 646–655. http://dx.doi.org/10.1002/rra.3432, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/rra.3432 URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rra.3432.

Sabo, J.L., Caron, M., Doucett, R., Dibble, K.L., Ruhi, A., Marks, J.C., Hungate, B.A., Kennedy, T.A., 2018. Pulsed flows, tributary inputs and food-web structure in a highly regulated river. J. Appl. Ecol. 55 (4), 1884–1895. http://dx.doi.org/10.1111/1365-2664.13109, arXiv:https://besjournals.onlinelibrary.wiley.com/doi/pdf/10.1111/1365-2664.13109 URL: https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/1365-2664.13109.

Saleh, A., Laradji, I.H., Konovalov, D.A., Bradley, M., Vazquez, D., Sheaves, M., 2020. A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis. Sci. Rep. 10 (1), 14671. http://dx.doi.org/10.1038/s41598-020-71639-x.

Schmutz, S., Sendzimir, J., 2018. Riverine Ecosystem Management: Science for Governing Towards a Sustainable Future. Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-73250-3.

scikit-learn developers, 2024. 3.1. Cross-validation: evaluating estimator performance — scikit-learn.org. (Accessed 10 June 2024) https://scikit-learn.org/stable/modules/cross_validation.html.

Shi, C., Wang, Q., He, X., Zhang, X., Li, D., 2020. An automatic method of fish length estimation using underwater stereo system based on LabVIEW. Comput. Electron. Agric. 173, 105419. http://dx.doi.org/10.1016/j.compag.2020.105419, URL: https://www.sciencedirect.com/science/article/pii/S0168169919325773.

Shi, C., Zhao, R., Liu, C., Li, D., 2022. Underwater fish mass estimation using pattern matching based on binocular system. Aquac. Eng. 99, 102285. http://dx.doi.org/10.1016/j.aquaeng.2022.102285, URL: https://www.sciencedirect.com/science/article/pii/S0144860922000619.

Silva, C., Aires, R., Rodrigues, F., 2024. A compact underwater stereo vision system for measuring fish. Aquac. Fish. 9 (6), 1000–1006. http://dx.doi.org/10.1016/j.aaf.2023.03.006, URL: https://www.sciencedirect.com/science/article/pii/S2468550X23000539.

Snyder, D., 2003. Electrofishing and its harmful effects on fish.

Soom, J., Pattanaik, V., Leier, M., Tuhtan, J.A., 2022. Environmentally adaptive fish or no-fish classification for river video fish counters using high-performance desktop and embedded hardware. Ecol. Informatics 72, 101817. http://dx.doi.org/10.1016/j.ecoinf.2022.101817, URL: https://www.sciencedirect.com/science/article/pii/S1574954122002679.

Tosi, F., Bartolomei, L., Poggi, M., 2024. A survey on deep stereo matching in the twenties. URL: https://api.semanticscholar.org/CorpusID:271088858.

Tseng, C.-H., Hsieh, C.-L., Kuo, Y.-F., 2020. Automatic measurement of the body length of harvested fish using convolutional neural networks. Biosyst. Eng. 189, 36–47. http://dx.doi.org/10.1016/j.biosystemseng.2019.11.002.

Tuhtan, J., Dubrovinskaya, E., Miasayedava, L., Pattanaik, V., Soom, J., Mockenhaupt, B., Schuetz, C., Haas, C., Thumser, P., 2022. Smart fish counter for monitoring species, size, migration behaviour and environmental conditions.

Ubina, N.A., Cheng, S.-C., Chang, C.-C., Cai, S.-Y., Lan, H.-Y., Lu, H.-Y., 2022. Intelligent underwater stereo camera design for fish metric estimation using reliable object matching. IEEE Access 10, 74605–74619. http://dx.doi.org/10.1109/ACCESS.2022.3185753.

U.S. Department of Energy, 2025. Benefits of hydropower. URL: https://www.energy.gov/eere/water/benefits-hydropower (Accessed: 13 February 2025).

U.S. Geological Survey, 2025. Hydroelectric power: Advantages of production and usage. URL: https://www.usgs.gov/special-topics/water-science-school/science/hydroelectric-power-advantages-production-and-usage (Accessed 13 February 2025).

Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., Han, J., Ding, G., 2024a. YOLOv10: Real-time end-to-end object detection. arXiv preprint arXiv:2405.14458.

Wang, G., Li, X., Yu, J., Xu, W., Akhter, M., Ji, S., Hao, Y., Li, D., 2024b. Stereo matching and 3D reconstruction with NeRF supervision for accurate weight estimation in free-swimming fish. Comput. Electron. Agric. 225, 109255. http://dx.doi.org/10.1016/j.compag.2024.109255, URL: https://www.sciencedirect.com/science/article/pii/S016816992400646X.

Yu, H., Song, H., Xu, L., Li, D., Chen, Y., 2024. SED-RCNN-BE: A SE-dual channel RCNN network optimized binocular estimation model for automatic size estimation of free swimming fish in aquaculture. Expert Syst. Appl. 255, 124519. http://dx.doi.org/10.1016/j.eswa.2024.124519, URL: https://www.sciencedirect.com/science/article/pii/S0957417424013861.

Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., Chen, J., 2024. DETRs beat YOLOs on real-time object detection. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 16965–16974. http://dx.doi.org/10.1109/CVPR52733.2024.01605.

Curriculum Vitae

1. Personal data

Name	Jürgen Soom
Date and place of birth	29 June 1994 Tallinn, Estonia
Nationality	Estonian

2. Contact information

Address	Tallinn University of Technology, Centre for Environmental Intelligence and Sensing, Department of Computer Systems, Akadeemia tee 15A, 12618 Tallinn, Estonia
E-mail	jurgen.soom@taltech.ee

3. Education

2020–2025...	Tallinn University of Technology, School of Information Technologies, Computer Systems and Engineering, [PhD]
2017–2020	Tallinn University of Technology, School of Information Technologies, Computer Systems, [MSc] *cum laude*
2014–2017	Tallinn University of Technology, Faculty of School of Information Technologies, Computer Systems, [BSc]

4. Language competence

Estonian	native
English	fluent
Russian	basic
Italian	basic

5. Professional employment

2024–...	DefSecIntel Solutions, Research-Engineer
2020–2024	Tallinn University of Technology, Early Stage Researcher
2017–2020	Tallinn University of Technology, Engineer

6. Voluntary work

2022–...	Toastmasters Tallinn

7. Computer skills

- Operating systems: Microsoft Windows, MacOS, Linux
- Document preparation: Microsoft Office, Libre Office, Latex, Overleaf
- Programming languages: Python, C, Matlab
- Scientific packages: Solid Edge, Matlab

8. Defended theses

- 2017, CAN bus based dashboard development for Formula Student, BSc, supervisor: Mairo Leier, Tallinn University of Technology, School of Information Technologies, Department of Computer Systems
- 2020, Smart Car Deck Sensor Network Development for Tallink Megastar, MSc, supervisor: Mairo Leier, Tallinn University of Technology, School of Information Technologies, Department of Computer Systems

9. Field of research

- Computer Vision
- Machine Learning
- Edge Computing

10. Scientific work
Journal Articles

1. J. Soom, V. Pattanaik, M. Leier, and J. A. Tuhtan. Environmentally adaptive fish or no-fish classification for river video fish counters using high-performance desktop and embedded hardware. *Ecological Informatics*, 72, 2022
2. J. Soom, M. Leier, K. Janson, and J. A. Tuhtan. Open urban mmwave radar and camera vehicle classification dataset for traffic monitoring. *IEEE Access*, 12:65128–65140, 2024
3. J. Soom, I. Boavida, R. Leite, M. J. Costa, G. Toming, M. Leier, and J. A. Tuhtan. Open real-time, non-invasive fish detection and size estimation utilizing binocular camera system in a portuguese river affected by hydropeaking. *Ecological Informatics*, 90, 2025

Elulookirjeldus

1. Isikuandmed

Nimi	Jürgen Soom
Sünniaeg ja -koht	29.06.1994, Tallinn, Eesti
Kodakondsus	Eesti

2. Kontaktandmed

Aadress	Tallinna Tehnikaülikool, Arvutisüsteemide instituut, Akadeemia tee 15A, 12618 Tallinn, Estonia
E-post	jurgen.soom@taltech.ee

3. Haridus

2020–2025...	Tallinna Tehnikaülikool, Infotehnoloogia teaduskond, Õppekava nimetus, doktoriõpe
2017–2020	Tallinna Tehnikaülikool, Infotehnoloogia teaduskond, Arvutisüsteemid, MSc *cum laude*
2014–2017	Tallinna Tehnikaülikool, Infotehnoloogia teaduskond, Arvutisüsteemid, BSc

4. Keelteoskus

eesti keel	emakeel
inglise keel	kõrgtase
vene keel	algtase
itaalia keel	algtase

5. Teenistuskäik

2024–...	DefSecIntel Solutions, Teadus-Insener
2020–2024	Tallinn University of Technology, Doktorant
2017–2020	Tallinn University of Technology, Insener

6. Vabatahtlik töö

2022–...	Toastmasters Tallinn

7. Arvutioskused

- Operatsioonisüsteemid: Microsoft Windows, MacOS, Linux
- Kontoritarkvara: Microsoft Office, Libre Office, Latex, Overleaf
- Programmeerimiskeeled: Python, C, Matlab
- Teadustarkvara paketid: Solid Edge, Matlab

8. Kaitstud lõputööd

- 2017, Tudengivormeli CAN liidesel põhinev näidikute ploki arendamine, BSc, juhendaja: Mairo Leier, Tallinn Tallinna Tehnikaülikool, Infotehnoloogia teaduskond, Arvutisüsteemide instituut
- 2020, Tallink Megastari targa autoteki sensorvõrgu arendus, MSc, juhendaja: Mairo Leier, Tallinn Tallinna Tehnikaülikool, Infotehnoloogia teaduskond, Arvutisüsteemide instituut

9. Teadustöö põhisuunad

- Masinnägemine
- Masinõpe
- Sardsüsteemid

10. Teadustegevus

Teadusartiklite, konverentsiteeside ja konverentsiettekannete loetelu on toodud ingliskeelse elulookirjelduse juures.

ISSN 2585-6901 (PDF)
ISBN 978-9916-80-458-2 (PDF)