



Sistemi e Tecnologie Industriali Intelligenti
per il Manifatturiero Avanzato
Consiglio Nazionale delle Ricerche

DRIM

Ph.D. Program of National Interest in Robotics and Intelligent Machines

Administrative Headquarters: Università di Genova

Autoencoders for Quality Control and Anomaly Detection in Smart and Autonomous Systems

by

Adriano Liso

Thesis submitted for the degree of *Doctor of Philosophy* (38° cycle)

December 2025

Dr. Vito Renò

Supervisor

Prof. Antonio Sgorbissa

Head of the PhD program

Thesis Jury:

Prof. César Domínguez Pérez, *Universidad de la Rioja*

External examiner

Dr. Stefano Mutti, *Scuola universitaria professionale della Svizzera italiana*

External examiner

Prof. Donato Impedovo, *Università degli Studi di Bari A. Moro*

External examiner



Dipartimento di Informatica, Bioingegneria, Robotica e Ingegneria dei Sistemi

Abstract

Anomaly detection plays a crucial role in the evolution of modern industry, where interconnected systems, pervasive sensing, and real-time analytics are central to achieving efficiency and resilience. In recent years, deep learning has been deeply studied and widely applied in the fields due to its powerful feature learning ability. The deep learning-based detection method integrates feature learning into the process of building model, so that the model automatically learns deeper features through massive data instead of designing custom feature extractor. This work is aimed to study, design and develop appropriate deep learning models to implement effective anomaly detection in industry 4.0 context, with a focus on robotics applications. The neural network model most extensively used in this work is based on the autoencoder architecture, which has proven effective across all the contexts in which it has been tested, due to its ability to perform dimensionality reduction and feature learning, both in unsupervised settings, where it captures intrinsic data representations without labeled information, and in supervised scenarios.

Preface

The rapid evolution of modern industry, often referred to as Industry 4.0, has led to the widespread digitalization of production systems and the integration of intelligent technologies within manufacturing environments. Industrial processes now rely heavily on a vast network of sensors and data acquisition systems capable of continuously monitoring operational conditions. As a result, massive volumes of heterogeneous data—ranging from vibration and acoustic signals to thermal, electromagnetic, and imaging data—are being generated. Extracting meaningful insights from such data streams is essential for ensuring process efficiency, predictive maintenance, and early fault detection. Traditional data analysis are often inadequate to handle this complexity, as they rely on prior knowledge of the system’s behavior and struggle to adapt to new or unforeseen conditions. In this context, artificial intelligence and, in particular, deep learning methods have emerged as powerful tools for discovering hidden patterns and deviations from normal behavior in complex datasets. These intelligent approaches allow for automatic feature extraction, robust generalization, and adaptive modeling of nonlinear relationships that are crucial for the identification of anomalies in industrial systems. Among the various deep learning architectures, autoencoders have gained particular attention due to their ability to learn compact and meaningful representations of data in an unsupervised or supervised manner. By training a network to reconstruct its input through a latent representation, autoencoders naturally capture the essential structure of the data distribution. When exposed to anomalous samples, their reconstruction error typically increases, making them highly effective for anomaly detection tasks. Moreover, their versatility allows them to be adapted to different data modalities—time series, images, and multidimensional sensor data—and to operate with limited or unbalanced labeled datasets, a common scenario in industrial environments. This thesis explores how deep learning models, and autoencoders in particular, can be effectively applied to anomaly detection across a variety of domains. Beyond industrial inspection and monitoring, such approaches have growing relevance in robotics. Robots equipped with intelligent sensing and perception capabilities can exploit latent representations learned by autoencoders to interpret environmental data, detect anomalies in their surroundings or internal states, and adapt their navigation or manipulation strategies accordingly. In this sense, deep learning-based

anomaly detection represents not only a tool for fault diagnosis but also a key enabler of cognitive, self-aware, and resilient robotic systems. The first research objective (RO1) has been devoted to the deep learning-based anomaly detection problem definition in industry 4.0, especially considering that this activity can be carried out using a huge amount of heterogeneous data coming from multiple sensors. The current literature has been extensively studied performing a complete literature review on the relevant aspects of anomaly detection. The second research objective (RO2) regards the design and development of techniques for characterizing and predicting anomalies, with particular emphasis of experimenting state of the art models as well as defining custom data processing algorithms and architectures. The third research objective (RO3) consists of the test of the developed models and algorithms using datasets acquired in a controlled environment such as the laboratory and/or small industrial pilots. These models have been trained and tested on real data directly coming from the field. This thesis is inspired by the mentioned research objectives and is structured as follows:

- **Chapter 1** entirely covers RO1 and examines the industrial sectors where anomaly detection plays a central role, the types of sensors employed for data acquisition, and the neural network models and methodological approaches that have had the greatest impact on the scientific community in recent years. The results of this analysis, published in [1] will serve as a foundation for adapting the identified methodologies to the various case studies presented in this thesis work.
- **Chapter 2** proposes a framework, published in [2] for detecting anomalies in linear aluminum welding. Specifically, the framework starts by acquiring the three-dimensional representation of the welding using a 3D laser profiler. Afterward, a one-class supervised approach is followed by training a deep autoencoder on non-defective data samples, allowing the model to learn a mapping function that successfully reconstructs non-defective weldings with a small reconstruction error while providing a large error for abnormal data samples. At the end of the chapter, a robotic application regarding the inspection of aeronautical components is also shown.
- **Chapter 3** explores a deep learning-based method for the detection of defects in composite materials. A composite laminate specimen containing six programmed defects as a test sample will be investigated. It will be shown that training deep learning-based models as autoencoders makes it possible to extract features that can be used to discern defective areas from non-defective ones in unsupervised fashion, as shown in [3].
- **Chapter 4** is centered on the development and application of autoencoder-based neural architectures for the detection of the content of water in the soil and sub-sequent integration in autonomous navigation tasks. By training the autoencoder in a supervised fashion, the latent variables are constrained to represent physically meaningful parameters, thus combining dimensionality reduction with interpretability [4]. In this case the framework is tested again synthetic

and real data.

- **Chapter 5** presents the conclusions of the thesis.

Contents

Abstract	iii
Preface	v
Acronyms	xi
1 Review of Deep Learning-Based Anomaly Detection Strategies in Industry 4.0	1
1.1 Introduction to anomaly detection	2
1.2 Contextual factors characterizing anomaly detection	3
1.2.1 First factor: Application area	4
1.2.2 Second factor: Sensing equipment	4
1.2.3 Third factor: Algorithms	5
1.3 Methodology of the survey	5
1.3.1 Research questions	5
1.3.2 Literature search strategy	6
1.3.3 Inclusion and exclusion criteria	6
1.4 Results and discussion of the literature review	8
1.4.1 Brief recap of autoencoders	9
1.4.2 Autoencoders for time series	10
1.4.3 Autoencoders for images	12
1.4.4 Autoencoders combined with neural networks	15
1.4.5 Discussion and challenges	17
1.5 Summary	19
2 Assessing Welding Anomalies via Non-Destructive Tests	21
2.1 Introduction to the use of AI in welding process	22
2.2 Experimental setup for acquiring welds' profiles	23

2.2.1	Dense autoencoder	25
2.2.2	Convolutional autoencoder	27
2.2.3	LSTM autoencoder	29
2.3	Evaluating metrics in one-class supervised approach	31
2.4	Application to robotics	32
2.5	Summary	32
3	Testing of Aircraft Components using Laser Ultrasonic Data	35
3.1	Overview on composite materials and ultrasonic testing	35
3.1.1	Ultrasonic Testing	35
3.1.2	Exploiting LUT and AI	37
3.1.3	Composite materials and defects detection	37
3.2	The proposed methodology	40
3.2.1	Data acquisition and preprocessing	41
3.2.2	Latent feature learning via autoencoders	43
3.2.3	Clustering and classification in the latent space	45
3.3	Experiments and Results	48
3.3.1	Models architectures	48
3.3.2	Experimental settings	48
3.3.3	Results	49
3.3.4	A comparison with Wavelet transform-based feature extraction	54
3.4	Summary	56
4	Ground Penetrating Radar for moisture estimation and navigation in agricultural soil	59
4.1	An overview about GPR	60
4.2	Creation of synthetic dataset	62
4.3	Modification to the loss function of the AE	63
4.4	Outcomes of the autoencoder trained with simulated data	64
4.5	Outcomes of the autoencoder trained with real data	65
4.6	DRL for robot navigation	69
4.7	Summary	70
5	Conclusions	75
	Bibliography	77

Acronyms

AE Autoencoder. x, 2, 8, 9, 10, 11, 16, 38, 39, 41, 43, 44, 45, 48, 49, 50, 51, 52, 56, 63

CFRP Carbon Fibre-Reinforced Plastic. 35, 37, 38, 40, 41

CNN Convolutional Neural Network. 5, 8, 12, 27, 37, 39, 61

CPS Cyber-Physical Systems. 1

CV Computer Vision. 38, 39

DBSCAN Density-Based Spatial Clustering of Applications with Noise. 37, 40, 45, 46, 47, 49, 50, 51, 53, 54, 56, 57

DL Deep Learning. 2, 3, 5, 8, 18, 39, 40

DRL Deep Reinforcement Learning. x, 60, 69

FDTD Finite-Difference Time-Domain. 59

GPR Ground Penetrating Radar. x, 59, 60, 61, 70, 71

GRU Gated Recurrent Unit. 12, 18, 29

IF Isolation Forest. 12

IoT Internet of Things. 1, 6

LSTM Long Short-Term Memory. 12, 18, 29

LUT Laser Ultrasonic Testing. x, 35, 37, 40, 54

ML Machine Learning. 3, 5, 18

MSE Mean-Squared Error. 12, 13, 63, 70

NDT Non-Destructive Testing. 21, 32, 35, 38, 39, 56, 58, 71

PCB Printed Circuit Board. 12, 22

RNN Recurrent Neural Network. 5, 29

RQ Research Questions. 5, 7

UT Ultrasonic Testing. 35, 36, 56, 58

VAE Variational Autoencoder. 10, 13, 42, 44, 45, 48, 49, 50, 51, 52, 55, 56

ZCA Zero-phase Component Analysis. 11

Chapter 1

Review of Deep Learning-Based Anomaly Detection Strategies in Industry 4.0

The industrial revolutions have successively transformed production: Industry 1.0 introduced mechanization via steam power; Industry 2.0 brought electrification and mass production; Industry 3.0 deployed digitalization, electronics, and basic automation; Industry 4.0 further integrates Cyber-Physical Systems (CPS), Internet of Things (IoT), big data, and artificial intelligence to create smart factories. The emerging concept of Industry 5.0 also emphasizes sustainability, human-machine collaboration, and socially conscious automation. Within Industry 4.0, the increasing complexity and volume of sensor-derived data create both opportunity and challenge: anomalous events (faults, defects, degradations) must be detected promptly to avoid production losses, safety hazards, and costly downtime. Artificial intelligence provides a promising framework to address these challenges, and this thesis is designed with that objective in mind. To this end, it was first necessary to conduct a systematic review of the field, examining the industrial sectors where anomaly detection plays a central role, the types of sensors employed for data acquisition, and the neural network models and methodological approaches that have had the greatest impact on the scientific community in recent years. The results of this analysis will serve as a foundation for adapting the identified methodologies to the various case studies presented in this thesis work.

1.1 Introduction to anomaly detection

Anomaly detection is a topic of interest in several areas, ranging from Industry 4.0 to Energy Management, Smart Agriculture, Cybersecurity, and Bioinformatics. In a wide sense, detecting anomalies implies finding samples generated within a process that differs from its standard data generation mechanisms. Identifying these samples is extremely important for a variety of reasons, depending on the specific application and scenario, ranging from the minimization of production costs to maintaining the required safety standards. As such, the increasing availability of wide networks of sensors that yield large amounts of data characterizing the processes under observation allowed the large adoption of Deep Learning (DL) techniques, which proved worthy of attention due to their capability of identifying anomalies with large precision, accuracy and reproducibility. Consequently, there is an extensive need to consolidate research results to provide a common framework to understand the topic and ensure a common foundation to establish future research trends. To respond to this need, this chapter systematically reviews the state of the art of anomaly detection in Industry 4.0, evaluating gaps in the current knowledge and proposing future directions of interest. To pursue this objective, three main dimensions have been considered: the scenario where the anomaly detection methodologies were applied, the sensing equipment used to gather data characterizing the underlying process, and the algorithm employed to properly interpret the phenomena. The study was conducted following the PRISMA protocol, which allowed the identification of a relevant selection of papers by extracting a meaningful dataset of 78 papers of interest. The analysis highlighted the diffusion of Autoencoders (AEs) in several configurations and application scenarios, highlighting their effectiveness and flexibility for anomaly detection. Anomaly detection refers to the problem of finding samples that differ from the normal distribution of data instances. Several factors, including system failures, human errors, malicious operations, or natural environmental changes, can cause anomalies. Thus, anomaly detection is essential in decision systems, as it reduces risks and costs associated with these unexpected events and, in some instances, can even prevent the failure of the critical parts of a system. In recent years, various fields and applications have widely used anomaly detection techniques. Examples include detection systems for precision agriculture and life monitoring of plants [5–7], fault detection in energy forecasting [8–24], real time fault detection in smart manufacturing [25–56], crack detection and structural monitoring [57–64], environmental monitoring [65–72], anomaly detection for vehicles and intelligent transportation systems [73–76], fault detectors for radar and hyperspectral data [77–80], and bioinformatics [81–83].

Most legacy fault detectors or monitoring systems were based on traditional signal processing and analysis, with sensors that raise the alarm when readings exceed a threshold, usually manually set

by domain experts. However, as the processes involved in these use cases become larger and more complex, traditional, handcrafted methods proved inadequate for proper modelling of the behaviour of the system.

Meanwhile, several concurring factors were introduced with the advent of Industry 4.0. First, wide networks of cheaper and more efficient sensors were deployed in real-world scenarios, thus providing larger quantities of data to characterize the processes under investigation. Finally, the improvements in the computational power in centralized servers and edge devices allowed the development, training, validation, and deployment of models with high representational capabilities. Consequently, the research community shifted its focus from traditional methods towards automatic learning techniques, exploiting the improved representational capability provided by Machine Learning (ML) and, more recently, DL models to deal with processes characterized by large amounts of complex data. As such, there is a need to consolidate the vast amount of methods, approaches, and results achieved by the research community, providing practitioners with a common ground upon which novel ideas, models and techniques can be developed.

To fill this gap, this work aims to review anomaly detection techniques applied to industrial scenarios. Research articles were read thoughtfully and selected following the PRISMA [84] protocol. Specifically, three main factors were first identified as the driving aspects leading innovations in the anomaly detection field. Then, three research questions were crafted, each dealing with one of the identified factors. From this background, a set of 78 papers was carefully selected and analyzed.

The remainder of this work is structured as follows. In Section 1.2, the three main factors leading the review work, namely application areas, data sources, and algorithms, are analyzed. Section 1.3 describes the methodology used to perform literature research for this review according to PRISMA principles. Then, Section 1.4 recaps the results of this review research, giving a deeper and more technical discussion of the most relevant algorithms and techniques used in most of the selected research papers. Finally, Section 1.5 concludes the chapter, giving future works perspectives.

1.2 Contextual factors characterizing anomaly detection

Given the strategic value of applications field related to anomaly detection, the scientific community has provided several advancements over the last few years, particularly by exploiting the power of deep learning methodologies. The selected contributions can be characterized and compared by three different factors.

1.2.1 First factor: Application area

The first factor on which the research in the anomaly detection area focused was the application area of the proposed detection method. Interestingly, most of the works are focused on three main areas of interest, that is, *smart manufacturing*, *energy management*, and *structural and environmental monitoring*. This is strictly related to the strategic relevance of these applications:

- *Smart manufacturing*: Real-time detection of anomalies in smart manufacturing, either on the production line or on the outcome, can help in cost reduction and waste mitigation from production lines, therefore improving the environmental impact and the sustainability of the manufacturing process.
- *Energy management*: Forecasting energy production and assessing anomalous consumption peaks can help identify losses in smart grids or possible misuse from malevolent users.
- *Structural and environmental monitoring*: Interpreting anomalous trends and spikes allows maintainers and decision-makers to quickly adapt to changes and critical situations. For example, maintainers can use anomaly detection systems to identify ongoing degradation in reinforced concrete infrastructures like bridges. Another example is when an anomalous trend is shown for specific environmental variables, which decision-makers can use to define policies to mitigate risks.

Let us mention that several works are also related to cybersecurity, medical applications, intelligent transport systems, and agriculture. However, this review did not consider these specific fields, as the discussion will focus on Industry 4.0-related challenges.

1.2.2 Second factor: Sensing equipment

The second factor worth highlighting in this review is the sensing equipment selected for data acquisition, mainly because it is useful to understand what hardware and enabling technologies researchers focused on over the last few years. There is a strict connection between the hardware choice and the application area, so this factor mainly depended on the specific operating context: for example, data acquired in the context of energy management systems were usually gathered directly from the SCADAs or PLCs connected to the generator (e.g., a wind turbine). In other instances, data were gathered using cameras operating in different domains, such as visible or hyperspectral cameras. It is also worth noting that if the work mainly focused on developing algorithms concerning a specific application domain, researchers used data synthesized via generative methods or acquired from publicly available datasets.

1.2.3 Third factor: Algorithms

The third and last factor on which research focused was the algorithms used for anomaly detection purposes. Most selected works used ML or DL approaches to implement the task. In that sense, the application and the data source led to selecting the specific algorithm. For example, in energy management, forecasting energy production over different periods can be the basis for evaluating an anomaly behavior of a certain generator (e.g., a slow decay in energy production by a wind turbine). Hence, most researchers focused on tools specific to time series prediction, such as linear predictors (e.g., ARIMA models and their variants) or predictors based on deep neural networks, such as Recurrent Neural Networks (RNNs). Another example was the visual inspection of manufactured samples, looking for defects and anomalies. In this case, researchers relied on traditional image processing techniques and DL tools, such as Convolutional Neural Networks (CNNs), to better exploit visual cues embedded in the acquired data.

1.3 Methodology of the survey

Given the three factors reported in the previous section, that refer to specific research dimensions, the methodology described hereafter was followed to systematically identify the most relevant works that refer to anomaly detection. This section describes the steps of the research method used in this systematic review, according to the PRISMA [84] protocol.

1.3.1 Research questions

First of all, the study focused on investigating the three relevant factors chosen and described in the previous section, aiming to understand better their influence on the research works of the last years. In particular, the following three specific Research Questions (RQs) were formulated in a 1 : 1 relationship with the relevant factors and then used as the basis of this systematic review:

- *About application fields - RQ1:* Which application fields mainly involved anomaly detection approaches under the general framework of Industry 4.0?
- *About sensing equipment - RQ2:* What sensing equipment is generally used to capture relevant data for anomaly detection purposes?
- *About algorithms - RQ3:* Which are the most relevant algorithms and techniques used to extract and predict the meaningful features for anomaly detection in different application fields?

As the literature analysis was designed over the PRISMA protocol [84], there was the need to define a *literature search strategy*, starting from a series of *inclusion and exclusion criteria*, followed by a *quality*

assessment, which finally led to *data extraction*. All these steps will be described in the next paragraphs.

1.3.2 Literature search strategy

The first step for defining the literature search strategy was to gather studies on anomaly detection by performing an extensive literature search on the Scopus database for the following reasons:

1. First, using more than one database for a literature search does not necessarily guarantee a positive impact on the research outcome. Using several databases may lead to duplicate papers and low-quality or non-relevant ones.
2. Second, Scopus is recognized as a high-reliability database for scientific literature. Hence, high-quality papers can be effectively found.

1.3.3 Inclusion and exclusion criteria

The papers have been selected by a combined inclusion/exclusion test starting from a first query that joined the keywords *anomaly*, *detection*, *deep*, *learning* using the AND logic operator. The research dates back to 2023 and produced a total of 5398 results. Then, after selecting only papers published after 2017 in international journals and written in English, 59% papers were excluded (2209/5398 papers remained). Since the goal of this review is to summarize those proposed approaches that have had a relevant impact on the scientific community, in accordance with the mentioned research questions, only papers from journals classified as Q1 at least once in the last years have been kept in the dataset. To this end, the "Scimago Journal & Country Rank" has been used to identify the information about the journals' best quartile, namely "SJR Best Quartile". Hence, Q2, Q3, and Q4 papers were discarded, resulting in 514 remaining papers. Among these, papers with several citations smaller than the first quartile of the correspondent publication year were also excluded, resulting in 381 papers. All 381 abstracts have been read to label each paper with its corresponding application field. Review papers have been discarded to avoid redundant reporting of overlapping contributions. About 23% of these papers relates to cybersecurity: different intrusion detection systems are proposed for detecting anomalous behaviors in video surveillance or for self-defending from threats hiding behind the huge growth of network traffic in domains such as IoT, 5g networks, smart grids, Industrial Control Systems. Another 15% of these papers relates to medical applications such as medical image analysis (with data from X-ray and magnetic resonance), ECG anomaly detection, daily human activity recognition, etc. A relevant number of authors try to detect anomalies in data coming from wearable sensors, fitting in a hybrid domain that is middle-way between the previous two, that is, the Internet Of Medical Things. Being interested in real industrial case studies where data from a huge variety of sensors is needed for

Table 1.1: Exclusion criteria

#	Exclusion criterion
1	Articles not written in English
2	Articles that are reviews
3	Articles that do not refer to anomaly detection
4	Articles published in journals with under-threshold impact factors (based on SJR quartiles)
5	Articles concerning anomaly detection in fields unrelated to Industry 4.0

Table 1.2: Inclusion criteria

#	Inclusion criterion
1	Articles are written in English
2	Research articles
3	Articles refer to anomaly detection with deep models
4	Articles published in journals with high impact factors (based on SJR quartiles)
5	Articles concerning anomaly detection in Industry 4.0

automation, papers that refer to these fields have not been considered. Rather, the focus was shifted towards fields such as Smart Manufacturing, Energy Forecasting, Structural health and environmental monitoring, or Intelligent transportation systems, which all together constitute the 25% of the 381 papers, namely 95 papers. So, more general purposes, even if they still refer to anomaly detection, have not been considered. The exclusion criteria are summarized in Table 1.1, while the inclusion ones (dual of the previous ones) are reported in Table 1.2.

Finally, table 1.3 highlights the process used to complete the data extraction step from the Scopus database. The process is described in terms of subsequent steps performed on the selected 95 papers. Hence, the paper analysis has been extended, going more deeply into detail about the three research questions and how the authors identified, modelled, and solved the problem to filter the papers critically. After further evaluating the relevance of the selected works with respect to the RQs implemented in the review, a final number of 77 papers remained.

Table 1.3: Recap of the subsequent steps performed on the selected papers

#	Step	Outcome
1	Paper reading	Yes/No (already performed)
2	Research question assessment	The clear description of the research question and the evaluation of whether the paper fits or not in the three RQs of this review
3	Type of article	Problem identification
4	Study outcomes	Short description of study outcomes
5	Critical assessment	

A visual representation of the protocol followed throughout this review work is shown in Figure 1.1, where the main topic addressed in this paper is represented in blue on top. From it, column-wise

details about the three research questions and the subtopics analyzed are represented using a different colour scheme depending on their granularity: green for the questions and yellow for the subtopics.

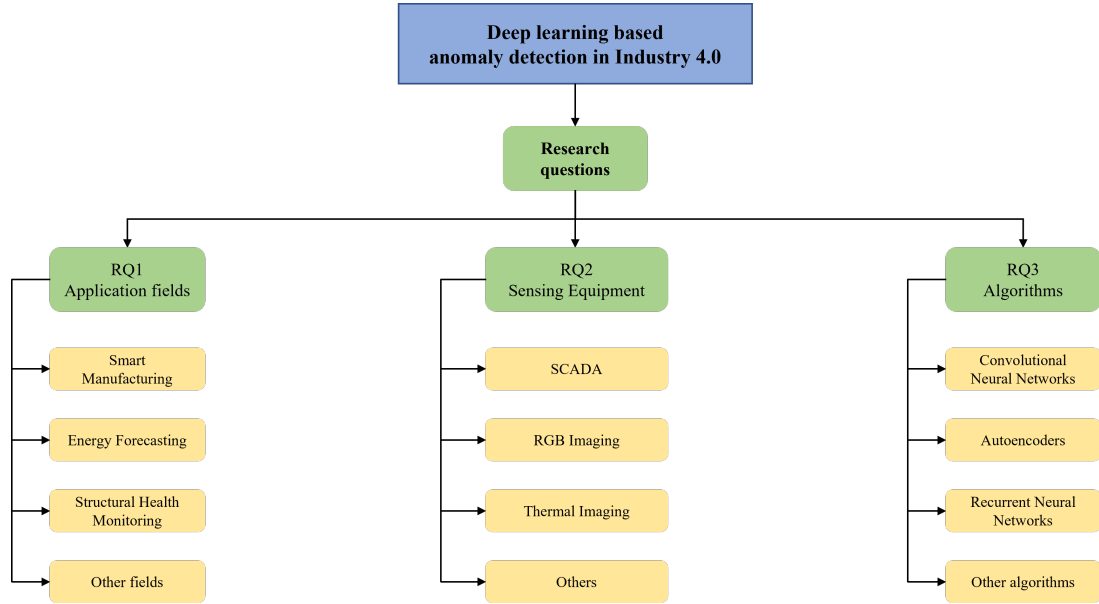


Figure 1.1: Visual representation of the scheme followed during the review. Starting from the topic of interest (i.e., DL anomaly detection in Industry 4.0), three research questions concerning application fields, sensing equipment, and algorithms led to identifying relevant works selected following the PRISMA protocol.

1.4 Results and discussion of the literature review

This section is dedicated to describing the results of the review of the selected papers, with a subsequent discussion about the relevant aspects that could be useful to answer the research questions posed critically. To perform this task, the section starts with a quantitative description of the selected works based on different grouping strategies, mainly related to the application area and algorithms/techniques exploited by the researchers. It is worth highlighting the following considerations:

- Table 1.4 shows the most relevant application areas the selected works consider. As it can be seen, most of the works are related to either the *Smart manufacturing* or the *Energy forecasting* application areas. This is mainly related to the high relevance of these topics in current society, whose main focus is improving the smart manufacturing system's overall throughput, reducing costs, and preserving the environment.
- Looking at Table 1.5, where a recap of the distribution of the paper based on the type of algorithm used is shown, more than 50% of the papers considered implement either an AE or a CNN, suggesting that the core of the subsequent description and discussion will be based on these approaches.

Table 1.4: Recap of the distribution of the paper based on their application area

Application area	% of papers
Agriculture	5.3
Energy forecasting	22.4
Smart manufacturing	38.2
Structural health monitoring	9.2
Anomaly detection	7.9
Environmental monitoring	6.6
Intrusion detection systems	3.9
Vehicles anomaly detection	6.6

Table 1.5: Recap of the distribution of the paper based on the type of algorithm used

Algorithm type	% of papers
Autoencoders	27.5
Artificial Neural Networks	4.9
Clustering	4.9
Convolutional Neural Networks	29.4
Generative Adversarial Networks	2
Recurrent Neural Networks	15.7
Graph Neural Networks	1
Support Vector Machines	4.9
Attention mechanism	3.9
Proprietary algorithms	1
Other ML/DL approaches	4.9

- In almost all selected papers, the datasets consist of time series (often acquired by SCADA systems) or images, with only a few of them using higher dimensional data.

For this reason, in the next paragraphs, a brief description of an autoencoder will be first given, followed by its declination on time series data and images, and finally, highlighting the combined approach of autoencoders coupled with other kinds of networks.

1.4.1 Brief recap of autoencoders

An autoencoder is an artificial neural network consisting of two parts: an encoder, which maps input data to a lower-dimensional representation called latent space, and a decoder, which restores encoded data to its original size. During the training phase, the AE learns to encode an input sample and decode it, minimizing reconstruction errors. The encoder and the decoder show a symmetric structure with multiple layers that reduce and enlarge the input. Each layer is followed by an activation function that introduces the non-linearity in the model. The bottleneck structure of an AE makes it particularly useful for dimensionality reduction and nonlinear feature learning with an unsupervised approach. In the anomaly detection context, where the number of anomalous samples is usually much smaller

than normal ones, such a neural network is used to identify the salient features that allow the decoder to reconstruct normal encoded samples with small reconstruction errors while failing to reconstruct abnormal inputs. The general scheme of an AE is shown in Figure 1.2.

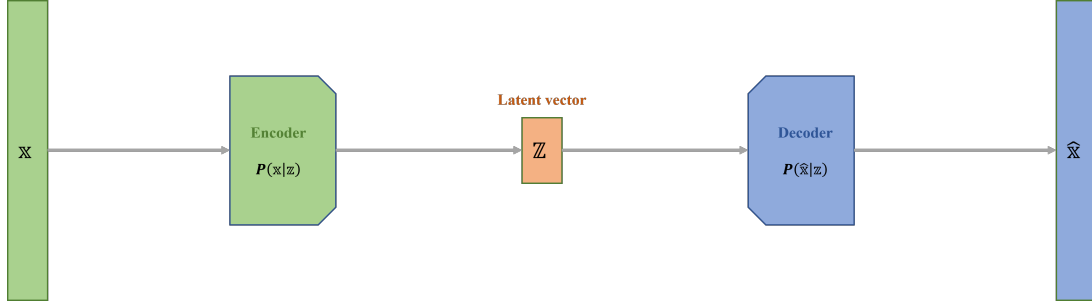


Figure 1.2: Working scheme of an autoencoder. The network is composed of an encoding part, which actively compresses the input $\mathbf{X}$, a latent space $\mathbf{Z}$, and a decoder, which attempts to reconstruct the input $\mathbf{X}$.

A commonly used variant of the standard, undercomplete autoencoder is the *Variational Autoencoder* (VAE), which attempts to map the provided input onto a distribution in the latent space instead of a single point. The working scheme of the VAE is reported in Figure 1.3.

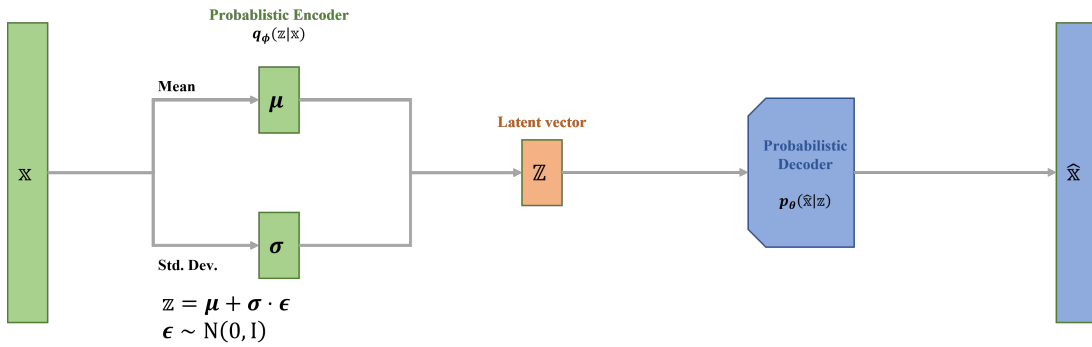


Figure 1.3: Working scheme of a variational autoencoder. A VAE differs from a undercomplete AE in the sense it projects $\mathbf{X}$ onto a distribution in the latent space $\mathbf{Z}$

1.4.2 Autoencoders for time series

Time series data comprises sequential observations collected at regular intervals. Different authors use time series data coming from SCADA systems, which stand for "Supervisory Control and Data Acquisition", even if, in some cases, scientists and engineers do not rely on such systems but prefer to directly sample data from the industrial plant or, more generally, from the field.

However, a SCADA system is generally used in industrial control systems to monitor and control processes, machinery, and infrastructure in various fields such as manufacturing, energy, and transportation. For this review, the works involving SCADA systems are interestingly related to the fault detec-

tion of wind turbines and are included in the "energy forecasting" domain. This monitoring system is installed by default on modern wind turbines and ensures a cost-effective solution for operators without additional measurement devices. Some key parameters typically monitored by SCADA include wind speed and direction, power output, rotor speed, and temperatures across different parts (gear-box, generator, and bearings) of the turbine. These data can be used to assess fault events, the health state of components, and for wind prediction, following a data-driven approach that exploits machine learning techniques and neural networks. This approach is more suitable and efficient than model-based methods, which use explicit system dynamic models and control theories to generate predictions and residuals for fault detection and isolation. In this context, the first comparison between different works concerns processing raw data from SCADA. In [11], for example, authors first normalize signals such that they have vanishing mean and unitary variance, then apply Zero-phase Component Analysis (ZCA) to decouple them from each other and obtain zero covariance. The ZCA is an algorithm closely related to PCA which transforms the vector x_t , whose components are the values of the signals selected from SCADA at time t , into $x_t^{ZCA} \equiv \Sigma^{-1/2}x_t$, where Σ is the estimated covariance matrix. Then, an autoencoder, consisting of fully connected layers, is fed with whitened vectors of signals from the normal operational conditions at different timestamps and reconstructs the input. Since the bottleneck structure of the AE forces it to learn only the most important features of the training data, anomalous test signals deviating from normal ones show large reconstruction errors and can be classified as faults. In this case, the reconstruction error, which is evaluated as mean-squared distance between input signal x_t^{ZCA} and reconstructed output $\hat{x}_t^{ZCA}$, is the Mahalanobis distance $\text{MD}(x_t, \hat{x}_t)$ between original sample x_t and $\hat{x}_t \equiv \Sigma^{1/2}\hat{x}_t^{ZCA}$:

$$(x_t^{ZCA} - \hat{x}_t^{ZCA})^2 = (x_t - \hat{x}_t)^T \Sigma^{-1} (x_t - \hat{x}_t) = \text{MD}(x_t, \hat{x}_t). \quad (1.1)$$

The threshold on the reconstruction error, which distinguishes between normal and abnormal behaviour, is calculated by averaging across signals. The authors also propose a post-processing step on output signals in the time domain consisting of a smoothing technique called exponentially weighted moving average. This additional step enables the procedure to capture even small shifts in the average signal without triggering alarms for big but rarely occurring spikes overwhelming the threshold.

An AE-based approach is also followed in [8]. Still, in this case, the autoencoder is composed of multiple Restricted Boltzmann Machines, and the signals do not undergo a complex pre-processing step, only normalized in the $[0, 1]$ interval. The reconstruction error of each sample is still estimated as the mean squared distance between the input vector of selected features and its reconstructed output. Unlike the previously described work, a non-stationary adaptive threshold is set based on extreme value

theory. In the other two selected works, [14] and [9], the anomaly detection task is confined to the pre-processing step. At the same time, the final goal is essentially the prediction of the signals of interest. The pre-processing consists of smoothing data by resampling over an appropriate period and removing outliers, exploiting a manual approach and the Isolation Forest (IF) algorithm. Feature engineering is performed to select relevant features from SCADA parameters, and a time series of correspondent signals is used to train a model for forecasting purposes. In both works, authors compare two models for the prediction task: in the first, XGBoost and Long Short-Term Memory (LSTM) are used, and in the second, Gated Recurrent Unit (GRU) and LSTM. XGBoost and GRU outperform the LSTM regarding computational cost and forecasting accuracy.

1.4.3 Autoencoders for images

Among papers whose datasets consist of images, the most used technique is generally based on exploiting CNNs. On the other hand, different works also introduce an encoder-decoder structure to identify anomalies in an end-to-end fashion or as a part of a bigger detection system. In most cases, the model is trained with normal samples, and pixel-wise anomaly maps are generated: each pixel presents an anomaly score responsible for the binary classification (normal or anomalous) of the pixel once an appropriate threshold is fixed. Different strategies are finally applied to find an overall anomaly score and a second threshold to classify the image, or a patch of the image, as normal or not.

In [57], the authors follow this approach to detect defects in concrete structures. The dataset consists of images of cracks acquired with an RGB camera. Then, multi-scale patches are extracted and fed to an autoencoder composed of several convolutional layers. The input samples are augmented via different types of geometric transformations (flipping and rotations) to enhance the learning process of invariant features. The number of channels of each layer is doubled during the encoding phase to augment the representation of the encoded features. Before transforming and storing feature representations in the code, the feature maps are flattened and down-sampled by a fully connected layer to render good coherences between features. During the decoding phase, the reverse mapping is obtained by concatenating transposed convolutional layers with several channels, each halved at every step. The Mean-Squared Error (MSE) is used as the loss function, and the anomaly maps are constructed by computing the pixel-wise squared difference between the inputs and outputs. The maximum anomaly score between pixels is extracted for each patch, and a threshold is set as the average of the maxima between all patches. A patch containing a pixel with an anomaly score above the threshold is classified as anomalous.

A similar strategy is pursued in [56]. In this case, the dataset consists of photos of Printed Circuit Boards (PCBs) acquired by an RGB camera. The approach assumes that the PCB is shown from an

overhead view. Then, the images undergo a registration step based on SIFT features and the RANSAC algorithm to correct planar distortions (the 3D appearance of the components is not considered). As before, an autoencoder presenting an architecture similar to the one previously described is fed using only anomaly-free patches extracted from the images. To improve network generalization, each input image is corrupted by randomly masking out rectangular regions, forcing the model to consider more of the image context when extracting features. The proposed loss function combines the pixel-wise MSE between the ground truth of the input of the autoencoder and its reconstructed output and the content loss, defined as the squared and normalized distance of the feature representations (extracted by a second network, VGG19, pre-trained on the ImageNet dataset) between the reference image (ground truth) and the reconstruction. The content loss function considers structures formed by the relations between pixel neighborhoods, while the MSE one assumes that pixels are not correlated. Finally, each pixel is classified as normal or anomalous based on its anomaly score and a threshold that maximizes the geometric mean, a combination of the true positive rate and false positive rate. In the specific work, an image with more than 10 anomalous pixels is then classified as anomalous.

Authors of [6] exploit a similar loss content function, also using a VGG19 network as a starting point, calling it perceptual loss, not during the training phase but to generate anomaly maps and detect unknown objects in pictures acquired by cameras on board of autonomous agricultural vehicles. Performances of a vanilla autoencoder, a vector quantized variational autoencoder, a denoising autoencoder, and a semisupervised autoencoder are compared. Vector quantized variational autoencoder combines VAEs with vector quantization to obtain a discrete latent representation. The encoder output is mapped to the nearest embedding vector from the shared discrete embedding space. The decoder uses the corresponding embedding vector as input. VAE tries to reconstruct an input corrupted with noise by minimizing the MSE between the reconstructed output and the non-corrupted input. A synthetic dataset is used to train the model and make it able to remove anomalies. A dataset containing abundant normal images and a relatively small percentage of anomalous samples is used for the semisupervised autoencoder. In this last case, the MSE loss function is modified with the so-called max-margin term, which keeps the reconstruction error of abnormal pixels above an experimentally fixed threshold, assuming that the ground truth mask is available. For each model, the anomaly map is generated as a relative perceptual L1 loss between the input image of the autoencoder and the reconstructed image. The threshold on the anomaly score of each pixel is fixed by maximizing the intersection over union with the ground truth. The threshold on the total anomaly score of an image, calculated as the percentage of pixels above the previous threshold value, maximizes the F1 score on the test set.

A modification of a VAE is introduced in [85] to detect underwater unknowns with dynamic under-sea backgrounds. In a VAE the encoder maps a point $\bar{x}$ in the original space to the conditional probability

$p_e(z|\bar{x})$, describing the distribution of encoded variable z in the latent space when the point $\bar{x}$ is picked up in the original space, while the decoder maps a point $\bar{z}$ sampled from $p_e(z|\bar{x})$ to the conditional probability $p_d(x|\bar{z})$, describing the distribution of decoded variable x conditioned by $\bar{z}$. During the training phase, a VAE models the encoding and decoding conditional distributions by maximizing the probability that the reconstructed output $\hat{x}$, sampled from $p_d(x|\bar{z})$, is equal to $\bar{x}$. In a vanilla autoencoder, $p(z)$ is supposed to be apriori known (generally, it is represented by the standard multivariate normal distribution). Since it is related to the conditional distributions through Bayes' theorem, it serves as a regularization term, encouraging the VAE to learn a structured and well-behaved latent space. The mentioned modification regards the latent distribution, which is not supposed to be known, but it is estimated by an autoregressive network. The latent vectors reformed by this network, composed of fully connected and masked fully connected layers, consider context information under dynamic sea background by considering the sequence dependencies in encoded space. The anomaly map, calculated at the patch level, contains two contributions: one from the patch reconstruction between the input and decode image and one from the similarity between an encoded vector and its reformed counterpart. Since the model is trained on normal samples, anomaly images containing unknown objects show a higher total anomaly score.

So far, approaches based on the definition of anomaly scores, with a threshold separating normal and anomalous samples, have been discussed. Nevertheless, clustering algorithms can also be implemented to detect anomalies directly in the latent space without considering the reconstruction error. An example of this approach is given by [73], where the goal is to detect anomalies in urban environments (like pedestrians on the road) based on stereovision acquisitions. A deep autoencoder is trained on V-disparity maps of mostly free scenes (using the cross-entropy as loss function), and flattened encoded training samples are classified in the latent space via the KNN algorithm, which is commonly used when the collected data are non-Gaussian distributed or cannot be linearly separable. Unlike clustering algorithms such as K-means, training a KNN algorithm does not require any prior assumption on the underlying data structure. During the test phase, the KNN scheme separates inliers from outliers by computing the distance of each encoded test sample from its k -nearest encoded training samples. The structure of the autoencoder makes sure that the distance between an abnormal sample (i.e., a scene with an obstacle) and its k -nearest neighbour normal training samples is larger than the distance between a normal sample (i.e., obstacle-free scene) and its k -nearest neighbour normal training samples. The threshold T on the distance is set according to the 3-sigma rule, that is:

$$T = \mu_D + 3\sigma_D, \quad (1.2)$$

In the previous equation, μ_D and σ_D are the mean and standard deviation of KNN distances under obstacle-free cases. In this case, anomaly detection is obtained by clustering the encoded representations of data samples to separate the set of normal samples, which should fall in a particular region of the latent space, from the anomalous ones, which should be characterized by out-of-region feature vectors.

Another example of the application of a clustering algorithm in the latent space is given by [33], where the authors provide a framework to process images of the melt pool acquired during Laser Powder Bed Fusion, which is an additive manufacturing process where laser power is applied to fuse the spread powder and fabricate industrial parts layer by layer. Here, an autoencoder composed of convolutional layers produces encoded representations of melt pool images, flattened and fed to an agglomerative clustering algorithm to annotate data as normal or anomalous. Then, anomalous data are discarded, and the autoencoder is retrained using normal samples only. Hotelling's and Shewhart's control charts are used to monitor the deep flattened representation vectors and the variance of residuals (the reconstruction error between flattened decoded and input images), respectively. The control limits on these two charts are set based on statistical analysis and used to decide if a real-time process is out of control (if both statistics fall inside the control limits, the process is considered in control).

1.4.4 Autoencoders combined with neural networks

Depending on the purposes, some authors combine an encoder-decoder structure with other neural networks to form a larger segmentation and anomaly detection framework. This is, for example, the case of [44], where images from metallography are inspected to segment impurities (inclusions originated from outside the sample like oxide particles added to the melt before solidification or precipitates formed from the sample itself) and grains (adjacent domains in the sample where atoms are arranged in a specific crystallographic orientation). Impurities, appearing as dots in the image, are segmented by a U-Net-inspired architecture consisting of convolutional layers organized in an encoder-decoder symmetric structure with skip connections between specular encoding and decoding layers. Skip connections allow the model to learn deep semantic information while preserving high-resolution information that might get lost during down-sampling and up-sampling. A VGG-16 neural network (composed of 13 convolutional layers followed by 3 fully connected layers) pre-trained on the ImageNet dataset is used as the encoder. Since the model is trained on small patches, a high-overlap sliding window technique is used to segment impurities on the whole image while averaging the overlapping pixels among the segmented windows to help in noise reduction. Once segmented, impurities are filled by a generative inpainting network to resemble a full metallographic scan without such defects. At this point, another U-Net model provides for the segmentation of grains. Finally, two kinds of anomaly scores are assigned

to classify segmented impurities as normal or anomalous: spatial anomaly measure, based on a version of kNN properly modified to account for how much an object is distant from its neighbours and how large it is compared to them and shape anomaly measure, based on autoencoder. In this case, however, the use of the AE is different: it is trained in a supervised manner to reconstruct not the input but a specific target image, which in the case of a normal sample is the sample itself, while in the case of an abnormal sample is a blank image. The authors find a larger reconstruction error, representing the shape anomaly score, than training the autoencoder with only normal impurities.

In [19], an autoencoder that fits in a pipeline for detecting faulty solar panels in photovoltaic power plants is presented. The dataset consists of thermal and RGB images of 6 photovoltaic power plants acquired by unmanned aerial vehicles equipped with infrared and RGB cameras under various flight conditions. As a first step, orthophotos for both kinds of images are generated and then aligned. Then, solar panels in the images are segmented by a mask region-based convolutional neural network (Mask R-CNN) architecture, and anomalies (mostly hotspots, appearing as high-intensity blobs in thermal images) are detected by an adapted version of Faster R-CNN. Thermal images undergo normalization and are augmented via geometrical and appearance-based transformation. Once an anomaly is detected, the goal of the autoencoder, which is trained to replicate the most salient features of a healthy solar panel from an RGB photo, is to detect the principal cause of the anomaly by inference on the correspondent RGB image based on a threshold on the reconstruction error. If the RGB image does not show any peculiarity, it may indicate that the cause is a deeper physical failure. The ground truth bounding box for each anomaly is manually annotated with the help of simple linear iterative clustering (SLIC), and a threshold set to 0.5 on the IoU between the predicted and true bounding box determines if the prediction is a true or false positive.

[12] resumes, in a sense, the approaches followed by the last described papers, since it compares mask R-CNN and UNet neural network (together with LinkNet and a feature pyramid network), still for anomaly cells detection in photovoltaic power plant. In particular, after the normalization of the dataset, the neural network EfficientNet is used as the backbone feature extractor, while UNet, LinkNet, and feature pyramid network as segmentation networks, which output an overall mask containing all the anomalous cells. Although UNet achieves the best results in terms of IoU, the authors underline the capability of Mask R-CNN to solve multiple tasks:

- Generation of a bounding box and its class label for each detected anomaly;
- Segmentation of the image at the instance level means that objects among the same class are clustered as different entities.

To compare the result of Mask R-CNN with other models, all the anomalous predicted cells are finally

merged into one overall mask. Moreover, the authors also compare the performance of Mask R-CNN in three different cases:

- training the network from scratch;
- pre-training the network on the Microsoft Common Objects in the Context dataset (MS-COCO), then retraining all layers;
- pre-training on the MS-COCO dataset, then retraining only the layers of the head section (the classifier section);

showing that the best results are obtained in the third case.

1.4.5 Discussion and challenges

Finally, it is worth highlighting some of the most relevant challenges posed by using autoencoders for anomaly detection in industrial scenarios. Let us note that the scope of some of these challenges is not limited to this specific topic but can also be applied to other kinds of deep learning models, even the new ones.

Let us first focus on generic issues that have to be addressed by all deep learning models. At first, there is the problem of dealing with *local minima*. Specifically, most deep learning models are trained using *parameters learning*, which involves minimizing a loss function by solving an optimization problem via a gradient descent algorithm. While effective, these algorithms are inherently affected by the possibility of falling into a local minima, providing a solution based on sub-optimal parameters. In other words, gradient descent algorithms update the network's current parameters according to the slope of the current solution of the loss function; ideally, this could be approached as a convex problem with a single global minimum, which would also define the desired set of parameters. However, in real and more complex cases, the loss function could not be globally convex and show several local minima, meaning that when the optimization reaches one of these points, it will stop upgrading the model's parameters, undermining the representational capabilities of the learning algorithm. To partially cope with this issue, several solutions were proposed. First, advanced gradient descent algorithms, such as SGD [86] or Adam [87], introduce stochasticity, which helps the optimizer escape from local minima. Furthermore, falling in a local minimum could also be avoided by adopting other techniques, such as variable learning rate and hyperparameter tuning.

Another challenge that must be considered is related to overfitting and generalization. Specifically, the optimization procedure described before "adapts" the values assumed by the model's parameters during training to minimize the loss function *mainly on training data*. Furthermore, it is not guaranteed that a neural network is capable of *generalization*, providing the same degree of accuracy when process-

ing data it was not trained on. To deal with this issue, domain experts may use different approaches, but the most common working pipeline involves the following steps:

- *Selecting a proper dataset*, that is, a dataset able to provide as many perspectives as possible on the phenomena under investigation.
- *Splitting the dataset*, using a certain percentage (usually 70 – 80%) of the data to train the model while the rest to validate and test its performance.

In the case of anomaly detection, the main problem to be solved could be represented as implementing a one-class model based on the observations of a certain phenomenon in *good operating conditions* so that the model itself could identify anomalous situations, capturing a comprehensive dataset becomes a crucial step. The second step is dataset splitting, which involves a randomized selection of the data samples to provide to the model during training, mainly to avoid biases related to subsets of data acquired under common and specific conditions. To further stress this randomization, k -fold cross-validation is often also used: data are first split into k folds, on which the model is trained separately, and, finally, results are merged via a specific strategy.

As for the issues specific to the application of autoencoders for anomaly detection, one of the most relevant is the lack of generalization capability of models trained on specific aspects of the application scenario. For example, the model trained in [11] was trained from data provided by a single turbine in a farm and validated accordingly. As different turbines may present several differences, for example, in weight, span, underlying technologies, or even orientation, the generalization capabilities of the trained model were not guaranteed. This was also the case for several other proposals, such as the anomaly detector on PCBs proposed in [56] or the work in [6].

Another challenge arisen from the review was the need to establish a baseline for improvements achievable by process optimization and hyperparameters tuning, such as [33]. Specifically, by gathering data of proper quality and using algorithms which take into account optimization procedures that involve the automatic tuning of parameters (e.g. the number of layers in the autoencoder, the number of neurons per layer, the size of the latent space, the optimization function, etc...), the reconstruction capabilities anomaly detection performance can vastly improve.

Using autoencoders may also lead to several improvements over classic ML techniques. This may be quantified in the first place using common metrics such as accuracy, precision, and recall: most of the works analyzed in this review demonstrate higher performance when autoencoders and DL methods are compared with classic ML approaches. Moreover, in several cases, the use of DL also allows for improved processing speed, such as [14] and [9] demonstrated when GRUs were compared with LSTMs and classic time series modelling approaches, therefore allowing for real-time implementation of the

Table 1.6: Results achieved by autoencoders for anomaly detection

Reference	Application area	Metric	Value
[22]	Energy forecasting	RMSE	27.43
[27]	Smart manufacturing	MSE	0.0044
[53]	Environmental	RMSE	2.980
[6]	Agriculture	F_1	85.91%
[7]	Agriculture	F_1	99.42%
[13]	Energy forecasting	F_1	93.50%
[17]	Energy forecasting	F_1	91.90%
[29]	Smart manufacturing	F_1	82.30%
[37]	Smart manufacturing	F_1	95.70%
[38]	Smart manufacturing	A	87.70%
[40]	Smart manufacturing	F_1	84.46%
[42]	Smart manufacturing	AUC	89.00%
[45]	Smart manufacturing	F_1	99.80%
[46]	Smart manufacturing	AUC	96.10%
[49]	Smart manufacturing	F_1	90.3%
[56]	Smart manufacturing	F_1	81.10%
[57]	SHM	F_1	65.87%
[58]	SHM	F_1	89.58%
[63]	SHM	AUC	99.93%
[70]	Environmental	F_1	81.35%
[73]	Vehicles	AUC	91.00%
[76]	Vehicles	R^2	$\geq 99.00\%$
[77]	Environmental	F_1	$\geq 84.00\%$

anomaly detector. Finally, the representational capabilities of autoencoders can be further extended, hinting not only at the rise of an anomaly but also at the nature behind the fault that likely caused it, effectively providing a way for domain experts to properly address potentially critical events, such as theorized by the authors in [11].

A summary of the results of the works that used autoencoders for anomaly detection is provided in Table 1.6. Due to the wide variety of results in application areas, evaluation metrics, considered datasets, and experimental procedures, it is important to underline that Table 1.6 only provides a *synthetic* summary, highlighting only the most relevant metric computed by the authors with regards to the scope of this survey.

1.5 Summary

This chapter presented a review of deep learning-based anomaly detection strategies that are applied to industrial scenarios. The literature strategy search and the subsequent paper analysis (designed over the PRISMA protocol) have been guided by three different factors under analysis: i) the application area of the papers – i.e., their operating context; ii) the sensing equipment used – i.e., the datasets and

the strategies used to capture data from the field in real scenarios; iii) the algorithms used to process the high amounts of data that today are straightforwardly available at relatively low costs. In brief, the three factors are related to the applications, hardware, and software used in real scenarios in the last years, suggesting the three research questions formulated in this review paper. From the literature analysis, it is possible to critically evaluate the research trends about applied anomaly detection, which result in best practices to be also further investigated as soon as new technologies and instruments become available in the near future. Even if all the reviewed papers based their analysis and consequent results on the effective use of machine/deep learning techniques, they are focused on solving a problem usually formulated on specific data, making a high customization of the proposed algorithms. Given the fact that more computational resources will likely be available to the large public in the future, the capability of exploiting innovative approaches using artificial intelligence should be encouraged, for example, investigating the cross-domain application of anomaly detection models, acting as the commonly used pre-training of large convolutional neural network models in computer vision.

Chapter 2

Assessing Welding Anomalies via Non-Destructive Tests

Detecting defects that may arise in weldings used in either critical or non-critical industrial applications is an extensive area of active research. The integrity of welded joints plays a critical role in ensuring the safety and durability of manufactured structures in industries ranging from aerospace to civil engineering. Given the crucial role of welding, the reliable detection of defects such as cracks, porosities, or lack of fusion is a central concern in quality assurance. As such, non-destructive tests are often required, mainly to preserve the integrity of normal samples while discarding defective ones. Traditional Non-Destructive Testing (NDT) techniques, including radiographic testing, ultrasonic testing, and magnetic particle inspection, have long been employed to identify anomalies in welds. However, these techniques often demand specialized equipment, significant human expertise, and can be costly or time-consuming. Recent advances in deep learning, particularly in anomaly detection, have opened new opportunities for automating weld inspection in a robust and cost-efficient manner. Profilometers provide a valuable modality for this task. By generating high-resolution surface profiles of weld seams, they offer a direct and non-invasive means to assess structural quality. Still, as these tasks require relevant efforts and could be biased by subjectivity and inexperience, the development of automated, objective tools that provide early warnings about the occurrence of one or more anomalies has become essential. To deal with these issues, this work proposes a framework for detecting anomalies in linear aluminum welding. Specifically, the framework starts by acquiring the three-dimensional representation of the welding using a 3D laser profiler. Afterward, a one-class supervised approach is followed by training a deep autoencoder on non-defective data samples, allowing the model to learn a mapping function that successfully reconstructs non-defective weldings with a small reconstruction error while providing a large error for abnormal data samples. Hence, by applying a proper threshold on the re-

construction error, which can be estimated via statistical analysis, the framework can provide real-time early warnings concerning surface defects on the welding. Different structures of autoencoder have been tested and found to reach an F1 score of over 90%.

2.1 Introduction to the use of AI in welding process

In the last years, technological innovations are being constantly improved and brought in industrial processes, leading to an increase in the production quantity and a subsequent need for cutting-edge methodologies for effective quality control operations. Different application scenarios and contexts in industrial applications (e.g., aerospace, railway, automotive, logistics) can define both critical and non-critical situations. A specific and quite common task in a wide variety of such applications is quality control aimed at the detection of defects of welded joints and products, in terms of anomalies that may occur during the welding process [88].

Welding monitoring is a task that can be effectively performed using visual inspection methods, either manually or semi-automatically, relying on computer vision, machine learning, and artificial intelligence techniques. In complex and fully automated systems, the welding is performed automatically by robotic arms, so a properly designed inspection system can be coupled to the welding device to perform the quality assessment.

The use of an AI-based methodology for this kind of quality assessment can effectively help domain experts because the AI model can be trained in order to reduce and avoid the bias that could be introduced by human operators – either experienced or not – especially when the task requires the analysis and assessment of a high number of welds for a high amount of time during the working day, leading to the operator fatigue. For example, in [89], machine vision is used to detect anomalies over a dataset composed of 1.000 images of PCBs. Specifically, the authors first perform a pre-processing step, converting the images into the lab color space, and calculating the saliency of the representation. Afterward, the type of welding spot anomaly is determined considering the Intersection over Union metric. The authors measure their results in terms of F1 score, achieving a 95.56% F1 score on resistance anomaly detection, and a 96.79% F1 score on welding spot anomaly detection. However, the inference time required is quite high in both cases, suggesting the requirements for optimization before considering real-time implementation. In [90], the authors acquired a dataset of defects concerning steel weldings using a three-dimensional reconstruction gathered from a 3D laser line system. The dataset contained 2.437 defective samples representing slag inclusions, cracks, floating holes, and unfused parts, and 1.712 non-defective samples. The authors trained VGG16 on the dataset, achieving an overall accuracy of 97.1%. In [91], a dataset composed of 7.600 images representing weldings in fuel

injectors has been analyzed using DenseNet-121. Due to the highly imbalanced data, the authors used specific data balancing techniques, along with pre-processing and hyperparameters tuning, to achieve a maximum accuracy of 96.30% in damage detection. Finally, in [92], the authors use a specifically tailored convolutional neural network to obtain a 0.63% error rate on a dataset containing 14.400 samples of surface defects gathered from a hot-rolled steel sheet production line.

This chapter investigates the use of deep autoencoders [93] for assessing welding anomalies, feeding them with 3D data acquired from the field. This type of neural network has been already used in anomaly detection tasks in related fields, such as in [94] and [95]. The problem is formulated as an anomaly detection due to the large availability of non-defected samples or equivalently good quality welds, that are employed with the aim of establishing a strong relationship between the 3D model of a single welding line with its geometrical features and external appearance in the case of a good welded surface. This way, we evaluate the reconstruction error in both cases of non-defected welds as well as defected ones, in order to define a proper way to discriminate between them without explicitly training the AI model using defective samples.

The rest of the chapter is organized as follows. In section 2.2 we recap the details about the data gathering setup, the dataset acquired and the AI models used to perform the experiments. In section 2.3 the results achieved on our dataset are reported, while section 2.5 concludes the chapter and gives a perspective about possible future works.

2.2 Experimental setup for acquiring welds' profiles

The experimental setup, as described in [88], consists of an aluminum plate crossed by a linear welding and a 3D laser profiler moving on a linear slide parallel to the welding. 3.187 one-dimensional series have been acquired from different plates by moving the laser along the slide. Each series corresponds to a fixed position of the laser and represents the thickness of the aluminum plate along the direction orthogonal to the linear welding. An illustrative scheme is reported in Fig. 2.1.

Among the 3.187 series, 1.509 are classified as *anomalous* (i.e., defective), and the others as *normals* (i.e., non-defective). Visually, anomalous series are characterized by a large discrepancy between the two sides of the welding. Series or pieces of series corresponding to the boundaries of the plate have been discarded. The final dataset is perfectly balanced, consisting of an equal number (1.300) of anomalous and normal series, made up of $d_i = 985$ points (the subscript i stands for "input") and smoothed with a Savitzky-Golay [96] filter. This kind of filter reduces the noise in the acquisitions without distorting the signal tendency. This is achieved by fitting successive sub-sets of adjacent data points with a low-degree polynomial by the method of linear least squares. The subsets size s and the

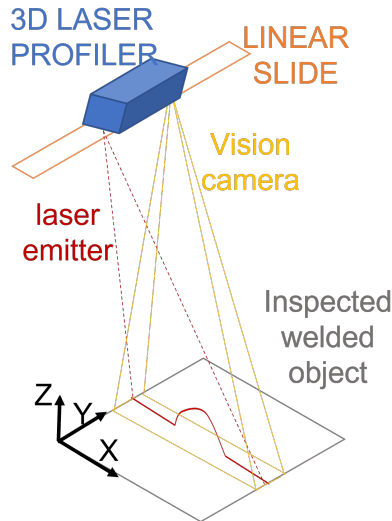


Figure 2.1: Experimental setup: the 3D laser profiler moves along a linear slide parallel to the welding (y - axis) and measures the thickness (z - axis) of the aluminium plate in the allowed range, denoted by a red line broadening on the x - axis. The bump on the red line points out the presence of the welding.

degree p of the polynomial have been fixed to $s = 7$ and $p = 2$, respectively.

Then, for each series, the initial value of a series has been subtracted from the other values. This trick allows one to get rid of the bias caused by the inclination of the plate along the y - axis. At this point, three types of autoencoders, dense-based, convolutional-based, and LSTM-based, have been built using Keras [97]. All the models have been trained with normal series and tested on both anomalous and normal series, following the approach of supervised learning in one-class classification problems. The number of epochs and the batch size have been fixed to 10 and 32, respectively; mean absolute error and Adam [98] have been chosen as the loss function and optimizer, respectively. All data have been scaled between 0 and 1.

A k -fold cross-validation procedure, with $k = 10$, has been adopted to mitigate the fluctuations of the results so that the test set includes all anomalous series and 130 normal series. Train data have been also shuffled to remove any bias due to the sequential acquisitions acquired when the laser moves along the welding. Secondly, the two labels have been swapped and all the steps have been repeated. For each type of autoencoder, different models have been implemented by varying the number of layers and the dimension d_{lat} of the latent-feature space. All the considered models had an equal number of encoding and decoding layers.

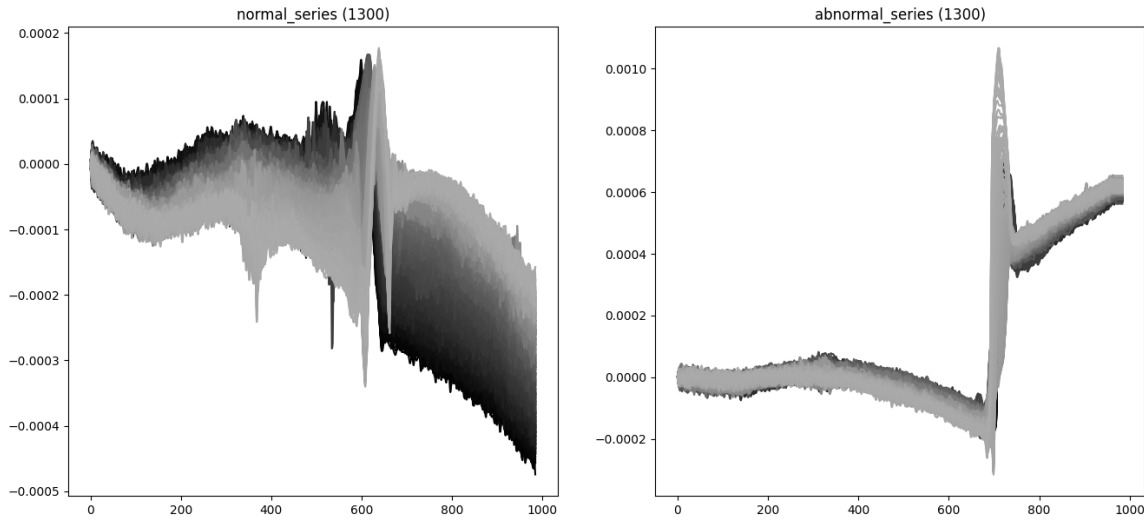


Figure 2.2: Normals and anomalous normalized series plotted in grey scale from the first one along the y -axis, in grey, to the last one, in black. x -axis and y -axis represent position indexes and thickness-related measured values.

2.2.1 Dense autoencoder

This is a fully connected neural network, also called multilayer perceptron. A dense layer connects each neuron (node) in its input layer to every neuron in its output layer, allowing the neural network to model complex relationships and patterns in data. Each connection between neurons has an associated weight, which the network learns during training. These weights determine the strength of the connection and are adjusted through backpropagation to optimize the network's performance on a specific task. In this case, train data are assembled in a matrix M with dimension $[n_{samples}, d_i]$, where $n_{samples}$ is the number of samples and each row of the matrix corresponds to a series. The first layer consists of $u_1 = d_i$ units and acts on the train data matrix returning a matrix with dimensions $[n_{samples}, o_1]$, where o_1 equals the number of units u_2 in the second layer (o and u stand for "output" and "units"). In general, a given layer l transforms its row-input with dimension u_l in a row-output with dimension $o_l = u_{l+1}$ and the number n_l^d of parameters, the weights of the transformation, introduced in the model by the layer is:

$$n_{l,dense} = (u_l + 1) \cdot o_l. \quad (2.1)$$

The "1" in the round brackets comes from the bias unit added in each layer. 21 models of dense autoencoder have been tested: 8 with $d_{lat} = 4$, 7 with $d_{lat} = 8$, and 6 with $d_{lat} = 16$ (l stands for "latent"). For each of these values, the simplest considered autoencoder consists of one encoding layer, with

Table 2.1: Results of dense autoencoders. m is the number of dense layers in both encoder and decoder, d_{lat} the dimension of the latent feature space, represented by the number of units of the first decoding layer.

Dense autoencoder					normal $\leftrightarrow$ anomalous			
$d_{lat} = 4$	a	p	r	F_1	a	p	r	F_1
m = 1	98.20	100.00	80.23	89.03	98.17	100.00	79.85	88.79
m = 2	98.36	100.00	82.00	90.11	98.40	100.00	82.38	90.34
m = 3	98.84	100.00	87.23	93.18	99.12	100.00	90.31	94.91
m = 4	98.71	100.00	85.77	92.34	98.80	100.00	86.85	92.96
m = 5	98.72	100.00	85.92	92.43	98.68	100.00	85.46	92.16
m = 6	98.89	100.00	87.77	93.49	98.52	100.00	83.69	91.12
m = 7	98.41	100.00	82.54	90.43	98.32	100.00	81.54	89.83
m = 8	98.62	100.00	84.85	91.80	98.35	100.00	81.85	90.02
$d_{lat} = 8$								
m = 1	98.05	100.00	78.54	87.98	98.34	100.00	81.69	89.92
m = 2	98.87	100.00	87.54	93.36	98.94	100.00	88.31	93.79
m = 3	98.97	100.00	88.62	93.96	99.02	100.00	89.23	94.31
m = 4	98.38	100.00	82.15	90.20	98.70	100.00	85.69	92.29
m = 5	98.80	100.00	86.85	92.96	98.73	100.00	86.00	92.47
m = 6	98.67	100.00	85.38	92.12	98.30	100.00	81.31	89.69
m = 7	98.57	100.00	84.23	91.44	98.43	100.00	82.69	90.53
$d_{lat} = 16$								
m = 1	98.15	100.00	79.69	88.70	98.49	100.00	83.38	90.94
m = 2	98.99	100.00	88.85	94.09	98.78	100.00	86.62	92.83
m = 3	98.94	100.00	88.31	93.79	98.69	100.00	85.54	92.21
m = 4	98.37	100.00	82.08	90.16	98.70	100.00	85.69	92.29
m = 5	98.80	100.00	86.85	92.96	98.88	100.00	87.69	93.44
m = 6	98.57	100.00	84.23	91.44	98.81	100.00	86.92	93.00

$$u_{e,1} = d_i, \quad o_{e,1} = d_{lat}, \quad (2.2)$$

and one decoding layer, with

$$u_{d,1} = d_{lat}, \quad o_{d,1} = d_i. \quad (2.3)$$

Successive models are iteratively built adding one layer per time, in encoding and decoding, in such a way that the autoencoder with m encoding (and decoding) layers satisfies the following constraints:

$$o_{e,1} = 2^{m-1} d_{lat} < d_i, \quad o_l = 2o_{l+1}. \quad (2.4)$$

Following this strategy, the number of units in each layer is a power of 2, and the maximum value of $o_{e,1}$ is 512 ($512 < d_i < 1024$). All layers are activated by the rectified linear unit [99] function, apart from the last one which is activated by the sigmoid activation function to deal with the binary

Table 2.2: Results of convolutional autoencoders. m is the number of one-dimensional convolutional layers in both encoder and decoder, d_{lat} the dimension of the latent feature space, represented by the number of filters of the last encoding layer.

Convolutional autoencoder					normal $\leftrightarrow$ anomalous			
$m = 1$	a	p	r	F_1	a	p	r	F_1
$d_{lat} = 2$	98.33	100.00	81.62	89.88	98.42	100.00	82.62	90.48
$d_{lat} = 4$	98.60	100.00	84.62	91.67	98.56	100.00	84.15	91.40
$d_{lat} = 8$	98.83	100.00	87.15	93.14	98.88	100.00	87.69	93.44
$d_{lat} = 16$	98.76	100.00	86.38	92.70	98.94	100.00	88.38	93.83
$m = 2$								
$d_{lat} = 2$	98.65	100.00	85.15	91.98	98.61	100.00	84.69	91.71
$d_{lat} = 4$	98.86	100.00	87.46	93.31	98.81	100.00	86.92	93.00
$d_{lat} = 8$	99.12	100.00	90.31	94.91	98.89	100.00	87.77	93.49
$d_{lat} = 16$	98.79	100.00	86.69	92.87	98.35	100.00	81.85	90.02
$m = 3$								
$d_{lat} = 2$	98.84	100.00	87.23	93.18	98.85	100.00	87.31	93.22
$d_{lat} = 4$	99.12	100.00	90.31	94.91	98.83	100.00	87.15	93.14
$d_{lat} = 8$	98.87	100.00	87.62	93.40	98.77	100.00	86.46	92.74
$d_{lat} = 16$	98.87	100.00	87.54	93.36	98.45	100.00	83.00	90.71

classification problem.

2.2.2 Convolutional autoencoder

This type of autoencoder [100] is based on a stack of one-dimensional convolutional layers. A convolutional layer is the key building block in CNNs, a type of deep learning architecture primarily designed for processing grid-like data. As information flows through successive convolutional layers, the network learns to capture increasingly abstract and hierarchical features. When the input is an image, for example, lower layers might detect basic patterns like edges and corners, while higher layers learn to recognize complex shapes, textures, and object parts. Generally speaking, convolutional layers are used to apply a convolution operation (either 1D, 2D, 3D or multidimensional) to the input data. This operation involves sliding a small filter (also known as a kernel) over the input data to compute a weighted sum of neighboring pixel values. The filter's weights are learnable parameters, and they determine the features that the layer extracts from the input. By focusing on local receptive fields within the input data, allowing them to capture patterns that are translationally invariant. This means they can recognize the same patterns regardless of their location in the input. In the general case multiple filters are applied simultaneously such as a filter bank, each of which is responsible for detecting a different feature or pattern. These filters operate in parallel to produce separate feature maps. Padding and stride are hyperparameters that affect the size of the output feature maps. Padding involves adding extra border pixels to the input data to control the spatial dimensions of the output. Stride determines the

Table 2.3: Results of LSTM autoencoders. m is the number of LSTM layers in both encoder and decoder, d_{lat} the dimension of the latent feature space, represented by the number of units of the last encoding layer.

LSTM autoencoder					normal $\leftrightarrow$ anomalous			
$d_{lat} = 4$	a	p	r	F_1	a	p	r	F_1
m = 1	99.02	100.00	89.23	94.31	98.79	100.00	86.69	92.87
m = 2	98.30	100.00	81.31	89.69	98.43	100.00	82.77	90.57
m = 3	98.10	100.00	79.08	88.32	98.10	100.00	79.08	88.32
m = 4	98.17	100.00	79.85	88.79	98.12	100.00	79.31	88.46
m = 5	98.24	100.00	80.62	89.27	98.19	100.00	80.08	88.94
m = 6	98.08	100.00	78.85	88.17	98.15	100.00	79.69	88.70
$d_{lat} = 8$								
m = 1	98.35	100.00	81.85	90.02	98.78	100.00	86.54	92.78
m = 2	98.25	100.00	80.77	89.36	98.36	100.00	81.92	90.06
m = 3	98.12	100.00	79.31	88.46	98.06	100.00	78.62	88.03
m = 4	98.20	100.00	80.15	88.98	98.13	100.00	79.38	88.51
m = 5	98.17	100.00	79.85	88.79	98.24	100.00	80.62	89.27
$d_{lat} = 16$								
m = 1	98.36	100.00	82.00	90.11	98.57	100.00	84.31	91.49
m = 2	98.27	100.00	81.00	89.50	98.12	100.00	79.31	88.46
m = 3	98.22	100.00	80.38	89.13	98.18	100.00	80.00	88.89
m = 4	98.08	100.00	78.85	88.17	98.20	100.00	80.15	88.98

step size by which the filter is moved across the input. A larger stride results in smaller output feature maps. Optionally, a pooling step can be inserted to reduce the spatial dimensions of the feature maps, helping to improve computational complexity and enhance translation invariance. In this work The kernel size of each layer has been set to s , with a stride length of 1. The matching between the input series length d_i and the corresponding output has been obtained by padding with zeros the output of each layer. Cropping the first and last ten values of the output ensures the results not to be affected by the padding. The input is expected to be a tensor with dimension $[n_{samples}, d_i, 1]^1$. If f_l is the number of filters of a given layer l , the output will have dimension $[n_{samples}, d_i, f_l]$. The number of parameters $n_{l,conv}$ (the weights of the convolution) introduced by the layer l is:

$$n_{l,conv} = (s \cdot f_{l-1} + 1) \cdot f_l, \quad (2.5)$$

with $f_0 = 1$. Three categories of convolutional autoencoders have been built up by fixing m of encoding/decoding layers to 1, 2, and 3. For each category four different models have been implemented, differing for the dimension d_{lat} of the latent feature space, given by the number of filters of the last encoding layer ($d_{lat} \in \{2, 4, 8, 16\}$). External layers have a double number of filters with respect to

¹If the one-dimensional series taken into consideration had mapped a numeric value into a vector of size m , the last dimension of the input tensor would have been m .

their adjacent internal layers and the final output layer has only one filter. The activation functions are the same as the dense autoencoder.

2.2.3 LSTM autoencoder

The last type of autoencoder that is going to be discussed falls into the class of recurrent neural network and has been first proposed in [101]. A recurrent layer is the fundamental building block of RNNs, particularly suited for capturing dependencies and patterns in data sequences. Unlike feedforward neural networks, which have no feedback loops, RNNs have recurrent connections that allow information to flow from one time step to the next within the sequence. Each time step in a sequence corresponds to one input element and one output element, and the recurrent connections enable the network to maintain a hidden state that carries information from previous time steps. The hidden state acts as the memory of the network, capturing information about previous elements in the sequence. At each time step, the hidden state is updated based on the current input, the previous hidden state, and a set of learnable parameters (weights and biases). Traditional RNNs are prone to the vanishing gradient problem, which can make it challenging for them to capture long-range dependencies in sequences. This is because the gradients of the loss function with respect to the parameters tend to become very small when backpropagated through many time steps. To mitigate this issue, more advanced RNN variants like Long Short-Term Memory LSTM and GRU have been developed. These architectures use gating mechanisms to control the flow of information through the network and address the vanishing gradient problem. LSTM layer consists of a sequence of building blocks, called memory cells, one for each element of the series. The n -th memory cell of a given layer l receives the following inputs:

- the n -th element of the series;
- two vectors, called "cell state vector" C_{n-1} and "hidden state vector" h_{n-1} , with dimension u_l , where u_l identifies the number of "units" of the layer (to not to be confused with the "units" of a dense layer).

Two new cell (hidden) state vectors $C_n(h_n)$ are returned, which will be the input of the next memory cell. If the option `return_sequences` is set to `True` then each memory cell will send the hidden state vector also to the n -th memory cell of the next layer, if any, otherwise, the last hidden state vector will be the output of the full layer. Then, given an initial input with shape $[n_{samples}, d_i, 1]$, the output of the first LSTM layer will have the shape:

- $[n_{samples}, d_i, u_1]$ if `return_sequences = True`,
- $[n_{samples}, u_1]$, if `return_sequences = False`.

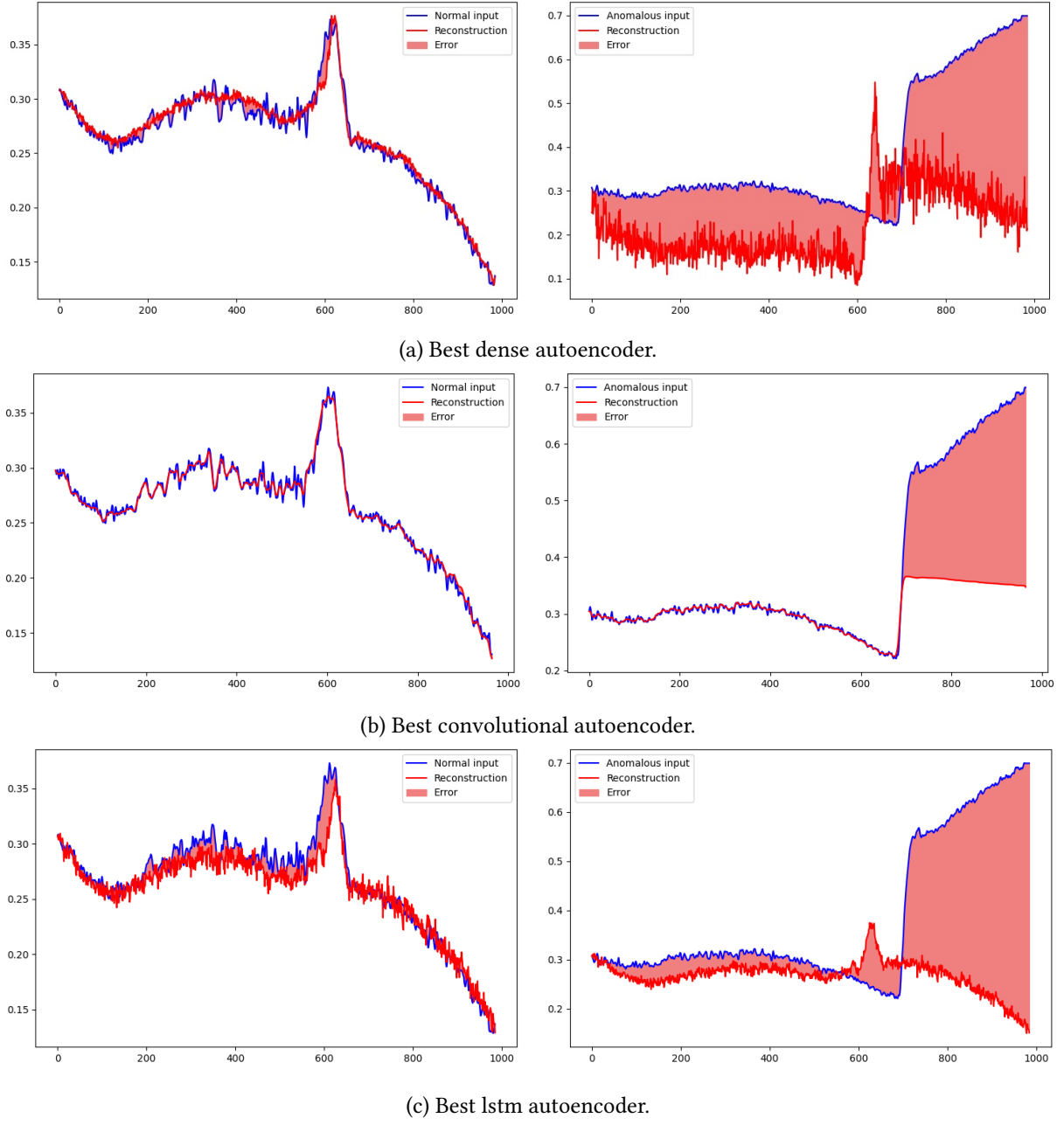


Figure 2.3: On the left, a given normal series (blue) from the test set and the reconstructed one (red); on the right, an example of anomalous series. x-axis and y-axis represent position indexes and normalized thickness, respectively.

In the second case, if one wants a second layer to follow the first, then the output must be put in a tensor form with shape $[n_{samples}, u_1, 1]$. The complete internal structure of the memory cell will not be deepened. The only relevant information to calculate the number of parameters is that the memory cell provides four dense layers, which receive as input the vector of size $u_l + 1$ obtained concatenating the correspondent element of the series and the hidden state vector and return vectors of size u_l . Remembering Eq. 2.1 one can write the number parameters of the l -th layer as:

$$n_{l,lstm} = 4(u_l + 1 + 1) \cdot u_l. \quad (2.6)$$

The latent space dimension d_{lat} is now represented by the number of units of the last layer in the encoder. Following the strategy of doubling the number of units of more and more outer layers, different models have been built. GPU memory restricted the maximum number m of encoding (decoding) layers to 6, 5, and 4 when d_{lat} has been equal to 4, 8, and 16, respectively. The last LSTM decoding layer must have always a number of units equal to d_i , to match the length of the input series. The hyperbolic tangent has been chosen as the activation function for all the layers. This choice meets the requirements of The NVIDIA CUDA Deep Neural Network library (cuDNN) and allows the deep neural network to run on GPU since the computational time on the CPU has been at least four times slower.

2.3 Evaluating metrics in one-class supervised approach

In Fig. 2.2 all the normals and anomalous series are plotted. By simply looking at the series one can see that the welding is identified by a peak: in anomalous series, the planes to the right and to the left of the welding are staggered. The goal of the autoencoder is to recognize this anomaly and consequently classify the series. The quality of the autoencoders has been estimated through the evaluation of the following metrics:

$$a = \frac{TP + TN}{TP + TN + FP + FN} \quad (\text{accuracy}), \quad (2.7a)$$

$$p = \frac{TP}{TP + FP} \quad (\text{precision}), \quad (2.7b)$$

$$r = \frac{TP}{TP + FN} \quad (\text{recall}), \quad (2.7c)$$

$$F_1 = 2 \frac{p \cdot r}{p + r} \quad (F_1 \text{ score}), \quad (2.7d)$$

where T and F stand for "True" and "False", P and N for "Positives" and "Negatives". The threshold for the classification of a positive or a negative test series has been chosen as one standard deviation of reconstruction errors of the train data series. Table 2.1, 2.2, 2.3 summarize the results, with a yellow marker denoting the autoencoder model with the best values of the metrics. All models achieve a precision of 100%, that is no false positives arise. This is due to the fact that the training data samples belong to one class, classified as positive. Then, all models are extremely able not to confuse negative series with positive series. Lower recall values are due to pieces of the welding correspondent to series labeled as positives but classified as negatives, probably because of shapes very close to the negative

ones. Summarizing, one can conclude that if a series does not belong to the same class as training samples then it certainly will not be misclassified. Another observation is that although some models reach better results, the difference with the other models is not so marked. A plausible explanation is that the structure of the analyzed dataset is not so rich to be sensitive to the change in the number of layers or the latent space dimension. In Fig. 2.3 an example of a series reconstructed by the best autoencoder is shown. The convolutional autoencoder reconstructs the input in the smoothest shape. On the contrary, the LSTM autoencoder tries to reconstruct the series with its original noise. It is evident that all autoencoders fail to well reconstruct an anomalous series since it has been not in the training dataset. This behavior allows us to fix a threshold on the reconstruction error above which most of the negative series fall and allows us to distinguish between normal and anomalous series, as announced at the beginning of this work.

2.4 Application to robotics

The integration of artificial intelligence into robotic systems presents a compelling avenue for future development. In [102] a laser profiler is integrated into a robot-based system that is used to fabricate metallic parts with powders. A machine learning model that combines clustering and multiclass classification is embedded in the in-situ point cloud processing node to identify surface defects. In [103], a single-dimensional laser profiler sensor is integrated with a 6-DoF robotic arm, enabling both systems to function independently and facilitating potential reconfiguration of the motion system. These works show how in modern manufacturing environments, robotic arms equipped with profilometers or other NDT sensors can acquire weld profiles autonomously during or immediately after the welding process. Within the framework of sensor systems integrated into robotic arms, an activity developed at Comau, a leader company in the field of industrial automation, and CNR, can be cited. AI-powered methodologies have been embedded in an innovative model-driven intuitive programming framework, applied to a self-adaptive system that autonomously inspects helicopter blades. By autonomously generating models of the surrounding environment, the proposed system localizes its target, detects any obstacle, calculates cycle time optimization and collision-free path planning and makes decisions that intuitively refine its movements and actions [104].

2.5 Summary

In this chapter, three different types of autoencoders have been compared for anomaly detection in a dataset containing lines gathered from a three-dimensional representation of aluminum welds. Each

one of the proposed architectures has been optimized using hyperparameters optimization, and tested against a perfectly balanced version of the original dataset. Results show the effectiveness of the proposed method, as all the optimized architectures show high accuracy, with the best results achieved by the simplest LSTM autoencoder, which has been able to achieve 99.02% accuracy in normal samples detection, and 98.79% accuracy while detecting defective samples.

The challenge lied in distinguishing between normal variations in weld geometry and anomalous profiles that indicated potential defects. Deep learning models, and autoencoders in particular, turned out to be well-suited for this type of analysis due to their ability to learn compact feature representations and reconstruct complex input signals. In the present approach, an autoencoder was trained exclusively on non-anomalous weld profiles acquired through profilometry. The underlying assumption was that the network would have learn to efficiently encode and reconstruct the normal class of data, capturing its underlying statistical distribution. When presented with an anomalous weld profile, however, the reconstruction error was significantly higher due to the network's inability to generalize features outside its training distribution. Thus, the reconstruction error served as a natural anomaly score, allowing for the automated classification of anomalous welds. This reconstruction-error-based approach differs from other strategies that also employ autoencoders for anomaly detection.

A widely explored alternative involves leveraging the latent space of the encoder. In such methods, latent vectors corresponding to training samples are clustered, and anomalies are detected by measuring the distance of a new latent representation from the learned cluster centers. While latent-space clustering methods can capture structural information beyond reconstruction accuracy, they often require additional steps such as clustering algorithms and thresholding procedures, as it will be discussed in the next chapter.

By contrast, reconstruction-based anomaly detection offers a more direct and computationally efficient pipeline: anomalies are inferred directly from the reconstruction error, without the need for external clustering. This makes the reconstruction-error-based approach particularly appealing for real-time applications where speed is critical. Embedding an autoencoder trained on normal weld profiles within robotic systems would enable online detection of anomalies without interrupting production. This synergy between deep learning and robotics aligns with the broader paradigm of Industry 5.0, in which sensor-rich robotic systems are expected not only to execute tasks but also to self-monitor and ensure quality compliance in real time. Furthermore, the scalability of autoencoder-based methods means that retraining can be performed as new data are collected, allowing the system to adapt to variations in welding processes or materials. By extending this approach to robotic inspection systems, manufacturers could achieve fully automated, adaptive quality control pipelines, marking a significant step toward autonomous, intelligent production environments.

Chapter 3

Testing of Aircraft Components using Laser Ultrasonic Data

Composite structures are commonly used in complex applications such as automotive and aerospace due to their high strength-to-weight ratio. Although strictly supervised and inspected, they are often subject to dynamic events during their useful life that can cause invisible failures that extend and severely compromise their performance over time. Detecting these defects preventively and repairing them could avoid dramatic accidents. In this chapter, a deep learning-based method for the detection of defects in Carbon Fibre-Reinforced Plastic (CFRP), based on Laser Ultrasonic Testing (LUT), is presented. LUT is a promising NDT method for detecting inner defects in a non-contact way, as it does not require liquid coupling media. A composite laminate specimen containing six programmed defects as a test sample will be investigated. It will be shown that training deep learning-based models as autoencoders makes it possible to extract features that can be used to discern defective areas from non-defective ones. The results demonstrate high detection accuracies (above 90% balanced accuracy and 75% F1-score), indicating a promising and effective approach to NDT on composite materials.

3.1 Overview on composite materials and ultrasonic testing

3.1.1 Ultrasonic Testing

LUT falls into the broader category of NDT techniques based on Ultrasonic Testing (UT), that uses high-frequency sound waves to detect defects, measure thickness, or characterize materials without causing any damage. The core principle involves the transmission of high-frequency sound waves into a material, where the waves interact with internal structures, reflecting back information about discontinuities in the sample. These discontinuities can be grouped into three types:

- cracks: these are fractures or separations within the material, often caused by stress, fatigue, or thermal cycling. Cracks can be surface-level or deep inside the material. Ultrasonic waves reflect differently from these discontinuities, making them detectable;
- inclusions: these refer to impurities or foreign materials trapped within the material during the manufacturing process. These can cause variations in the propagation of ultrasonic waves, showing up as changes in the signal, which helps detect their presence;
- delaminations: these are separations between layers within laminated or composite materials, often found in aerospace or automotive components. Voids are empty spaces within the material caused by improper manufacturing or stress. Both anomalies disrupt the uniform transmission of ultrasonic waves, which are reflected at these interfaces, indicating their presence.

The inspection techniques which exploit ultrasound signals can be contact-based or contact-free. Contact-based ultrasound inspection involves direct interaction between the surface of the material being tested and the transducer, which generates the high-frequency sound waves. The need for physical contact and the eventual application of coupling agents between the transducer and the surface of the material can be impractical for samples with irregular surfaces or components in difficult-to-reach locations. In contrast, non-contact ultrasonic techniques overcome some of these limitations by eliminating the need for direct contact between the transducer and the material. Ultrasonic immersion testing is a common example, where the component is submerged in a liquid, usually water, which acts as the coupling medium. The transducer operates within the liquid, transmitting sound waves through the medium into the material. This method allows for uniform transmission across complex geometries and is particularly suitable for automated inspections in industrial settings.

UT is a widely used method for quality control and inspection of materials in various industries. It is also used for preventive maintenance purposes, meaning it does not cause any damage to the inspected material. However, from the literature analysis, it was clear that, despite the effectiveness of UT in subsurface defect detection, a-priori knowledge of characteristic parameters (such as fibre direction or material thickness) could be necessary to perform effective quality control [105]. Moreover, extracting suitable features from raw data for this purpose can be challenging too, as pointed out in [106], where the authors used feature engineering to optimally feed supervised ensemble classifiers for non-destructive testing by ultrasounds. Over the years, several researchers have processed ultrasonic data differently, trying to find the best trade-off between traditional approaches and innovations. For example, integrating different supervised machine learning models fed by 2D images, obtained properly rearranging A-scan wavelet coefficients, was first introduced in [107]. Also, in [108], an attempt at supervised binary classification of defective and non-defective areas of a specimen based on A-scan data from 3D braided composites was investigated, enhancing classification accuracy by implementing fil-

tering and denoising of A-scan data. Moreover, a comparison study that evaluated the effect of training traditional CNNs using synthetic ultrasonic data with noise was presented in [109], showing that synthetic data could have the potential of being introduced in deep learning pipelines because significant improvements using simulated datasets can be effectively made. However, the authors point out that simulation drawbacks were related to replicating the same noise distributions of experimental data, which could result in overfitting and can be particularly challenging when models with limited complexity were used. Finally, in recent works such as [110] the authors focus on the processing of B-scan data obtained via an innovative gating method based on unsupervised clustering Density-Based Spatial Clustering of Applications with Noise (DBSCAN) and then using such B-scan data as convolutional autoencoder input to achieve satisfactory results based on reconstruction error.

3.1.2 Exploring LUT and AI

LUT represents a more advanced non-contact technique. Here, lasers are used to generate and detect ultrasonic waves. An optical microphone detects the acoustic signal transmitted or reflected from the sample. LUT is especially valuable for materials that are sensitive to contact or difficult to access, such as aerospace components or high-temperature surfaces. Despite the effectiveness of LUT, extracting suitable features from raw data for defect detection remains challenging. The proposed strategy provides a framework for feature extraction from input signals gathered from LUT via autoencoders. Extracted features undergo an additional dimensionality reduction technique (t-SNE, that preserves the local structure of data) while an unsupervised clustering algorithm (DBSCAN) is used for subsequent classification.

Unlike the case of welded joints previously analyzed—where the autoencoder was trained according to one-class supervised approach, using exclusively one of the two classes of samples, in this study a fully unsupervised approach is pursued. The key difference lies in how anomalies are identified: instead of relying on reconstruction errors against a normal-only baseline, the division between anomalous and non-anomalous pixels is carried out directly in the latent space of the autoencoder, employing clustering techniques such as k-means or density-based methods. This allows for a more flexible definition of abnormality, one that does not require a clean separation between “normal” and “anomalous” data during training, which is often unrealistic in industrial inspection settings.

3.1.3 Composite materials and defects detection

Advanced composite materials such as CFRP have been increasingly used in a wide range of industrial and manufacturing applications, mainly due to their excellent characteristics in terms of resistance and stiffness. This led to the large adoption of such material in civil aviation, with aircraft benefitting from

a significant reduction in weight without compromising mechanical properties.

However, despite being closely monitored, the manufacturing process of CFRP-based aviation parts can occasionally result in defects in the final products. This, along with unexpected events, can lead to damages or failures of some aircraft components, posing risks to quality, safety, and performance. Thus, given the safety-critical nature of aviation applications, it is mandatory to define methods to rapidly identify and repair defects that can occur during the manufacturing process in a non-destructive way, without compromising the usability of the tested components.

Several research studies contributed to the field of non-destructive tests NDT. For example, in reference [111], the authors compared several NDT techniques on aerospace industry components, suggesting that combining such methods could enhance the reliability and accuracy of quality control processes. Another innovative proposal involved a micromachined flexible ultrasonic transducer device designed for real-time inspection of aircraft components [112]. By adjusting the diameter of cells, a broader ultrasonic band can target specific frequencies relevant to defect detection. In reference [113], the authors proposed a method for inspecting repaired composites using guided Lamb waves. Signal processing with complex wavelet transform enhanced damage detection, demonstrating promising results.

Defect detection benefited from Computer Vision (CV), which was extensively used to identify surface defects in the manufacturing scenario. Specifically, a combination of 3D reconstruction via laser profiler, with data interpreted using several machine learning classifiers [114] delivered an adequate performance in detecting different anomalies on aluminium welds.

Similar methods were also used in the aircraft scenario, providing good performance in the reconstruction and surface defect detection for aircraft interiors [115].

Still, the fast progress in artificial intelligence and, specifically, deep learning favored the development of new methods and architectures for anomaly detection. For example, the authors in [116] analyzed images from commercially available cameras to locate the surface deformation field via three object detection DL models, that is, R-CNN, Mask R-CNN, and YOLOv3, with the final aim to detect and categorize sub-surface defects.

Another example of an anomaly detection system based on a convolutional autoencoder was proposed in [117] to identify surface defects on aircraft landing gear components; however, the main limitation of this approach was in the requirement for manually determining the threshold required to separate defective samples from non-detective, therefore yielding suboptimal performance.

This was addressed in [2], where several different AEs, based on fully connected, convolutional, and recurrent internal architectures, were compared to detect anomalies on the dataset proposed by [114]. In this case, the threshold for identifying anomalies was automatically determined statistically

by exploiting the first and second moments of the distribution of the reconstruction errors.

An example of a deep anomaly detection model for inspecting composite materials using thermal data was proposed in [118]. Specifically, transfer learning was employed to extract deep features from thermal profiles using sound region features. A method for easily characterizing defects was also proposed, exploiting the chirp z-transform phase-based depth estimation.

A comprehensive framework for defect detection and localization in the automated placement of carbon fibre plies was presented in [119]. The combination of unsupervised DL and classical CV-based algorithms avoided the necessity of having a significant number of labelled samples related to manufacturing defects. Specifically, the laser scanner based on optical coherence tomography technology was adopted for data gathering, and the CV-based algorithm detected individual tows in carbon fibre plies composites, creating a dataset containing images obtained by sliding a window along the centre of each strip. The dataset was used to train an AE for anomaly identification, producing an anomaly map fed to a monodimensional blob detection method to generate bounding boxes for defect localization. Despite the method showing satisfactory detection accuracy while requiring a limited number of images, it was not suited for sub-surface defect detection.

The ultrasonic testing technology was involved in [120], where the authors proposed a single-shot multi-box detector based on CNNs to detect delamination defects commonly occurring in composite materials. Still, a disadvantage of the proposed detection system was in the ability to handle complex defect geometries, providing limited performance when non-laminated defects, such as linear cracks, occurred.

Laminated composites were also investigated in [121], where the authors proposed a real-time, CNN-based NDT method to detect defects using highly nonlinear solitary waves. Specifically, three different types of input signals, i.e., the original signal, a time-sliced variant, and a time-sliced noise-cutting version, were provided to the CNNs, whose performance was fixed by adjusting the number of layers and several hyperparameters, achieving an average score detection of about 90%. Finally, active infrared thermography and automatic semantic segmentation were employed on composite components by [122]. Specifically, a deep neural network composed of several temporal base blocks and max pooling layers was trained on the gathered dataset to classify the points of the specimen, separating defects and background.

Table 3.1 recaps a comparison of the works mentioned above based on the presence/absence of preprocessing, analyzing the inputs, the data types, and the algorithms used by the authors.

Table 3.1: Comparative analysis of the works based on the presence/absence of preprocessing, analyzing the inputs and data types and the approach adopted by the authors. The last row is referred to our work.

Reference	Preprocessing	Input type	Data type	Approach
[106]	Features extracted on ultrasound signals		Real	Supervised, ensemble
[107]	Wavelet transform on A-scans	2D image from wavelet coefficients	Real	Supervised
[108]	Filtering and denoising	A-scans	Real	Supervised
[110]	DBSCAN based gating	B-scans	Real	Unsupervised, based on reconstruction error
[109]	DL-generated scans		Synthetic	Supervised
Our	0–1 Normalization	Raw A-Scans	Real	Unsupervised, based on latent feature space (either normal or t-SNE transformed) clustering via DBSCAN

3.2 The proposed methodology

This chapter addresses the challenge of automatically localizing anomalous areas in CFRP specimens used in aircraft manufacturing using LUT conducted via free-contact transmission, identifying robust and distinctive features via DL-based models, specifically autoencoders. Overall, the main contributions are the following.

- Two methods for robust feature extraction from an ultrasound signal, namely raw A-scans normalized in the 0 – 1 range, based on AEs and variational AEs (VAEs), are proposed.
- The features extracted from the aforementioned methods were leveraged via the DBSCAN and the t-distributed Stochastic Neighbor Embedding (t-SNE) algorithms for latent space clustering and anomaly classification from the background.

The main focus of the approach is on the unsupervised anomaly detection problem while also finding the best trade-off in terms of the computational complexity of the models, bringing innovation due to the simultaneous combination of the following main factors:

- The use of a reduced architectural complexity to deal with raw A-scan data only, facilitating the deployment of the model even on low-power or low-cost hardware.
- The use of real data only, without involving any simulated or interpolated/extrapolated data.
- Our approach works on relatively small portions of unlabeled raw data. It is an unsupervised anomaly detection approach that automatically extracts meaningful features, resulting in an auto-

consistent model.

- The absence of prior knowledge of characteristic parameters in the model and problem formulation.

As already stated, a framework for feature extraction from input signals gathered from laser ultrasonic via AEs is provided. The framework is described in Figure 3.1.

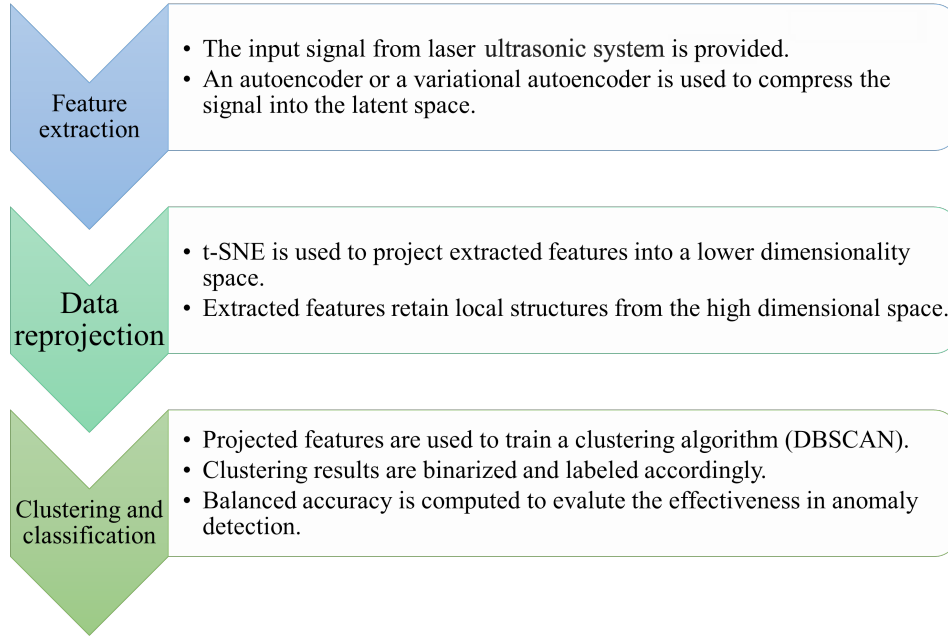


Figure 3.1: The processing pipeline followed by the proposed framework.

Formally, given an input signal $x \in \mathbb{R}^l$, where l is the dimension of the input, a latent representation $z \in \mathbb{R}^{d_{lat}}$ of the input is extracted via a proper deep neural network, with d_{lat} the dimension of the latent space. Once computed, z is used by an unsupervised algorithm to classify it as background or anomalous. To provide an effective description of the proposed method, the dataset acquisition and processing will be shown in Section 3.2.1. Then, feature extraction will be discussed in Section 3.2.2, while the unsupervised algorithm will be described in Section 3.2.3.

3.2.1 Data acquisition and preprocessing

The inspection system was based on the Laser-Excited Acoustics technology from XARION [123], which enabled contact-free ultrasonic testing using fibre-coupled pulsed lasers to generate a broadband ultrasound signal within the sample. The main idea behind the approach is that defects such as delamination or porosity affect signal propagation. Consequently, these effects could be detected via a single-sided (i.e., reflection) or transmission setup.

In this work, a portion of an 8-ply CFRP plate of size $110 \times 130 \text{ mm}^2$ was scanned using the transmission setup shown in Figure 3.3 to detect delamination defects placed at different depths and positions

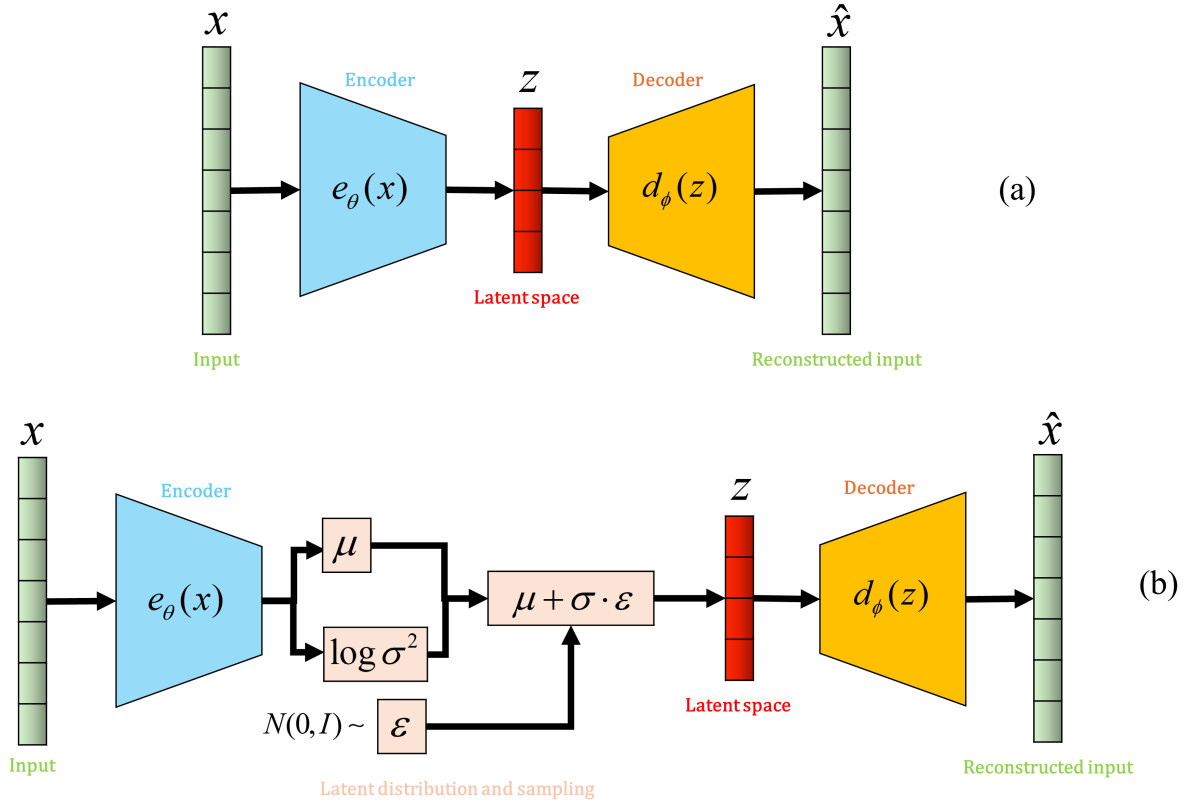


Figure 3.2: The comparison between the architectures of an autoencoder (a) and of a variational autoencoder (b). It is important to underline how the architecture of the VAE shows the reparametrization trick, highlighting how z is sampled.

within the sample. The transmission method allows us to obtain a good signal without requiring a higher level of excitation laser, thus avoiding surface damage or a higher value of microphone gain, which would result in extreme noise levels. Specifically, the microphone scanned the specimen from left to right every 80 s, with a fixed step of 0.2 mm, while acquiring the ultrasound signal (*A-scan* signal), with length $l = 1264$, corresponding to the current position. Then, the microphone shifted 0.5 mm to the row below and started scanning in the opposite direction (i.e., right to left), resulting in a scanned area of size 260×550 pixels. Afterwards, pixels were averaged using a 2×5 window for each time step, consisting of a final dataset of a $130 \times 110 \times 1264$ grid, whose first two dimensions represented the pixel's position. In contrast, the third represented the time step.

As for data preprocessing, the first two dimensions were flattened to provide an overall $n_s \times l$ matrix, where $n_s = 14300$ was the overall number of samples, whose rows represented the signal for each pixel. The elements a'_{ij} of the resulting matrix went through a column-wise normalization within the range $[0, 1]$ to normalize data *time-by-time*, under the reasonable assumption that the same physical conditions characterized the signals at a given time step. A graphical recap of the data preprocessing is reported in Figure 3.4.

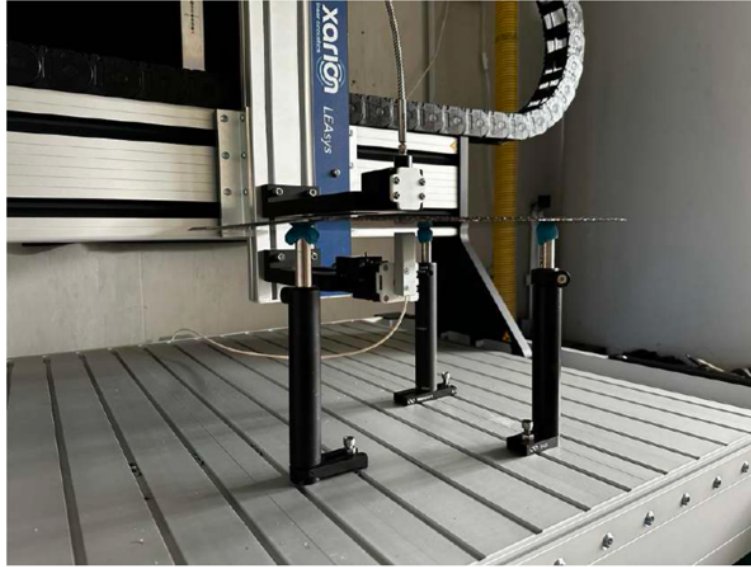


Figure 3.3: Acquisition setup of the ultrasonic laser system in transmission configuration

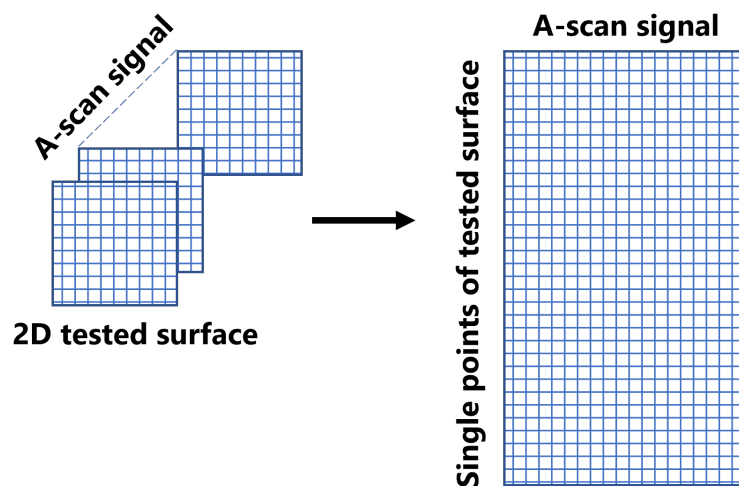


Figure 3.4: Graphical recap of the data preprocessing

3.2.2 Latent feature learning via autoencoders

This section briefly recaps the theory behind AEs and VAEs and the differences in the provided latent feature representation.

An AE is an artificial neural network composed of two parts, that is, an *encoder*, which learns a mapping function from the original space of the input data to a lower-dimensional representation called *latent space*, and a *decoder*, which learns to restore the encoded data to its original size. This is achieved by minimizing a loss function such as the mean average error (MAE), described by the following:

$$L(x, \hat{x}) = \frac{x - \hat{x}}{l}, \quad (3.1)$$

Specifically, in Equation 3.1 x is the original input, while $\hat{x}$ is the reconstructed output. The encoder and the decoder have a mirrored architecture. Hence, the AE is characterized by a bottleneck structure suited for dimensionality reduction and nonlinear feature learning with an unsupervised approach.

As for VAEs, this type of AE maps the input into a distribution instead of a latent space, thus introducing a probability distribution learning mechanism between the encoder and the decoder. In a VAE, the encoder maps a point $\bar{x}$ in the original space to the conditional probability $p(z|\bar{x})$, describing the (posterior) distribution of encoded variable z in the latent space when the point $\bar{x}$ is picked up in the original space. Then, the decoder maps a point $\bar{z}$ sampled from $p(z|\bar{x})$ to a point $\hat{x}$ sampled from the conditional probability $p(x|\bar{z})$, describing the (likelihood) distribution of decoded variable x conditioned by $\bar{z}$. Bayes' theorem relates the posterior and likelihood distributions:

$$p(z|x) = \frac{p(x|z)p_l(z)}{p(x)}, \quad (3.2)$$

where $p_l(z)$ and $p(x)$ are, respectively, the latent (prior) and input space (observed data) distributions, and $p_l(z)$ is assumed to be a standard Gaussian distribution $N(0, 1)$.

During the training phase, a VAE models two encoding functions $\mu_\theta(x), \sigma_\theta^2(x)$ and a decoding function $d_\phi(z)$, parametrized by a set of encoding parameters $\{\theta\}$ and decoding parameters $\{\phi\}$ that the neural network learns while pursuing two objectives:

- The minimization of the Kullback-Leibler (KL) divergence of an inference distribution $q(z|x)$, resembling a normal distribution $N(\mu_\theta(x), \sigma_\theta^2(x))$, from the posterior distribution $p(z|x)$.
- The maximization of the probability that the reconstructed output $\hat{x}$ is equal to the input x , which corresponds to maximising the likelihood, assumed to be still normally distributed, $p(x|z) = N(d_\phi(z), (2\lambda)^{-1})$, when z is picked up from the inference distribution.

These two requests correspond to minimize the cited KL divergence. This leads to the final loss function $L'(x, \hat{x})$, which can be written as:

$$L'(x, \hat{x}) = L(x, \hat{x}) + \frac{1}{2} (\mu_\theta^2(x) + \sigma_\theta^2(x) - \log \sigma_\theta^2(x) - 1). \quad (3.3)$$

Where $L(x, \hat{x})$ is the contribution of the reconstruction error. Figure 3.2 highlights the differences between AEs and VAEs.

It is important to underline how sampling from the inferred distribution in the VAE would make the error impossible to backpropagate through the network. To deal with this issue, the so-called

reparametrization trick was introduced; that is, instead of sampling z from $q(z|x) = N(\mu_\theta(x), \sigma_\theta^2(x))$, one samples ϵ from a standard normal distribution, hence $z = \mu_\theta(x) + \epsilon, \sigma_\theta(x)$. This way, the values in z are still normally distributed around the mean $\mu_\theta(x)$, except for those at a distance larger than σ , which are discarded.

Given the probabilistic nature of VAEs, this architecture is mainly used for generative purposes. In anomaly detection, VAEs can be compared to the classical approach while pursuing an improvement of classification in the latent space. Indeed, a well-known advantage of VAEs, if compared to AEs, is regularizing the latent space. Specifically, the embedded vectors in a VAE are arranged with continuity: points close in the latent space will be similarly reconstructed by the decoder, and the variations of the reconstructed outputs become smooth while moving across the latent space.

Feature extraction

The features were extracted using two different strategies. The first is based on training a standard AE, whose encoder was used during inference to build the latent space. One obvious drawback of this strategy is that the latent space is not regularized. Thus, close points in the original space can be distant in the latent space. This problem was solved via the second strategy, which proposed training a VAE and a standard AE. This is related to the probabilistic nature of the VAE, which associates each input sample to the mean and variance of a normal distribution. Thus, the approach based on VAE requires an additional step. First, the VAE is trained using the complete dataset, allowing the decoder to learn to map each point of a well-regularized latent space to the original sample space. The encoder of the VAE is discarded, while its decoder is used with frozen weights in a standard AE. During this step, the encoder of the new AE learns to map each sample to its embedded counterpart in the latent space. At this point, it should be clear that only the encoder is involved during inference in both strategies. Let us also note that even if the reconstruction quality was not the focus of this work, a good reconstruction ability could be associated with efficient latent feature learning. For this reason, a test set for measuring the reconstruction quality was unnecessary; rather, the whole dataset constituted the test set in the form of an anomaly map compared with the ground truth.

3.2.3 Clustering and classification in the latent space

Algorithm and hyperparameters selection

The features learned by the AE and the VAE were clustered in the latent space to distinguish between anomalies and background. To this end, the DBSCAN algorithm was selected. The algorithm works by iteratively grouping together data points that are closer than a certain threshold and requires two

parameters:

- ϵ is the maximum distance between two instances to be considered neighbors;
- n_{min} is the minimum number of samples in the ϵ -neighborhood of an instance to classify the latter as a *core instance*.

At the end of the iterations, DBSCAN guarantees that all the instances in the ϵ -neighborhood of a core instance belong to the same cluster; at the same time, any instance that is not a core instance and does not have one in its neighbourhood is considered an outlier. The DBSCAN algorithm was selected over other clustering methods mainly due to its robustness when dealing with clusters whose overall number and distribution characteristics are unknown beforehand.

In this work, ϵ was selected to maximize the *balanced accuracy* between the ground truth labels and the binarized labels predicted by DBSCAN. Specifically, a context-dependent assumption about the majority of the samples belonging to non-anomalous class can be made. At the same time, a minority part refers to all the different types of anomaly found within the dataset. The optimum values were found after a grid search procedure performed starting from a distance metric $\Delta(\cdot)$ computed on the whole transformed instances z to obtain and then considering the average value of all distances and the maximum one respectively as lower and upper bounds for the grid search.

A grid search was carried out to find this optimum, considering a range of values with lower and upper boundaries equal to the average and maximum distance between all instances.

Evaluation metrics

The *balanced accuracy* a_{bal} was selected as the metric to evaluate the effectiveness of DBSCAN and select proper values for ϵ . The balanced accuracy was selected given the characteristics of the scenario under analysis. In other words, even if an unsupervised clustering algorithm was selected to define how feature vectors are distributed, a supervised evaluation was possible, as the ground truth was known beforehand.

To this end, selecting the metric to use had to consider the bias introduced by scenarios such as anomaly detection, in which the dataset is highly unbalanced. In other words, given the minority of one class (that, in this case, is the positive class of anomalies), a high value for standard accuracy can be achieved even if the rate of true prediction of this class is low. To deal with this issue, balanced accuracy rectifies standard accuracy, weighting the number of positive and negative correct predictions, i.e., *true positives* (TP) and *true negatives* (TN), as follows:

$$a_{bal} = \frac{1}{n_s} \left(TP \frac{n_s}{2P} + TN \frac{n_s}{2N} \right), \quad (3.4)$$

Where $n_s = TP + TN + FP + FN$, i.e., the size of the whole dataset. For a perfectly balanced dataset, the number of actual positives (P) and actual negatives (N) is the halved number of samples, and the balanced accuracy becomes just the standard accuracy. Equation 3.4 can also be written as:

$$a_{bal} = \frac{1}{2} \left(\frac{TP}{P} + \frac{TN}{N} \right), \quad (3.5)$$

Equation 3.5 shows how the balanced accuracy corresponds to the mean between the recall r_P of positive samples (sensitivity) and the recall r_N of negative samples (specificity). Besides the balanced accuracy and the recall, the precision p is computed. This is defined as the rate of true positives TP among all detected positives $TP + FP$. Finally, the performances of each architecture will be evaluated in terms of F_1 score, which is defined as the harmonic mean between precision and recall:

$$F_1 = 2 \left(\frac{1}{p} + \frac{1}{r_P} \right)^{-1}. \quad (3.6)$$

t-SNE on encoded data

An additional step is performed on latent vectors before applying DBSCAN, consisting of projecting the latent representation on a lower dimensional space using the t-distributed Stochastic Neighbor Embedding (t-SNE) algorithm [124]. Specifically, t-SNE is a non-linear dimensionality reduction technique that preserves the local structure of data. First, a normal distribution $p_{j|i}$ is defined for each point $\mathbf{x}_i$, this describes how the remaining points $\mathbf{x}_j$ distribute around it and it is defined as:

$$p_{j|i} = C \exp \left(\frac{\|\mathbf{x}_i - \mathbf{x}_j^2\|}{2\sigma_i^2} \right) \quad (3.7)$$

where C is a normalization factor such that $\sum_{j \neq i} p_{j|i} = 1$. Then, a similarity measure p_{ij} between two points $\mathbf{x}_i$ and $\mathbf{x}_j$ is introduced:

$$p_{ij} = \begin{cases} \frac{p_{i|j} + p_{j|i}}{2n_s} & \text{for } i \neq j \\ 0 & \text{for } i = j \end{cases} \quad (3.8)$$

Given the pairwise similarities in the original space, t-SNE aims to learn a map $\{\mathbf{x}_1, \dots, \mathbf{x}_{n_s}\} \rightarrow \{\mathbf{y}_1, \dots, \mathbf{y}_{n_s}\}$, with $\mathbf{y}_i \in \mathbb{R}^d$, that minimize the KL divergence from a similarity measure q_{ij} in the new embedded space, which follows the Student t-distribution with one degree of freedom:

$$q_{ij} = \begin{cases} C' (1 + \|\mathbf{x}_i - \mathbf{x}_j\|^2)^{-1} & \text{for } i \neq j \\ 0 & \text{for } i = j \end{cases} \quad (3.9)$$

Here C' is another normalization factor such that $\sum_{ij} q_{ij} = 1$. The most important parameters in a t-SNE algorithm are the already cited dimension d of the embedded final space and the perplexity $\mathcal{P}$, which is defined as

$$\mathcal{P} = 2^{-\sum_j p_{j|i} \log_2 p_{j|i}}, \quad (3.10)$$

For every i , t-SNE performs a binary search for the values of the bandwidths of Gaussian kernels σ_i , resulting in the specified value of $\mathcal{P}$.

3.3 Experiments and Results

3.3.1 Models architectures

The two families of feature extractors compared in this chapter were designed with closely resembling architectures. Design similarity is connected to the fact that both structural components of the model and the methodologies employed to build it share a significant degree of commonality to ensure that their functionalities and performance are consistent for the anomaly detection task. Specifically, the architectures of the AE used in the first approach were composed of:

- A fixed encoder made of a dense layer with 1264 neurons and ReLU activation function;
- A latent space whose dimensions varied from 3 to 12 neurons;
- A fixed decoder composed of a dense layer with 1264 neurons and a sigmoid activation function.

As for the VAE architectures evaluated in the second approach, the following building blocks were used:

- The encoder comprised two fixed parallel dense layers with the same number of neurons of the latent space each, one for computing μ and the other for computing σ^2 ;
- The dimension of the latent space representing μ and σ^2 varied from 3 to 12 neurons;
- The fixed decoder comprised a dense layer and a sigmoid activation function.

3.3.2 Experimental settings

As for training, all the kernels and biases for all the layers of the architectures were initialized according to the approach suggested in [125]. The implemented variational and standard autoencoders were trained with 100% and 90% of the available data, respectively, as for the considerations already described in Section 3.2, and the Adamax optimization algorithm was used. All the architectures were

trained for 50 epochs with a fixed batch size of 32. The following parameters were set: learning rate $= 10^{-3}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-7}$.

For what concerns the DBSCAN algorithm, its parameters were set as follows. First, the number of samples n_{min} was set equal to 30, with the Euclidean distance used to compute the distance between data samples. As stated in Section 3.2, ϵ was optimized using a grid search procedure. Furthermore, for the t-SNE, a value of $d = 2$ was selected, starting from the assumption that a final bidimensional embedded space implied that data could be easily visualized using a Cartesian space; as for the perplexity, it was experimentally set to maximize the achieved results.

Finally, the software used for the experiments was written in Python, using Keras 3.3 and TensorFlow 2.16.1. All the models were trained on a machine equipped with an Intel Core i9-14900HK, 32 GB of RAM, and an RTX 4090 with 32 GB of VRAM.

3.3.3 Results

Results are provided in Tables 3.2 and 3.4 for the first and the second approaches, respectively. Each Table highlights how the balanced accuracy a_{bal} and the F_1 score change with the increase of the number of neurons within the latent space while using an AE (Table 3.2) and a VAE (Table 3.4), respectively. Both Tables also report the value of ϵ found by the optimization procedure described in section 3.2.3 and compare the results *before* (i.e., *w/o t-SNE*) and *after* (i.e., *after t-SNE*) the application of t-SNE. The best results are highlighted in bold over both tables. In addition, it is worth reporting the details of the precision and recall values achieved by the different models due to the variation of the parameter d_{lat} and the application of t-SNE that are respectively depicted in Figure 3.5 for the autoencoder and in Figure 3.6 for the variational autoencoder. From a numerical point of view, it is worth noticing that, in both cases, the recall values are higher than 86% independently from the effects of the t-SNE transformation. A second observation is that precision values are generally higher when t-SNE is applied, reflecting $\approx 10\%$ higher values of the F_1 score metric. Regarding the variational autoencoder architecture, both precision and recall values increase proportionally with the $d_{lat} \in [3, 7]$ when t-SNE is not applied before forming a visible cluster around coordinates $(P, R) = (50, 93)$. Notably, the application of t-SNE mainly affects the recall values since precision ones are the highest reported. Finally, it is worth highlighting the presence of one outlier in the upper left part of Figure 3.6, corresponding to the lowest F_1 score achieved by all the tested architectures. Details about inference time are reported in Table 3.3, showing that even if the VAE architecture is more complex than the AE one, the difference between the two architectures is neglectable when the model is used in inference. Finally, an insight into the performance of all the architectures tested during the experiments is reported in Figure 3.7, in which a boxplot of the mean absolute reconstruction error of each analyzed A-scan is reported. In the

figure, the X-axis labels indicate whether AE or VAE are considered and report the d_{lat} value for the latent feature space dimension. It is worth noticing that reconstruction errors are generally low. Still, there is a certain number of outliers, meaning that both AE and VAE architectures are not overfitting on input data when the original signal is passed and reconstructed through the deep learning models. Another remark should be given about VAE reconstruction error boxes that are always slightly higher in value and bigger than the correspondent AE ones, showing that the variational autoencoder approach introduces context-aware data sampling and generation so it can be considered a sort of data-driven data augmenter. However, the final results in our problem formulation must be evaluated regarding the clustering capabilities of dividing the background signal from the anomalies.

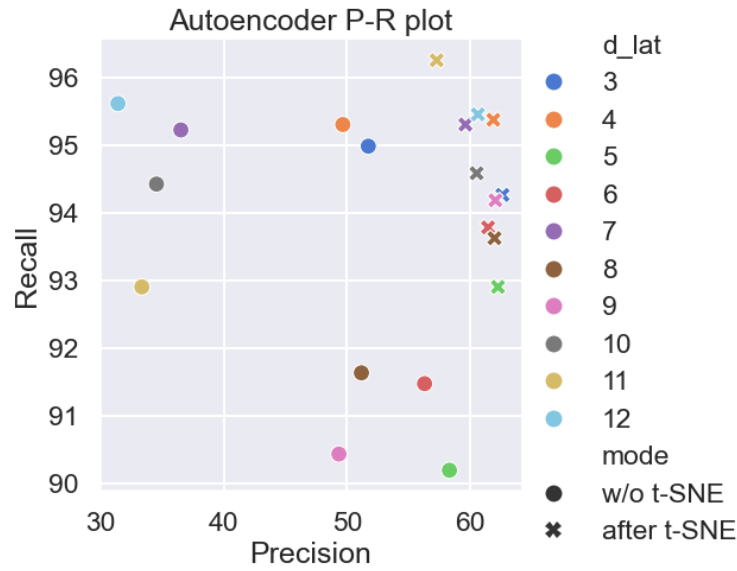


Figure 3.5: Precision-Recall scatter plot of the AE model for all the values of d_{lat} . The dots refer to the scores achieved before the application of t-SNE, whilst the cross marks are related to the scores after the t-SNE application.

The VAE could provide slightly better results regarding the F_1 score, with $F_1 = 75.47\%$, when t-SNE was applied and a latent space of 11 neurons was considered. At the same time, the AE could yield the best overall value for balanced accuracy, $a_{bal} = 94.87\%$, when a total of 4 neurons was considered for the latent space along with applying t-SNE. Interestingly, the application of t-SNE provided improved results in most situations. This is probably due to the specific capability of the t-SNE to capture local data structures. In other words, the t-SNE retains the locality of the points in the initial latent space, keeping close points nearby after the projection while pushing away pairs of instances with low similarity. This perfectly fits the operation of DBSCAN, providing an optimal preprocessing for density-based clustering.

Figures 3.8 and 3.9 show clustering results on data transformed via t-SNE. These figures can be

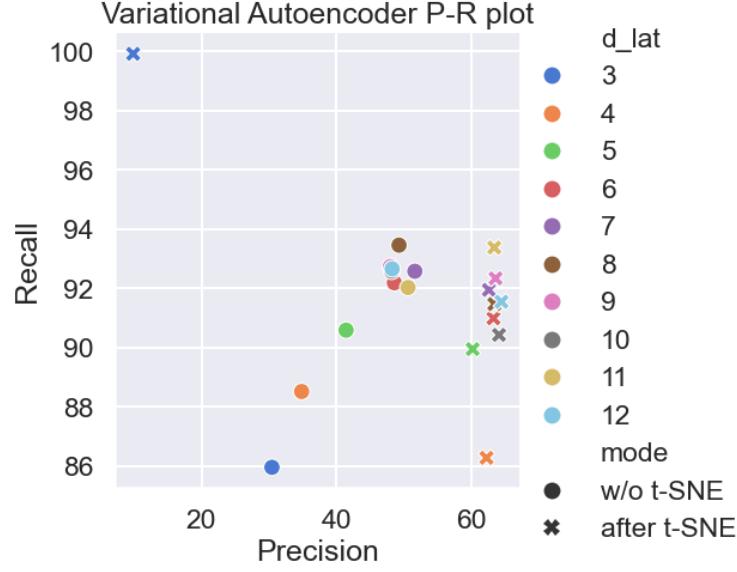


Figure 3.6: Precision-Recall scatter plot of the VAE model for all the values of d_{lat} . The dots refer to the scores achieved before the application of t-SNE, whilst the cross marks are related to the scores after the t-SNE application.

Table 3.2: Balanced accuracy, F_1 score and ϵ -value that optimizes a_{bal} for each tested latent space dimension for the AE architecture, before and after application of t-SNE on latent space.

d_{lat}	w/o t-SNE			after t-SNE		
	a_{bal}	F_1	ϵ	a_{bal}	F_1	ϵ
3	93.23	66.99	0.41	94.43	75.26	2.92
4	93.01	65.30	0.48	94.87	75.07	2.93
5	92.00	70.84	0.63	93.75	74.56	2.60
6	92.32	69.71	0.68	94.07	74.26	2.60
7	89.65	52.78	0.65	94.55	73.36	2.39
8	91.61	65.68	0.77	94.05	74.59	2.80
9	90.75	63.85	0.82	94.32	74.82	3.29
10	88.60	50.57	0.76	94.33	73.83	2.49
11	87.53	49.07	0.79	94.68	71.83	2.28
12	87.77	47.27	0.80	94.75	74.16	2.39

compared with Figures 3.10 and 3.11, resulting from applying DBSCAN directly on encoded data. Figure 3.12 depicts the binarized version of anomaly maps in Figures 3.8, 3.9, 3.10, 3.11. Here, it can be noticed that the results of the anomaly map obtained by applying the DBSCAN after the t-SNE are much less noisy. From the inspection of anomaly maps, one can see that the left and right boundaries of the plate are detected as anomalies. This behaviour reduces the precision value and then the F_1 score but reflects the difference of UT signals in the side boundaries from the rest of the plate.

In summary, most of the proposed architectures achieved values of a_{bal} and F_1 score above 90% and 70%, respectively, except for the variational autoencoder with $d_{lat} = 3$, which shows very low performances. In this case, the optimization procedure for the parameter ϵ of DBSCAN on TSNE data

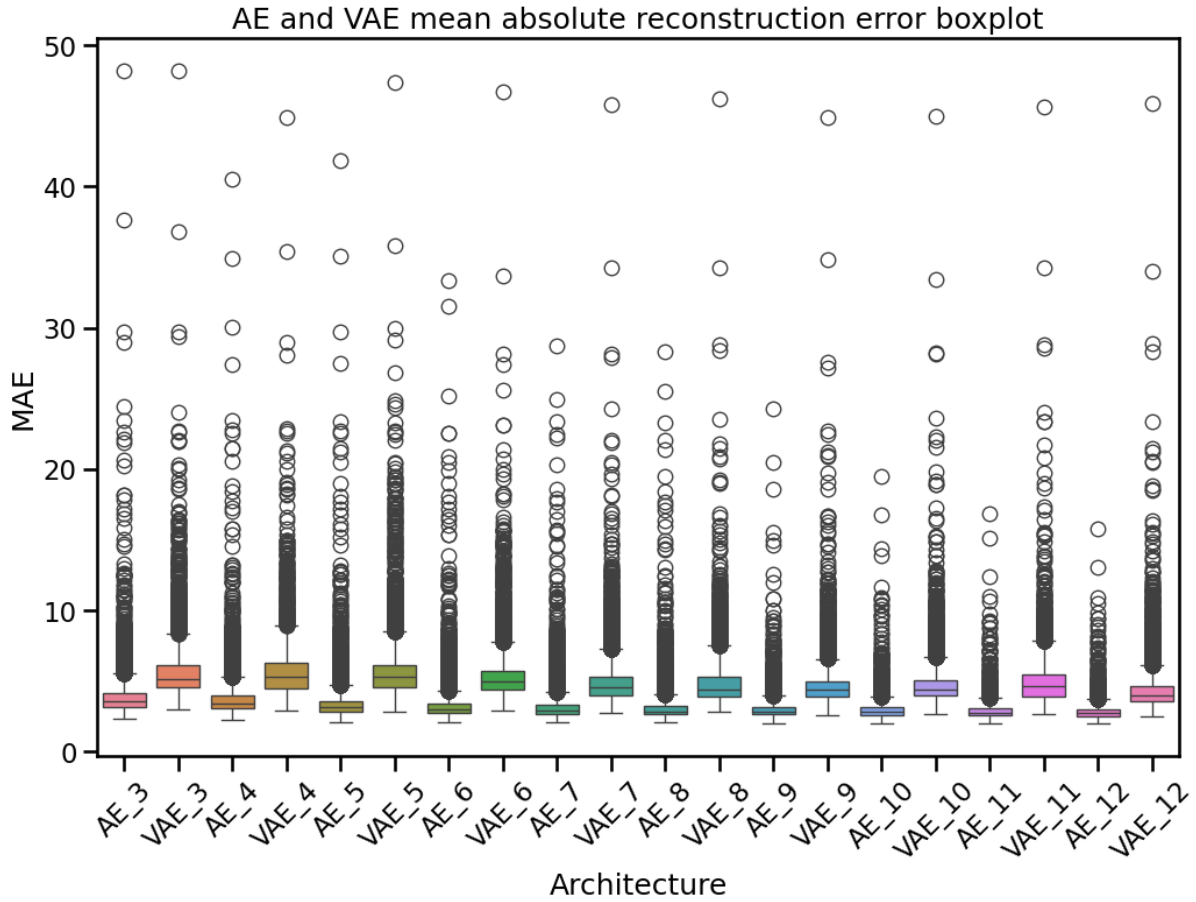


Figure 3.7: Boxplot of mean absolute reconstruction error for all the architectures tested. The X-axis labels indicate whether AE or VAE are considered and also report the d_{lat} value for the latent feature space dimension

Table 3.3: Average inference time μ and standard deviation σ over 100 model queries for both AE and VAE architectures.

d_{lat}	AE		VAE	
	μ [s]	σ [s]	μ [s]	σ [s]
3	4.758e-04	6.321e-05	4.657e-04	1.234e-04
4	5.263e-04	8.694e-05	1.235e-03	6.763e-03
5	6.076e-04	1.490e-04	7.770e-04	2.245e-03
6	5.396e-04	7.500e-05	5.383e-04	1.003e-04
7	5.286e-04	9.503e-05	5.671e-04	1.259e-04
8	5.587e-04	1.611e-04	5.904e-04	1.096e-04
9	5.361e-04	8.269e-05	5.745e-04	1.566e-04
10	5.353e-04	8.159e-05	5.483e-04	9.233e-05
11	5.382e-04	8.305e-05	5.984e-04	1.642e-04
12	5.448e-04	8.630e-05	5.465e-04	8.003e-05

results in a very fragmented clustering with no background class emerging as the most numerous, as shown in Figure 3.13. Enlarging the value of ϵ , the number of detected classes decreases, but still, not all the anomalies can be detected, as shown in Fig. 3.14. Arguably, it can be concluded that three latent features can be used by the variational architecture to create a meaningfully separated regularized

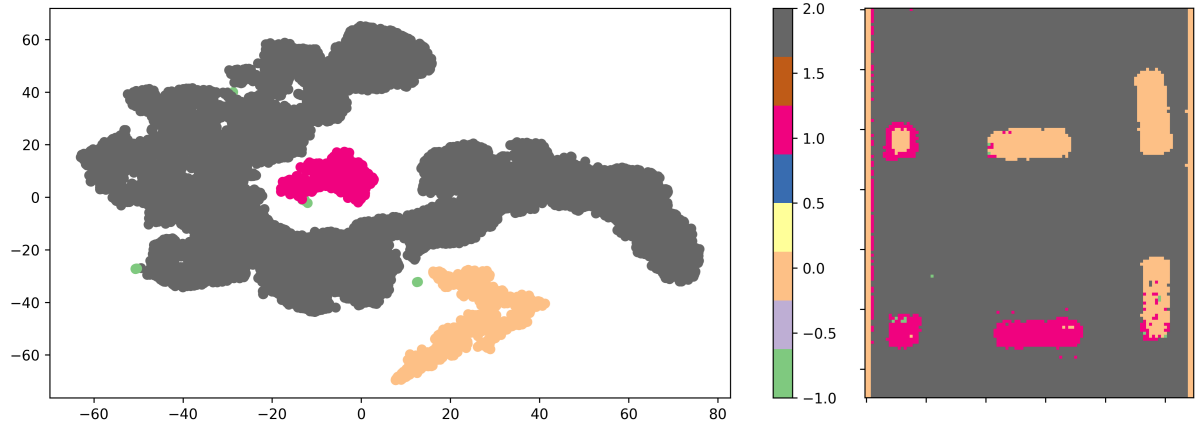


Figure 3.8: Latent space (on the left) resulted from the application of DBSCAN on t-SNE transformed data encoded by standard autoencoder with $d_{lat} = 3$ and correspondent anomaly map (on the right).

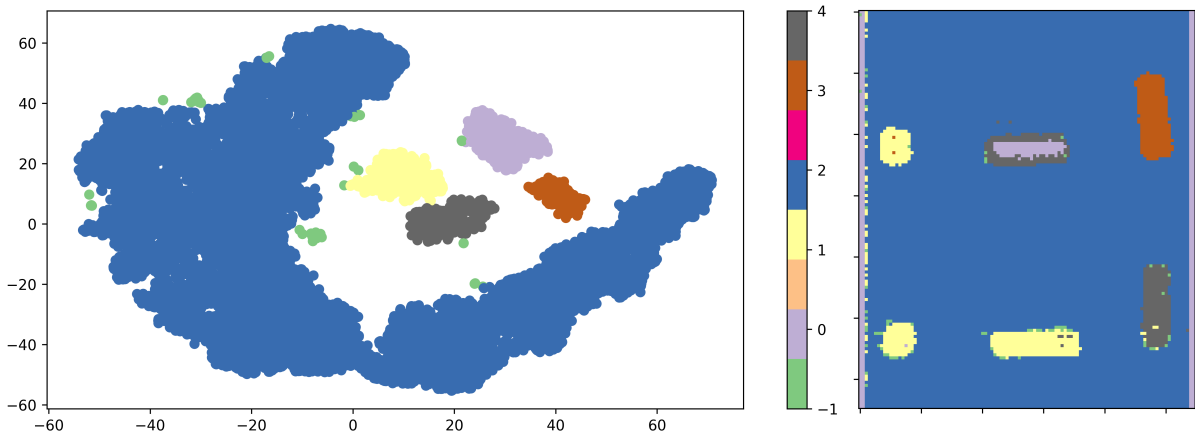


Figure 3.9: Latent space (on the left) resulted from the application of DBSCAN on t-SNE transformed data encoded by variational autoencoder with $d_{lat} = 11$ and correspondent anomaly map (on the right).

space, given the acceptable results achieved without t-SNE; however, the application of a further dimensionality reduction step leaves out a significant amount of the original data variability, therefore yielding a significant reduction in the achievable results.

Nevertheless, the discussion of this case remains useful in highlighting the ability of autoencoders to reconstruct the input and to extract meaningful features to separate the anomalies from the background signals effectively. This can be qualitatively verified by comparing Figure 3.15, representing the value of all the input samples at two fixed time steps, with the anomaly map in Figure 3.14: the network learns to reconstruct the input samples capturing the variations of the signals along the plate and separating in the latent space pixels whose correspondent signals assume almost the same value, at a certain time, from pixels whose signals are significantly different. The aim is to highlight the nature of the acquired signals, in which the anomalies could be difficult to highlight when they are represented with their original spatial shape, to clarify better the capability of the deep learning architecture to achieve

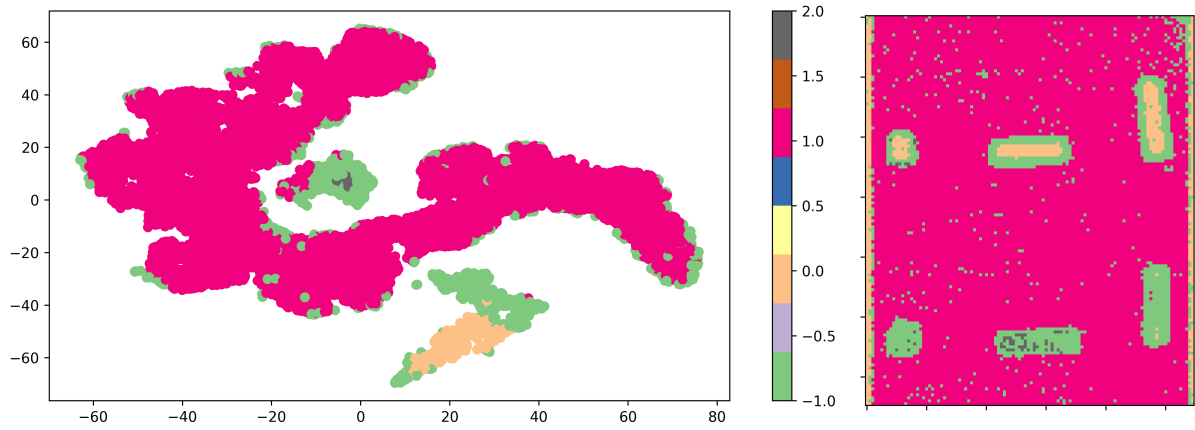


Figure 3.10: Anomaly map (on the right) resulting from the application of DBSCAN on data encoded by standard autoencoder with $d_{lat} = 3$ and correspondent clustered latent space visualized via t-SNE (on the left).

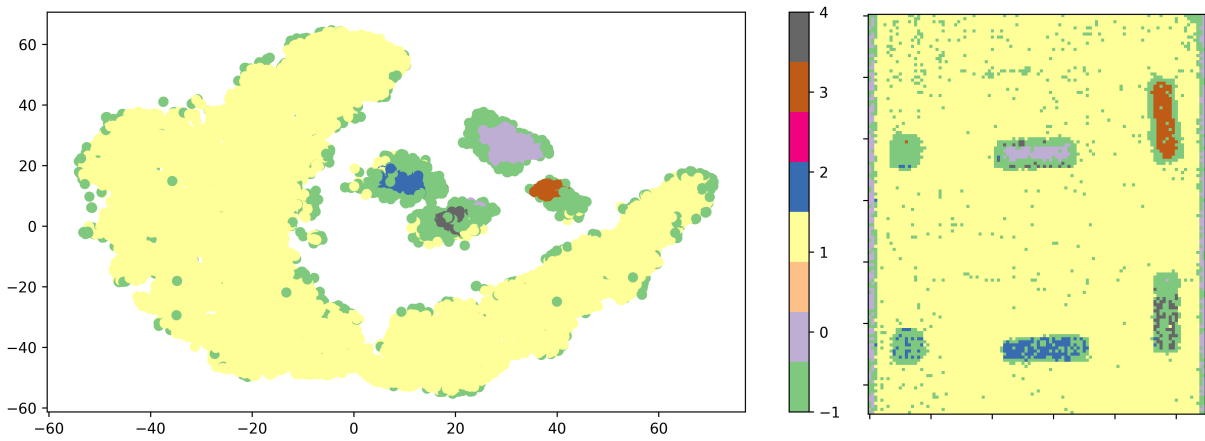


Figure 3.11: Anomaly map (on the right) resulting from the application of DBSCAN on data encoded by variational autoencoder with $d_{lat} = 11$ and correspondent clustered latent space visualized via t-SNE (on the left).

the results presented in this work when all the A-scan signals are considered as separate instances of the input dataset, without considering any other spatial relationship or constraint. This capability is common to all the architectures and emerges when the parameter ϵ is small enough. However, let us remember that, while interesting and suitable for future works, creating subclasses in the background was outside of the scope of this specific set of experiments.

3.3.4 A comparison with Wavelet transform-based feature extraction

As a comparison, the approach based on the Wavelet transform presented in [126], where active ultrasonic technology is still used to inspect composite components, can be discussed. Specifically, the Wavelet domain was exploited to extract significative and distinctive features from the LUT, which were then used to feed different machine learning-based methods, yielding high accuracies in anomaly

Table 3.4: Balanced accuracy, F_1 score and ϵ -value that optimizes a_{bal} for each tested latent space dimension for the VAE architecture, before and after application of t-SNE on latent space.

d_{lat}	w/o t-SNE			after t-SNE		
	a_{bal}	F_1	ϵ	a_{bal}	F_1	ϵ
3	83.54	44.96	3.30	56.28	18.03	2.25
4	86.30	49.99	6.42	90.62	72.27	2.64
5	89.14	56.84	7.04	92.11	72.08	2.63
6	91.40	63.60	9.19	92.95	74.60	2.82
7	92.11	66.25	8.86	93.32	74.44	2.45
8	92.10	64.50	9.78	93.19	74.85	3.15
9	91.53	63.22	10.45	93.62	75.27	2.75
10	91.50	63.35	11.33	92.77	74.97	2.64
11	91.69	65.27	12.61	94.09	75.47	2.73
12	91.55	63.43	11.43	93.34	75.60	2.64

detection. In this approach, the signal is analyzed in the time–frequency domain through spectrogram decomposition, and features are extracted by concatenating the spectral information along both the horizontal and vertical axes of the spectrogram. This process yields a distinctive signature that encodes the structural response of the inspected component. The methodology provides a rich and detailed description of the ultrasonic signal, making it possible to capture subtle variations that may indicate the presence of defects.

A comparative analysis between this approach and that one proposed in this section highlights their respective advantages and trade-offs. Autoencoders excel at feature compression: the latent embeddings are compact, low-dimensional representations that still preserve the essential structure of the data. This makes them particularly efficient for tasks where computational cost and memory constraints are critical, such as real-time or large-scale inspections. On the other hand, the wavelet–spectrogram method, while slightly more demanding in terms of data dimensionality, can offer higher sensitivity to certain types of anomalies. By preserving a larger amount of frequency–temporal information, it achieves marginally better detection metrics, especially in cases where the defects manifest as localized, subtle perturbations in the ultrasonic response.

Nevertheless, this improvement comes at a cost. The concatenated spectrogram-based signature is considerably longer than the embeddings produced by the autoencoder, leading to heavier storage requirements and potentially longer processing times. In industrial applications, this trade-off must be carefully evaluated: while enhanced detection accuracy is desirable, efficiency and scalability are equally crucial. For instance, in the inspection of large aeronautical components where millions of data points may be collected, the computational overhead of long feature vectors may pose practical limitations. In contrast, autoencoder embeddings provide a compressed representation that facilitates faster clustering, easier visualization, and more manageable downstream analysis.

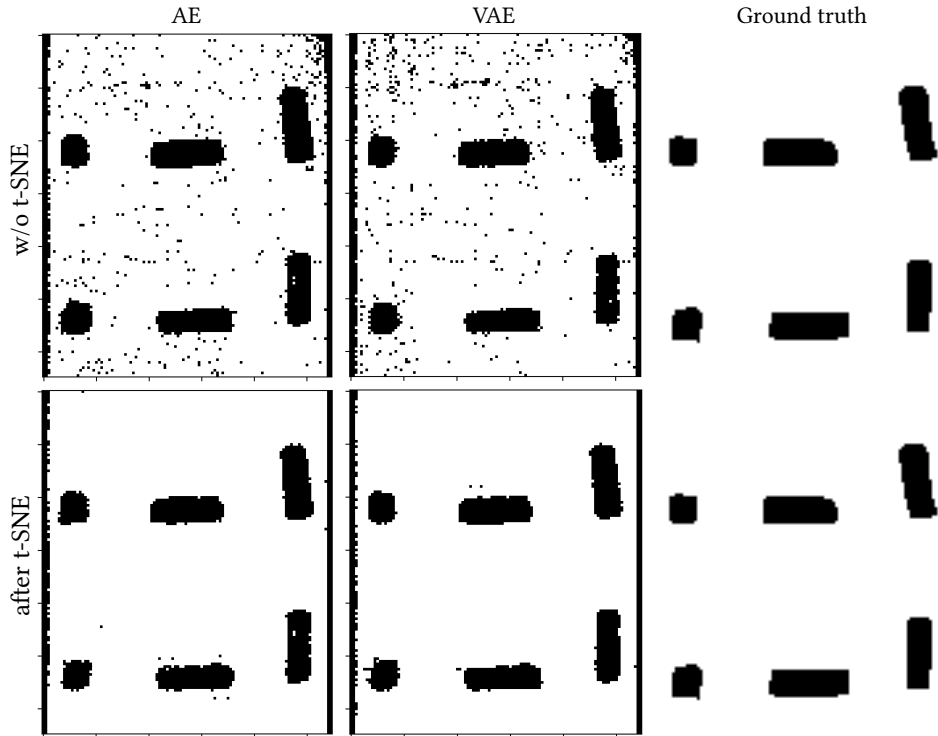


Figure 3.12: Results of the binarized anomaly map with ground truth reference for the best architectures. The first row contains the anomaly map computed by applying the DBSCAN algorithm directly on the feature space predicted by the feature extractor (AE first column and VAE second column). In the second row, the DBSCAN was applied to the result of the t-SNE algorithm.

Taken together, the two approaches illustrate complementary strategies for unsupervised anomaly detection in laser UT of aeronautical components. The autoencoder-based method emphasizes efficiency and compact representation, whereas the wavelet–spectrogram approach prioritizes richer signal characterization at the expense of feature length. Their juxtaposition highlights the importance of balancing accuracy and feasibility in the design of NDT pipelines, especially in critical industrial sectors such as aerospace where both precision and practicality are essential.

3.4 Summary

In this chapter an anomaly detection framework has been proposed. This framework was based on a unified DL pipeline involving different AE and VAE architectures for feature extraction and leveraging unsupervised clustering algorithms. Specifically, two autoencoder-based approaches were used for feature extraction and extensively tested to identify the best hyperparameter selection for determining the correct size for the latent space. The extracted features were then processed by an unsupervised clustering algorithm (DBSCAN) in combination with a dimensionality reduction mechanism (t-SNE) mainly used for data visualization, but that, in this context, helped with the regularization of the latent

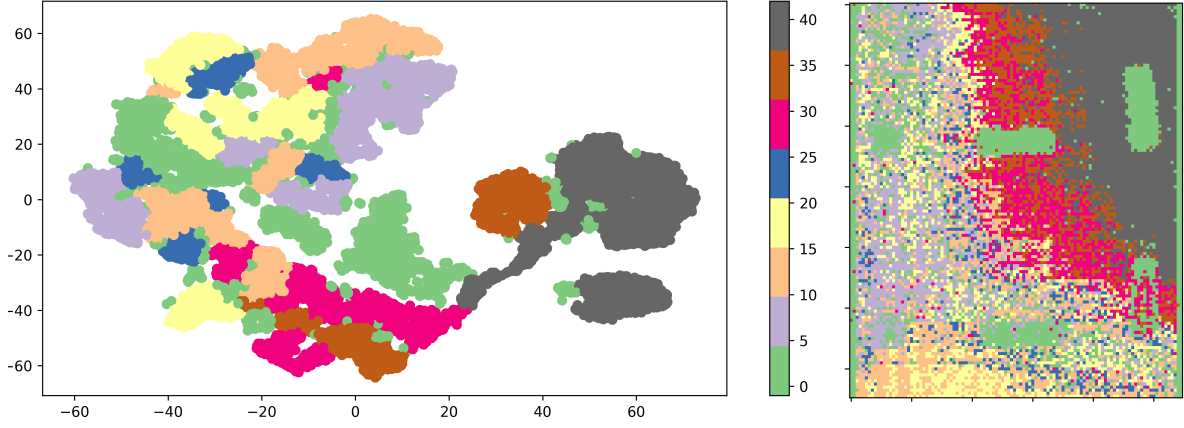


Figure 3.13: Latent space (on the left) resulted from the application of DBSCAN on t-SNE transformed data encoded by variational autoencoder with $d_{lat} = 3$ and correspondent anomaly map (on the right). Since the detected classes are more than the colours available in the colormap, the results for some classes appear to be graphically merged.

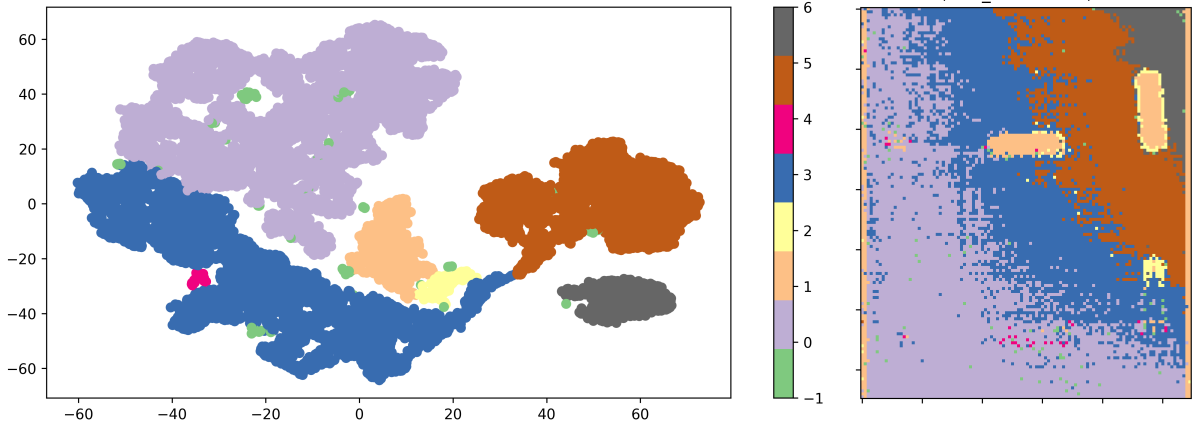


Figure 3.14: Latent space (on the left) resulted from the application of DBSCAN on t-SNE transformed data encoded by variational autoencoder with $d_{lat} = 3$ and correspondent anomaly map (on the right). In this case, no optimization procedure has been followed for the parameter ϵ , which has been set to 2.6.

feature space. Quantitative results show that while both the proposed feature extraction approaches can be effectively used as the backbone of the method, the usage of t-SNE in combination with DBSCAN achieved the best performance, with a maximum balanced accuracy of 94.87% and a F_1 -score of 75.26% with the autoencoder backbone, and balanced accuracy of 94.09% and a F_1 -score of 75.47% when using the variational autoencoder backbone.

Future works could be focused on addressing the limitations of the current proposal. For example, the dataset extension issue can be addressed by including the evaluation pipeline of more complex samples acquired under varying conditions and subject to an unbiased set of preliminary assessments. Using simulated data generated via deep generative models or synthetic knowledge can also be evaluated. The specificities of the model can also be extensively evaluated in different domains, considering the

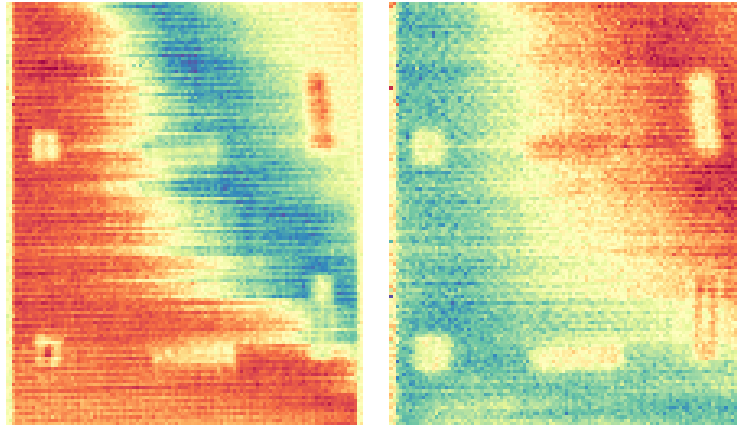


Figure 3.15: Input map at frame 441 (left) and 497 (right), representing the normalized value of all the input signals at those time steps.

possibility of applying the acquired knowledge in different scenarios even with constrained hardware, adopting an optimal tradeoff between computational complexity, deployment feasibility, and representation capabilities. Particular attention can be given to new specializations of the models to classify the anomalies further, providing a more in-depth error analysis that will also focus on the types of misclassified anomalies and their potential impact on the domain. Furthermore, the framework's effectiveness can also be tested in similar domains.

Beyond the direct methodological comparison, it is worth reflecting on possible extensions of laser UT technology in the field of robotics. The Laser-Excited Acoustics (LEA) system used in this work currently requires that the panel under inspection be fixed in place, and industrial demonstrations already exist where robotic cells are used to automate the loading, positioning, and clamping of components within the optical measurement area. Such integration offers significant advantages in terms of throughput, repeatability, and traceability, particularly in aerospace manufacturing environments where high inspection volumes are required. At the same time, an alternative vision foresees the deployment of a robot-mounted inspection head, in which the laser excitation and optical detection units are integrated onto a robotic arm [127]. This approach would alleviate the need to reposition or clamp large or complex structures, enabling flexible on-site inspection of aeronautical components or other critical assets. Both directions illustrate how laser-based NDT can be synergistically combined with robotics: either through automated handling of fixed components in inspection cells or through robotic mobility of the inspection probe itself. In both cases, the convergence of laser ultrasonics and robotic automation highlights a pathway toward scalable, autonomous, and efficient non-destructive evaluation in advanced industrial contexts.

Chapter 4

Ground Penetrating Radar for moisture estimation and navigation in agricultural soil

Detecting the water content in agricultural soil is critical for optimizing irrigation, improving crop yield and promoting sustainable water management. Traditional soil moisture measurement methods are often labour-intensive, invasive, or limited in spatial coverage. Ground Penetrating Radar (GPR) offers a promising alternative due to its non-invasive nature, high spatial resolution, and ability to cover large areas efficiently. This chapter explores the application of synthetic and real GPR data and deep neural networks, such as autoencoders, for assessing relative permittivity and conductivity of agricultural soils, depending on their water content. The integration of such models in unmanned vehicles could enable real-time monitoring for addressing the challenges of water management in agriculture.

In order to systematically study the relationship between soil physical parameters and the corresponding GPR response, two case study will be described. The first study employed gprMax, an open-source software package widely adopted for the forward modeling of EM wave propagation. GprMax is based on the Finite-Difference Time-Domain (FDTD) method, which enables accurate simulation of Maxwell's equations in complex heterogeneous environments. It allows the user to control antenna configurations, soil composition, dielectric properties, and water content, thus providing a reliable platform for generating large, labeled datasets under controlled conditions. Using gprMax, a set of A-scans corresponding to different values of soil moisture and permittivity was simulated. These parameters were chosen as ground-truth labels, and the autoencoder was trained so that its two-dimensional latent embedding explicitly represented moisture content and permittivity. This approach allows the neural model to learn not only to reconstruct input signals but also to embed them in a latent space aligned

with physically interpretable variables.

The second study extends this methodology to real-world GPR measurements acquired with different antenna types. Unlike the simulated dataset, where parameters such as permittivity could be precisely defined, real-world data introduce additional challenges, including noise, hardware variability, and environmental heterogeneity. In this context, the two latent dimensions were trained to encode soil moisture and the class of the antenna used during acquisition. By embedding the GPR data in this way, the autoencoder enabled a compact yet informative representation of heterogeneous field measurements. These latent embeddings were then integrated into a Deep Reinforcement Learning (DRL) framework designed for robotic navigation. In this setting, the embedding space provided the robot with actionable information about subsurface conditions, guiding decision-making as the robot navigated through reconstructed maps. This combination of representation learning and DRL illustrates the potential of autoencoder-based embeddings to go beyond classification or regression, serving as foundational building blocks for intelligent robotic systems.

4.1 An overview about GPR

According to the Food and Agriculture Organization, agriculture accounts for 70 of global freshwater withdrawals [128]. The amount of water evaporated compared to the amount of water delivered to the field, has been estimated to be 37 percent, due to inefficient irrigation practices. This would mean 63 of the water delivered to field is lost as runoff and/or drainage [129]. The pursuit of accurate soil moisture estimation has driven the development of various invasive and non-invasive techniques. Invasive methods typically rely on direct soil sampling using probes, followed by laboratory analysis. Although these techniques offer high precision, they are often criticized for being time-consuming, labor-intensive, and disruptive to both soil and plant integrity—making them impractical for large-scale agricultural operations. In contrast, non-invasive approaches, such as satellite imagery and GPR, enable moisture estimation without disturbing the soil, offering a more scalable and plant-friendly alternative.

GPR is a well-established non-destructive testing technique that operates by transmitting electromagnetic waves into the ground and recording the backscattered signals produced by subsurface structures. The resulting A-scans are one-dimensional time series reflecting the dielectric properties and geometry of the investigated medium. Applications of GPR span from geotechnical investigations, archaeology, and civil engineering, to precision agriculture and environmental monitoring. This technique enables the rapid and accurate acquisition of large volumes of data, which can be utilized to evaluate key subsurface properties such as soil moisture and texture.

The moisture content and composition of soil determine its relative permittivity, denoted as ϵ_r ,

which characterizes the soil's ability to transmit electric fields. This parameter plays a critical role in shaping the GPR signal, affecting its delay, amplitude, and attenuation. As a result, variations in soil permittivity directly influence the signal's behavior in both the time and frequency domains [130] [131]. Among the factors influencing GPR-based soil moisture estimation, the permittivity of the transmission channel has the most pronounced effect on the characteristics of the received signal [132].

Effective deployment of machine learning models requires the development of a versatile and, more importantly, accurately labeled GPR training dataset. Proper labeling is essential to prevent issues such as overfitting and underfitting, ensuring that the model performs reliably in real-world applications. However, manually labeling GPR data is a time-consuming and impractical task, particularly at the scale of large agricultural operations. To overcome this limitation, researchers increasingly rely on synthetic data generation through electromagnetic simulation tools such as *gprMax* [133], enabling the creation of realistic, labeled datasets without the need for extensive fieldwork.

H. Namdari et al. [134] use feature engineering techniques to extract meaningful features from the GPR signals, followed by a selection process to identify the most effective machine learning model for soil moisture prediction. J. Zheng et al. [135] have designed two separate CNNs to address the classification and regression tasks independently. Each network consists of two main components: a feature extraction module composed of one-dimensional convolutional layers followed by max-pooling layers, and an output module functioning either as a classifier or a regressor. The classification network predicts discrete moisture levels (quantitative moisture classification), while the regression network outputs a continuous estimate of soil water content. The classification network is trained first, and its learned feature extractor is then transferred to the regression network through a transfer learning approach, allowing for efficient reuse of extracted features and improved training performance.

A similar approach is followed in the present work. The first step of the procedure consists in training an autoencoder with data relative to homogenous soils, simulated by *gprMax* by setting specific values of permittivity and conductivity for which the correspondent content of water is known. The different A-scans constitute the instances of the autoencoder, whose custom loss function allows the network to learn the permittivity and conductivity of each sample and encode them into the latent space, while minimizing the reconstruction error between input and output of the network, in order to avoid overfitting. After training, the network can be used to infer relative permittivity and conductivity of heterogenous soils with fixed content of water, whose correspondent A-scans are simulated in *gprMax*. The ability of the network to generalize from homogenous to heterogeneous and more realistic soils suggests the possibility to integrate this kind of deep learning models into real scenarios, enabling real-time monitoring for addressing the challenges of water management in agriculture.

Table 4.1: Ground truth values of relative permittivity and moisture for the different classes.

θ_v	0.001	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4
ϵ_r	3.04	3.85	5.34	7.45	10.12	13.28	16.89	20.88	25.20

4.2 Creation of synthetic dataset

By using the open source software GprMax, a virtual scenario has been created that represents an antenna (in particular, a GSSI 400 MHz bowtie antenna) moving on the soil. The training and test data sets have then been simulated. The soil used to generate the training data set consists of a homogeneous material box of size $0.3 \times 0.3 \times 0.35$, characterized by conductivity $\sigma = 0.001$, permeability $\mu = 1$, and without magnetic loss. Different instances have been launched, which differ for the value of relative permittivity ϵ_r . These values are derived from the Topp equation [136]

$$\epsilon_r = 3.03 + 9.3\theta_v + 146\theta_v^2 - 76.7\theta_v^3, \quad (4.1)$$

and correspond to different fixed volumetric content of water θ_v . The couples of values that constitute the different classes are summarized in Table 4.1. For each value of relative permittivity 300 A-scans have been simulated, corresponding to 300 different distances of the antenna from the top of the soil, equally spaced in range from 3.9cm to 7.1cm. The single A-scan represents the signal recorded by the receiver of the antenna, which consists of the three components of the electric field and three components of the magnetic field. Since the components of the magnetic field are discarded, the training dataset consists of 9 classes of 300 samples (3-channel A-scans).

The test dataset consists of 3-channel A-scans recorded by the GSSI antenna while it moves horizontally above a heterogenous soil, at a distance of 7cm. The heterogenous soil can be implemented in GprMax by the so called Peplinski soil [132] and the fractal box. Instead of a homogeneous box with defined electromagnetic parameters, a box of the same size has been created, that is characterized by a fractal distribution (with fractal dimension equal to 1.5) of soil composed by 40% of sand, 10% of clay and moisture in the range $[\theta_v - 0.001, \theta_v + 0.001]$, where θ_v takes at each instance one of the value among those reported in Table 4.1, for a total of 9 instances. In every instance the antenna moves along the y-axis covering all the box and recording the signal in 30 equally spaced positions. The test dataset that is generated consists of 9 classes, each one containing 30 samples (3-channel one-dimensional A-scans). Then, the flat surface is replaced by rough surface, with a roughness of 3cm, the simulation is repeated and another dataset is generated. By adding in the model also the grass (see Figure 4.1) a third dataset, structured as the first two, is produced.

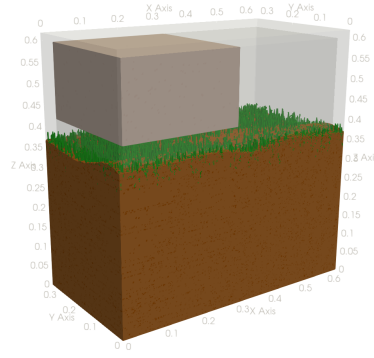


Figure 4.1: Geometry of the model with grass (in green). The box above the soil represents the antenna.

4.3 Modification to the loss function of the AE

- reconstruction loss, that is the MSE between the input and the output of the neural network;
- content loss, that is the MSE between the first two components of the latent array and the 2-dimensional array containing the ground truth parameters of relative permittivity and moisture;
- intermediate loss, that is the MSE between the outputs of the first layers of the encoder and the decoder.

The first term of the loss is the basic loss of an the autoencoder, that can be used for clustering purposes for its ability to encode the input into the latent space of features. Here the content loss allows the autoencoder to work also as a regressor, since it learns an encoding function that maps the samples into specific positions of the two dimensional space spanned by the first two components of the latent array, that represent the predicted moisture and relative permittivity. The normalization of the data is obtained dividing each A-scan by the maximum absolute value of the relative component of the direct wave (the wave that goes from the transmitter to the receiver of the antenna in absence of obstacles that scatters the signal), while the ground truth parameters are normalized in the range $[-\tanh(1), \tanh(1)]$ according to the following:

$$ground_truth \rightarrow \tanh(1) \left(2 \frac{ground_truth - min_truth}{max_truth - min_truth} - 1 \right), \quad (4.2)$$

where min_truth and max_truth are respectively the minimum and maximum values among the ground truth values. This normalization allows the network to properly compare the output of the activation function of the dense layer of the encoder, the hyperbolic tangent, and the ground truth. Finally, all the layers are initialized with Glorot uniform and Adamax is used as optimizer.

Table 4.2: Mean values of predicted moisture and mean absolute deviation from ground truth for the different classes, rounded to the second decimal digit.

θ_v	0	0.05	0.1	0.15	0.2	0.26	0.31	0.35	0.4
mae	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

(a) Flat surface without grass.

θ_v	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
mae	0.00	0.00	0.00	0.00	0.01	0.10	0.01	0.01	0.01

(b) Rough surface without grass.

θ_v	0.00	0.05	0.10	0.15	0.20	0.26	0.31	0.35	0.40
mae	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01

(c) Rough surface with grass.

4.4 Outcomes of the autoencoder trained with simulated data

The described procedure of training the network with samples relative to homogeneous and testing it on samples that come from heterogeneous soil slightly depends on the initialization of the layers. Figure 4.5 shows how the error of the predictions spreads around the ground truth, for each class, after that 30 instances of the model have been trained. The distribution suggests the possibility to correct the predictions with the mean value of the error relative to each class. The correction on the value of the moisture is smaller than 0.05, which is the distance between the value of moisture of two consecutive classes.

The results obtained from an instance of the model that was not used to evaluate the aforementioned correction are summarized in Table 4.2, while Figures 4.2, 4.3, 4.4 shows the encoding of test samples in the moisture-permittivity plane. The magnitude of the errors showed in 4.2 is coherent with the spread of error around the ground truth, represented by the size of boxes in Fig. 4.5. One can see that in the scenario with flat heterogeneous surface without grass all the samples in one class are encoded in the same point. This is due to the fact that the flat surface and the absence of grass make the A-scans of a given class very similar to each other. By the way, in all the scenarios the proposed network is able to generalize from homogeneous to heterogeneous soils, even if some corrections are needed to give better predictions. When the MSE loss term that forces the first two components of the latent array to match the ground truth values of relative permittivity and volumetric water content is removed, and only the reconstruction loss between the input and output (as in a standard autoencoder) is retained, both the error magnitude and variability across different network initializations increase significantly. This indicates that the approach described — using latent space representations to directly capture

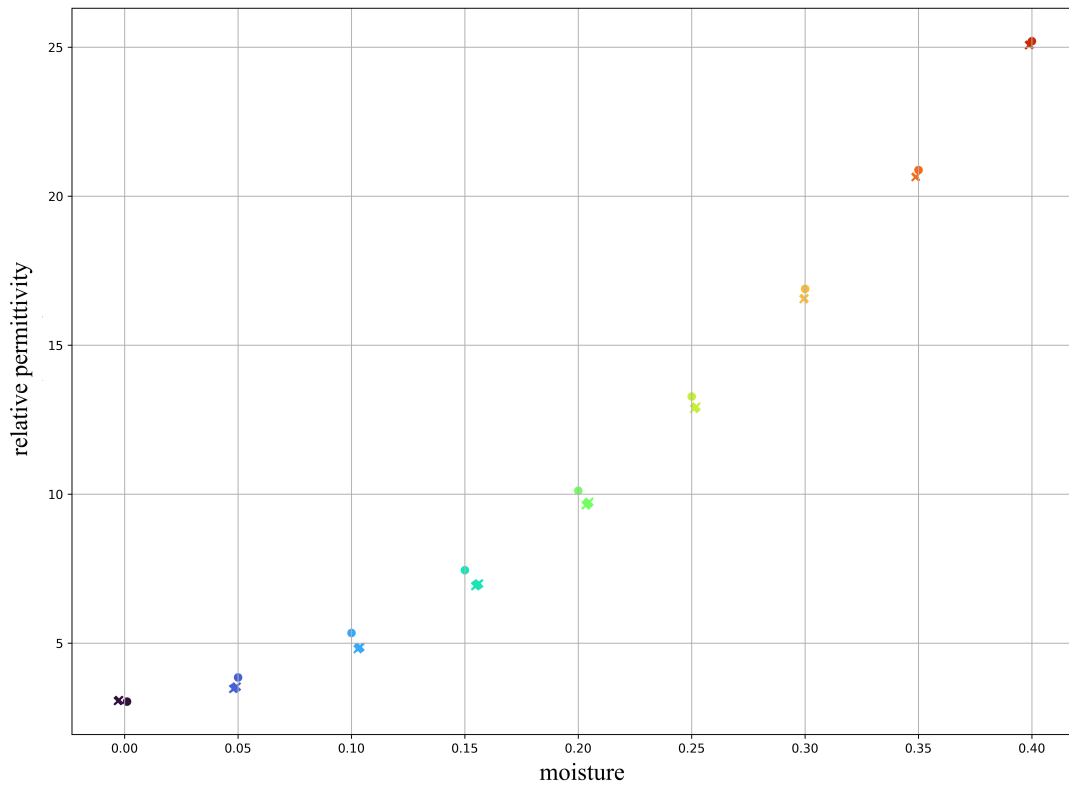


Figure 4.2: Predicted values of moisture (x-axis) and relative permittivity (y-axis) in the scenario with flat surface.

physically meaningful parameters — only works reliably when the loss function includes the supervised term linking the latent space to permittivity and water content. Without this supervised constraint, the network fails to consistently organize its latent space in a physically interpretable way, making the approach ineffective. This highlights the importance of explicitly encoding physical knowledge in the training objective.

4.5 Outcomes of the autoencoder trained with real data

The ability of the network to generalize from homogeneous to heterogeneous and more realistic soils suggests the possibility to integrate this kind of deep learning models into real scenarios, enabling real-time monitoring for addressing the challenges of water management in agriculture. To this end, different types of soils have been analyzed, which varied in terms of water content and composition, ranging from purely sandy soils to agricultural soils and flowerbeds. The sandy soil was obtained by filling a container with sandstone up to a fixed level and adding water to adjust the moisture content. Static measurements were carried out using two antennas: MALA 1.6 GHz and MALA 2.3 GHz shielded antenna.

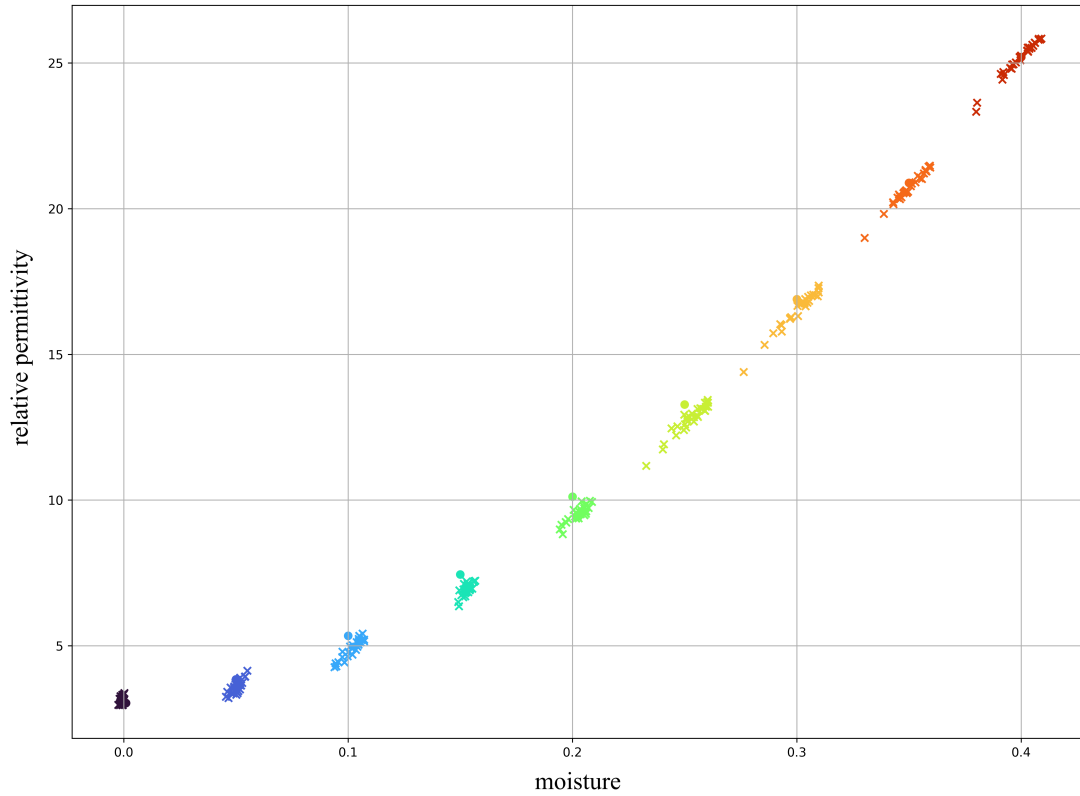


Figure 4.3: Predicted values of moisture (x-axis) and relative permittivity (y-axis) in the scenario with rough surface without grass.

Measurements of the non-sandy soils, instead, were performed with three different types of antennas: MALA 800 MHz, MALA 500 MHz shielded antenna, and Proceq GP8800. The water content was obtained by collecting a sample from each analyzed area and measuring its weight before and after drying in a laboratory oven. The measurements consist of one-dimensional signals, the A-scans, which represent the signal recorded by the antenna at the point where it is placed. The collection of A-scans, cropped to the length of the shortest recorded A-scan, constitutes the training and validation dataset of the selected neural network model, i.e., an autoencoder. Specifically, the dataset is composed of 5 classes, corresponding to the number of antennas used, each is further divided into subclasses corresponding to the different water-content levels collected. To balance the dataset, all subclasses consist of the same number of samples. The overall structure of the autoencoder is shown in Fig. 4.6. The optimizer used during training was Adamax with a learning rate set to 0.01 for 1/9 of the steps (200 epochs with a batch size of 69), 0.1 for half of the remaining steps, and again 0.01 for the final steps. The dataset was also normalized dividing all the A-scans by the maximum value of the dataset and randomly divided into 70% training and 30% test. Fig. 4.7 shows the encoding of all samples in a plane, with the x-axis representing the water content and the y-axis representing the class. The predicted mean values for the test samples and root mean squared error for each subclass are also reported in

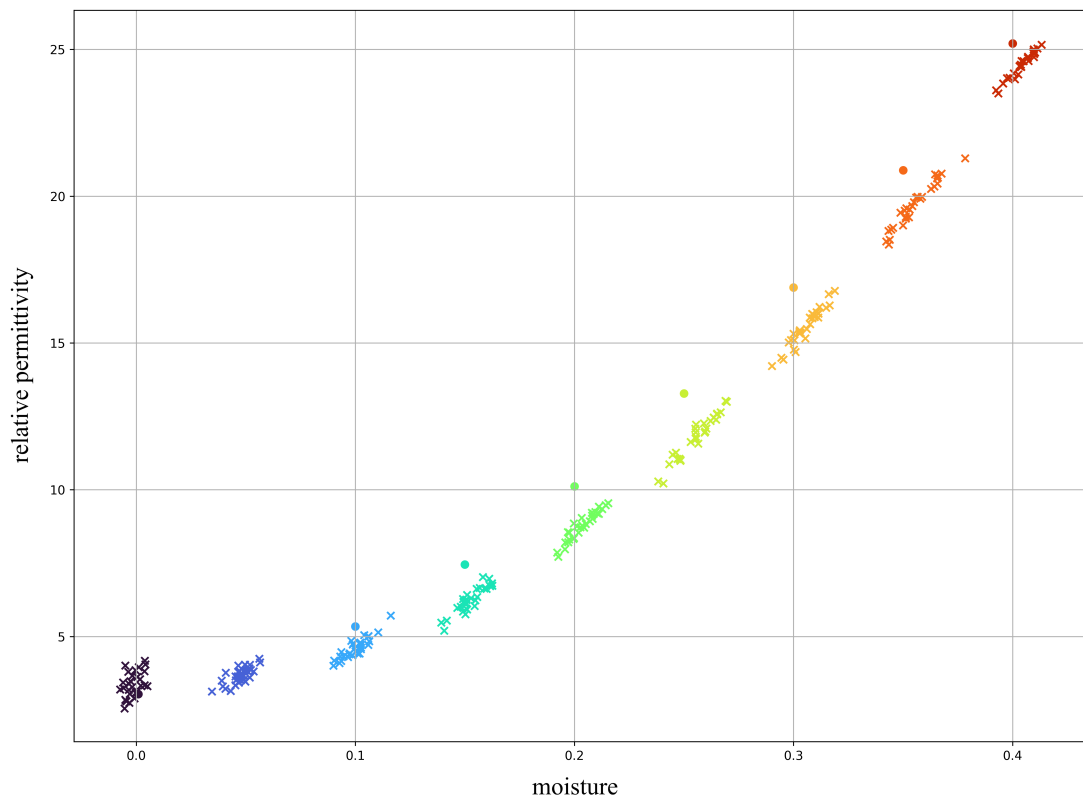
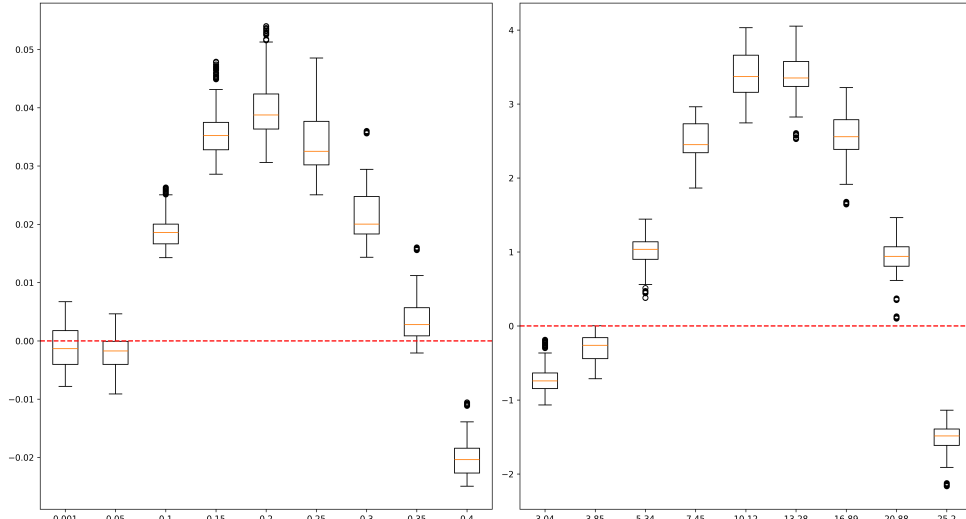
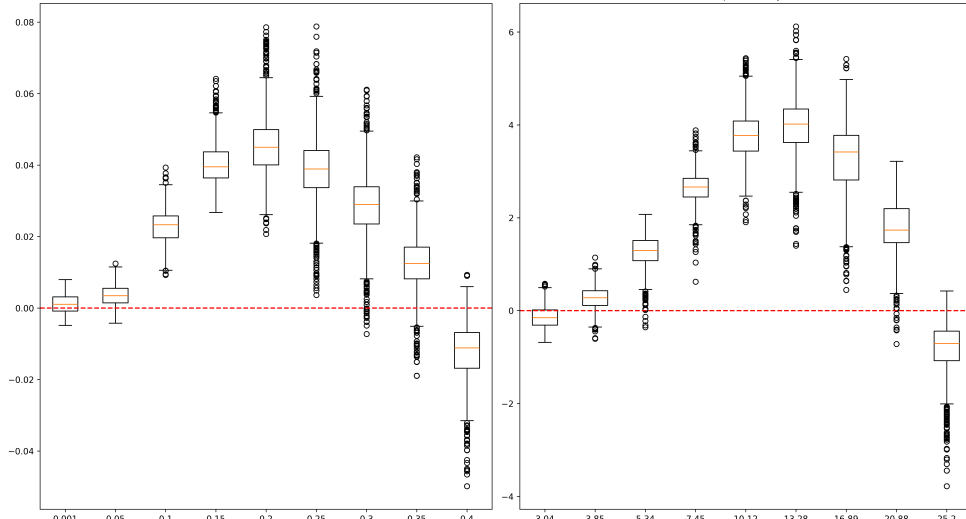


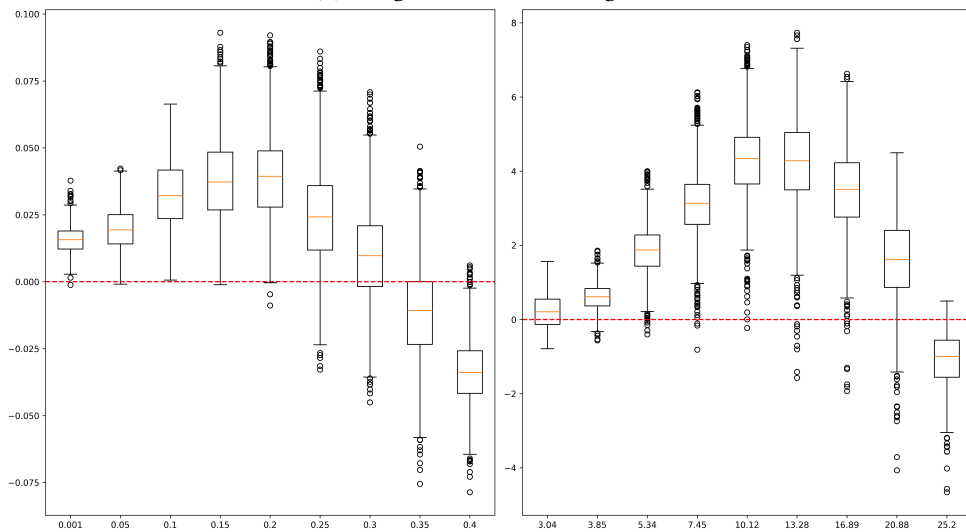
Figure 4.4: Predicted values of moisture (x-axis) and relative permittivity (y-axis) in the scenario with rough surface and grass.



(a) Flat surface without grass.



(b) Rough surface without grass.



(c) Rough surface with grass.

Figure 4.5: Boxplots of the prediction error of each class for moisture, on the left, and for relative permittivity, to the right. The horizontal dotted line represent the zero.

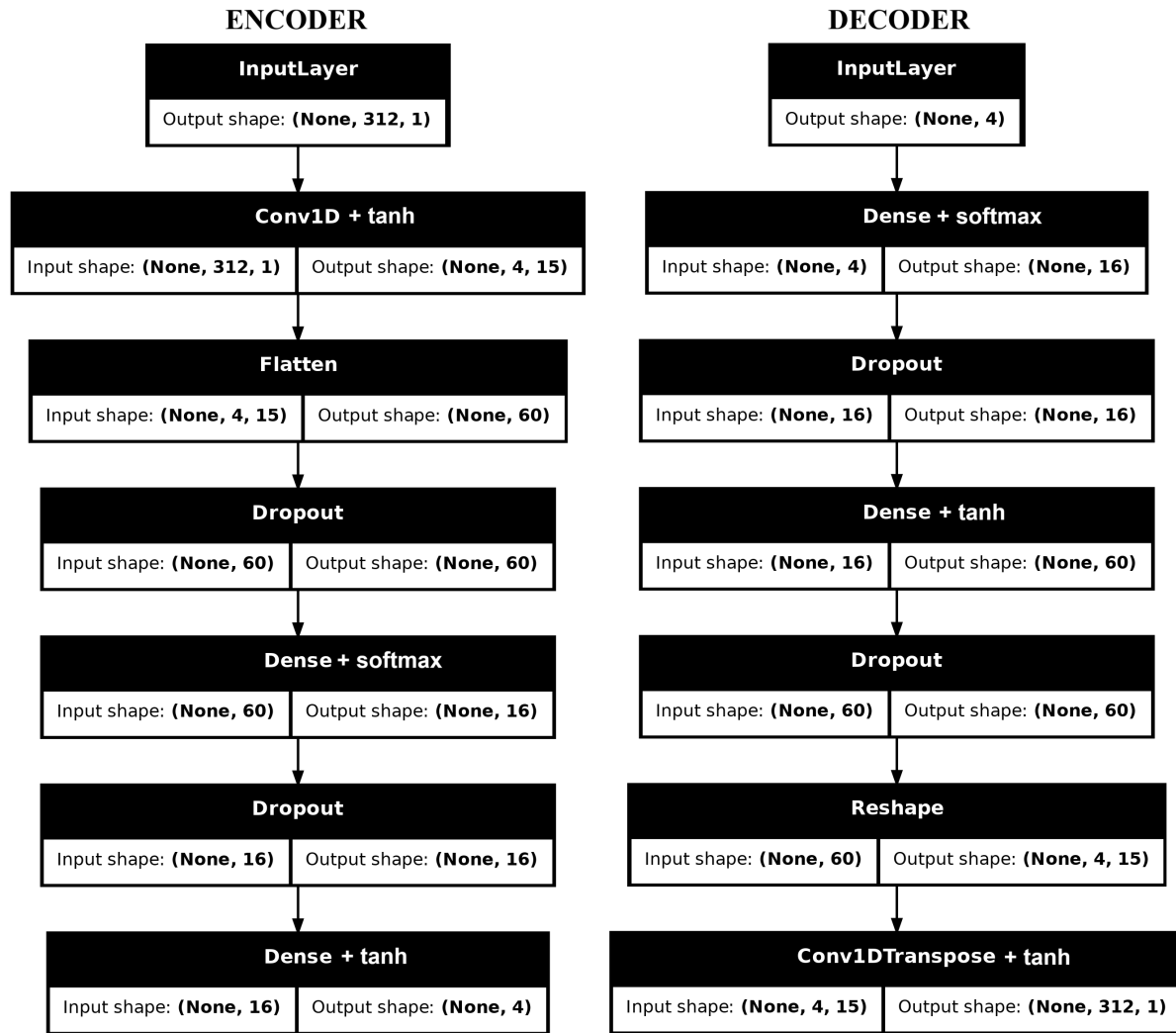


Figure 4.6: Overall structure of the autoencoder used to predict moisture of real data.

Table 4.3. It is also possible to compare the true class labels with the predicted ones by rounding to the nearest integer neighbor the y-component of the outputs of the network. Results show bigger errors for samples measured with antenna of class 2, probably because the acquisition conditions of that particular antenna caused higher intra-class variability along with lower inter-class variability.

4.6 DRL for robot navigation

Two state-of-the-art DRL algorithms have been adopted: *Proximal Policy Optimization* (PPO) [137] and *Soft Actor-Critic* (SAC) [138]. In this application, PPO is used for the discrete-action framework developed, while SAC is used for the continuous-action framework. Both are policy gradient methods, but differ in their objective functions and exploration strategies. In particular, a moisture-avoidance policy has been adopted: agents learn to prefer drier terrain while reaching target locations. The frameworks

are adaptable to alternative objectives such as moisture-seeking for irrigation. The discrete reinforcement learning framework models both the state and action spaces as finite and enumerated sets. The environment is represented as a 2D latent grid derived from autoencoder embeddings of GPR data. The agent occupies a grid cell and selects among four actions (*up, down, left, right*). To increase variability and ensure meaningful performance assessment, some latent grids are enhanced with additional dry paths, as well as sparse dry and wet zones. The reward function integrates multiple components to guide the agent toward efficient and safe navigation while accounting for both soil moisture conditions and goal-reaching performance. Its design jointly promotes navigation through drier regions, consistent progress toward the target, and avoidance of undesirable behaviors such as looping, boundary violations, or stagnation. The continuous reinforcement learning setup replaces the grid with a latent plane augmented by multi-octave Perlin noise [139]. This design brings the formulation closer to realistic navigation scenarios while preserving localized, goal-relevant perception under partial observability settings. A continuous observation mask centered on the agent is sampled at each timestep directly from the continuous latent plane, enabling online adaptation to evolving field conditions and ensuring that navigation policies remain valid even when the environment changes over time. The action space consists of real-valued displacements $(\Delta x, \Delta y)$, clipped to reflect physical actuation limits [140].

4.7 Summary

The research presented in this chapter was centered on the development and application of autoencoder-based neural architectures for the interpretation of GPR signals, specifically A-scans, and for their subsequent integration in autonomous navigation tasks. A mapping of each A-scan into a two-dimensional latent representation has been achieved by explicitly using ground-truth information. By training the autoencoder in a supervised fashion, the latent variables are constrained to represent physically meaningful parameters, thus combining dimensionality reduction with interpretability. This chapter demonstrates the effectiveness of leveraging synthetic GPR data and autoencoder-based architectures to extract physically meaningful features such as soil permittivity and volumetric water content. By constraining the first two components of the latent space to match these ground truth parameters through a dedicated MSE loss term, a stable and interpretable representation with minimal sensitivity to network initialization is achieved. Removing this supervised constraint leads to a significant increase in reconstruction error and variability, indicating a collapse in the physical interpretability of the latent space. These findings take advantage of geophysical data to show how deep learning models usually used for unsupervised feature learning [3] can be also bent to embed domain knowledge directly into the training objective. The two works presented here underscore the versatility of

autoencoders in addressing different problem domains within subsurface sensing. By supervising the latent space with ground truth parameters, the autoencoder achieves a dual objective: it compresses high-dimensional GPR signals into a compact embedding, while ensuring that this embedding encodes meaningful physical or contextual properties. A particularly important aspect of this approach is its potential for generalization. Since the embeddings represent continuous variables (e.g., permittivity, water content), the trained models can, in principle, interpolate between conditions encountered during training. This suggests that the methodology may be extended to cases not explicitly included in the training dataset, effectively performing regression or interpolation in the latent space. Such capability is especially valuable for NDT applications, where exhaustive data collection across all possible conditions is often impractical. The integration of autoencoder embeddings into robotic navigation further highlights the relevance of this methodology for Industry 4.0 and beyond. As robotics and AI become increasingly intertwined, the ability to compress raw sensor data into compact, interpretable, and actionable features is crucial for autonomous decision-making. The supervised autoencoder framework demonstrated here provides a path forward, enabling robots not only to perceive subsurface conditions but also to adapt navigation strategies accordingly. In summary, the two studies presented in this chapter demonstrate the effectiveness of supervised autoencoders for embedding GPR A-scans. The first application focuses on simulated data to estimate soil moisture and permittivity, while the second leverages real data and multiple antenna types to support DRL-based navigation. Together, they show how autoencoder-based embeddings can serve as a bridge between raw sensing data, interpretable physical parameters, and intelligent robotic decision-making.

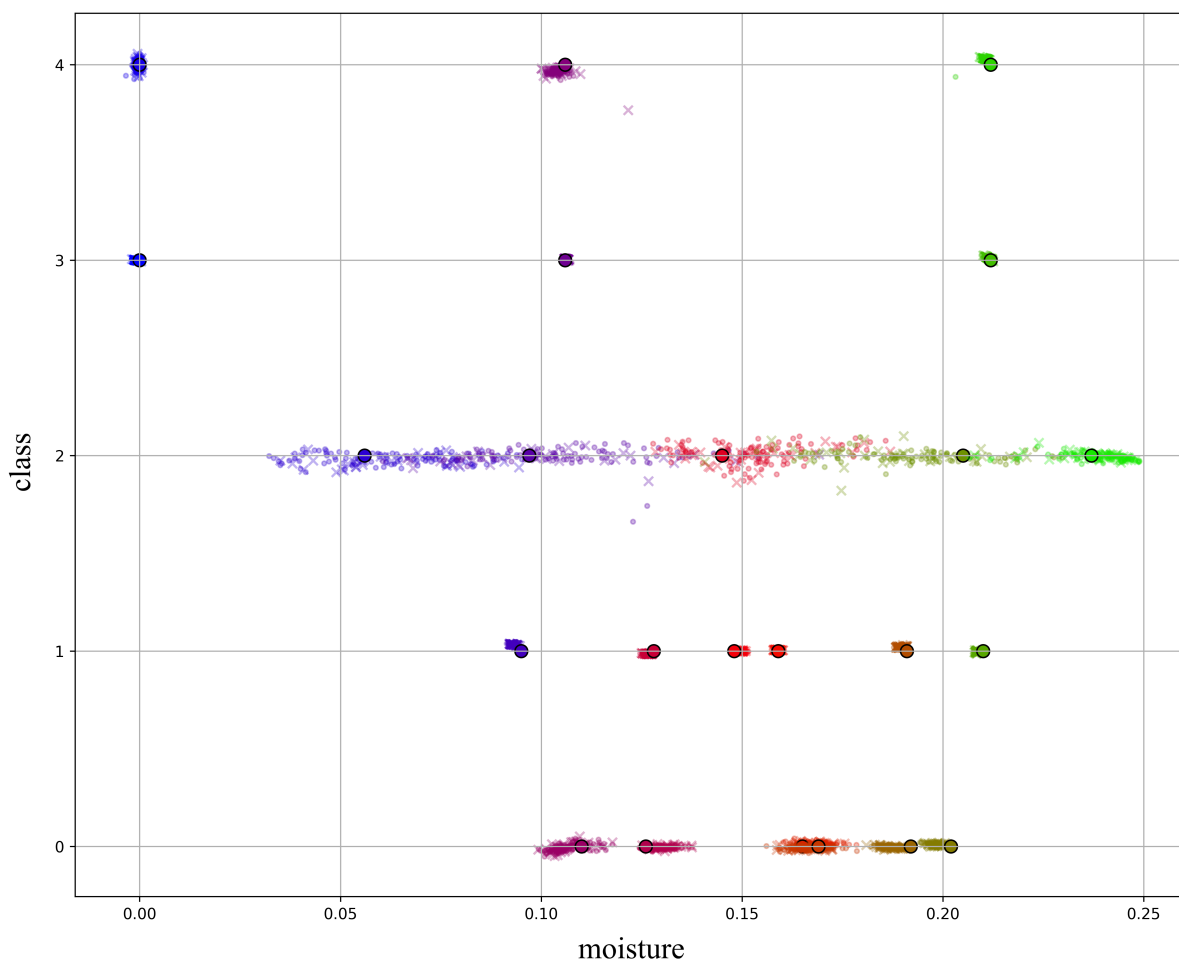


Figure 4.7: Predicted values of moisture (x-axis) and class of the antenna (y-axis).

Table 4.3: Results relative to the real scenario. The first two columns represent the true labels, the third column the mean predicted moisture, the fourth column the root mean square error of predictions.

moisture_true	class_true	mean	rmse
0.110	0	0.1075	0.0044
0.126	0	0.1317	0.0063
0.165	0	0.1690	0.0052
0.169	0	0.1692	0.0042
0.192	0	0.1897	0.0035
0.202	0	0.1999	0.0025
0.095	1	0.0945	0.0015
0.128	1	0.1273	0.0013
0.148	1	0.1487	0.0011
0.159	1	0.1598	0.0012
0.191	1	0.1900	0.0016
0.210	1	0.2106	0.0018
0.056	2	0.0619	0.0177
0.097	2	0.1030	0.0265
0.145	2	0.1454	0.0133
0.205	2	0.1890	0.0260
0.237	2	0.2337	0.0112
0.000	3	0.0032	0.0092
0.106	3	0.1059	0.0004
0.212	3	0.2119	0.0009
0.000	4	0.0002	0.0010
0.106	4	0.1071	0.0032
0.212	4	0.2117	0.0008

Chapter 5

Conclusions

This thesis has explored the use of deep learning—and specifically autoencoder-based architectures—for anomaly detection across multiple industrial and sensing contexts. The work was structured into four main chapters, each addressing a distinct challenge. The first chapter provided a comprehensive systematic review of anomaly detection in the context of Industry 4.0, outlining the main application domains, the sensing technologies employed, and the most widely adopted deep learning models. This analysis highlighted the growing role of intelligent algorithms in managing heterogeneous data sources and the increasing integration of machine learning methods into industrial and robotic systems. The second chapter focused on the detection of defective welds acquired through 3D profilometry. In this case, a one-class autoencoder was employed to model the normal behavior of welding profiles, identifying anomalies through deviations in reconstruction error. The results confirmed the capability of the network to learn representative latent features of non-defective welds, offering a robust and data-driven alternative to traditional inspection techniques. The third chapter extended the anomaly detection framework to composite materials analyzed using Laser Ultrasound Technology. An unsupervised approach was adopted, allowing the autoencoder to autonomously discover latent representations that distinguish healthy regions from defective ones. The findings emphasized the adaptability of autoencoders to different sensing modalities and their potential for integration into non-destructive testing workflows. In the fourth and final chapter, a supervised autoencoder was introduced to estimate soil water content from Ground Penetrating Radar data, both simulated and acquired experimentally. In this case, domain knowledge was incorporated into the loss function, enabling the latent space to explicitly encode relevant physical quantities such as water content and permittivity. This approach demonstrated that autoencoders can be effectively guided by domain-specific constraints, improving interpretability and regression performance. Despite the promising results, several critical aspects remain open. The generalization ability of the proposed models should be tested on larger and more

diverse datasets, potentially acquired in real-time and under varying environmental conditions. Moreover, future work should explore hybrid frameworks combining autoencoders with clustering or reinforcement learning strategies, to enable real-time fully autonomous systems capable of continuous monitoring and adaptive decision-making. Ultimately, this research demonstrates the versatility of autoencoders as a unifying framework for anomaly detection across different modalities and application domains—from materials inspection to environmental sensing—paving the way toward intelligent, self-aware, and adaptive industrial and robotic systems.

Bibliography

- [1] A. Liso, A. Cardellicchio, C. Patruno, M. Nitti, P. Ardino, E. Stella, and V. Renò, “A review of deep learning-based anomaly detection strategies in industry 4.0 focused on application fields, sensing equipment, and algorithms,” *IEEE Access*, vol. 12, pp. 93 911–93 923, 2024.
- [2] A. Liso, A. Cardellicchio, C. Patruno, M. Nitti, E. Stella, and V. Renò, “Awandt: assessing welding anomalies via non-destructive tests,” in *Multimodal Sensing and Artificial Intelligence: Technologies and Applications III*, vol. 12621. SPIE, 2023, pp. 66–74.
- [3] A. Liso, C. Patruno, A. Cardellicchio, P. Ardino, N. Gallo, G. del Prete, V. Dentico, V. Vespini, S. Coppola, P. Ferraro, and V. Renò, “A deep learning-based probabilistic approach for non-destructive testing of aircraft components using laser ultrasonic data,” *IEEE Access*, vol. 13, pp. 66 761–66 773, 2025.
- [4] A. Liso, N. R. Calderón, V. Perez-Gracia, T. R. D’Orazio, and V. Renò, “Detecting the content of water in agricultural soils by GPR data and autoencoders,” in *Multimodal Sensing and Artificial Intelligence for Sustainable Future*, F. Soldovieri, P. Picart, V. Bianco, and C. Falldorf, Eds., vol. 13570, International Society for Optics and Photonics. SPIE, 2025, p. 135700F. [Online]. Available: <https://doi.org/10.1117/12.3062608>
- [5] T. Anand, S. Sinha, M. Mandal, V. Chamola, and F. R. Yu, “AgriSegNet: Deep Aerial Semantic Segmentation Framework for IoT-Assisted Precision Agriculture,” *IEEE Sensors Journal*, vol. 21, no. 16, pp. 17 581–17 590, Aug. 2021, conference Name: IEEE Sensors Journal. [Online]. Available: <https://ieeexplore.ieee.org/document/9395478>
- [6] E. Mujkic, M. P. Philipsen, T. B. Moeslund, M. P. Christiansen, and O. Ravn, “Anomaly Detection for Agricultural Vehicles Using Autoencoders,” *Sensors*, vol. 22, no. 10, p. 3608, Jan. 2022, number: 10 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/22/10/3608>

- [7] M. Nawaz, T. Nazir, A. Javed, M. Masood, J. Rashid, J. Kim, and A. Hussain, "A robust deep learning approach for tomato plant leaf disease localization and classification," *Scientific Reports*, vol. 12, no. 1, p. 18568, Nov. 2022, number: 1 Publisher: Nature Publishing Group. [Online]. Available: <https://www.nature.com/articles/s41598-022-21498-5>
- [8] H. Zhao, H. Liu, W. Hu, and X. Yan, "Anomaly detection and fault analysis of wind turbine components based on deep learning network," *Renewable Energy*, vol. 127, pp. 825–834, Nov. 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0960148118305457>
- [9] A. Kisvari, Z. Lin, and X. Liu, "Wind power forecasting – A data-driven method along with gated recurrent neural network," *Renewable Energy*, vol. 163, pp. 1895–1909, Jan. 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0960148120316918>
- [10] I. Ullah, R. U. Khan, F. Yang, and L. Wuttisittikulkij, "Deep Learning Image-Based Defect Detection in High Voltage Electrical Equipment," *Energies*, vol. 13, no. 2, p. 392, Jan. 2020, number: 2 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1996-1073/13/2/392>
- [11] N. Renström, P. Bangalore, and E. Highcock, "System-wide anomaly detection in wind turbines using deep autoencoders," *Renewable Energy*, vol. 157, pp. 647–659, Sep. 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0960148120306856>
- [12] R. Pierdicca, M. Paolanti, A. Felicetti, F. Piccinini, and P. Zingaretti, "Automatic Faults Detection of Photovoltaic Farms: solAIr, a Deep Learning-Based System for Thermal Images," *Energies*, vol. 13, no. 24, p. 6496, Jan. 2020, number: 24 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1996-1073/13/24/6496>
- [13] M. Alrifayy, W. H. Lim, and C. K. Ang, "A Novel Deep Learning Framework Based RNN-SAE for Fault Detection of Electrical Gas Generator," *IEEE Access*, vol. 9, pp. 21 433–21 442, 2021, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9337852>
- [14] P. Trizoglou, X. Liu, and Z. Lin, "Fault detection by an ensemble framework of Extreme Gradient Boosting (XGBoost) in the operation of offshore wind turbines," *Renewable Energy*, vol. 179, pp. 945–962, Dec. 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0960148121010909>

- [15] Y. Guo, Z. Pang, J. Du, F. Jiang, and Q. Hu, "An Improved AlexNet for Power Edge Transmission Line Anomaly Detection," *IEEE Access*, vol. 8, pp. 97 830–97 838, 2020, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9097293>
- [16] M. Bai, J. Liu, Y. Ma, X. Zhao, Z. Long, and D. Yu, "Long Short-Term Memory Network-Based Normal Pattern Group for Fault Detection of Three-Shaft Marine Gas Turbine," *Energies*, vol. 14, no. 1, p. 13, Jan. 2021, number: 1 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1996-1073/14/1/13>
- [17] H. Wang, X. Liu, L. Ma, and Y. Zhang, "Anomaly detection for hydropower turbine unit based on variational modal decomposition and deep autoencoder," *Energy Reports*, vol. 7, pp. 938–946, Nov. 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2352484721009951>
- [18] Y. Zhu, C. Zhu, J. Tan, Y. Tan, and L. Rao, "Anomaly detection and condition monitoring of wind turbine gearbox based on LSTM-FS and transfer learning," *Renewable Energy*, vol. 189, pp. 90–103, Apr. 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0960148122002038>
- [19] M. Vlamincx, R. Heidebuchel, W. Philips, and H. Luong, "Region-Based CNN for Anomaly Detection in PV Power Plants Using Aerial Imagery," *Sensors*, vol. 22, no. 3, p. 1244, Jan. 2022, number: 3 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/22/3/1244>
- [20] M. F. Ahmed, J. C. Mohanta, and A. Sanyal, "Inspection and identification of transmission line insulator breakdown based on deep learning using aerial images," *Electric Power Systems Research*, vol. 211, p. 108199, Oct. 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0378779622004084>
- [21] O. Attallah, R. A. Ibrahim, and N. E. Zakzouk, "CAD system for inter-turn fault diagnosis of offshore wind turbines via multi-CNNs & feature selection," *Renewable Energy*, vol. 203, pp. 870–880, Feb. 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0960148122018493>
- [22] Y. Zhang, G. Suzuki, and H. Shioya, "Prediction and Detection of Sewage Treatment Process Using N-BEATS Autoencoder Network," *IEEE Access*, vol. 10, pp. 112 594–112 608, 2022, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9928291>
- [23] H. Pan, Z. Yin, and X. Jiang, "High-Dimensional Energy Consumption Anomaly Detection: A Deep Learning-Based Method for Detecting Anomalies," *Energies*, vol. 15, no. 17, p. 6139, Jan.

- 2022, number: 17 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1996-1073/15/17/6139>
- [24] T. Liu, G. Li, and Y. Gao, "Fault diagnosis method of substation equipment based on You Only Look Once algorithm and infrared imaging," *Energy Reports*, vol. 8, pp. 171–180, Oct. 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2352484722009234>
- [25] L. Scime and J. Beuth, "A multi-scale convolutional neural network for autonomous anomaly detection and classification in a laser powder bed fusion additive manufacturing process," *Additive Manufacturing*, vol. 24, pp. 273–286, Dec. 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2214860418305165>
- [26] M. Canizo, I. Triguero, A. Conde, and E. Onieva, "Multi-head CNN-RNN for multi-time series anomaly detection: An industrial case study," *Neurocomputing*, vol. 363, pp. 246–260, Oct. 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0925231219309877>
- [27] A. L. Alfeo, M. G. C. A. Cimino, G. Manco, E. Ritacco, and G. Vaglini, "Using an autoencoder in the design of an anomaly detector for smart manufacturing," *Pattern Recognition Letters*, vol. 136, pp. 272–278, Aug. 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167865520302269>
- [28] A. P. Garland, B. C. White, B. H. Jared, M. Heiden, E. Donahue, and B. L. Boyce, "Deep Convolutional Neural Networks as a Rapid Screening Tool for Complex Additively Manufactured Structures," *Additive Manufacturing*, vol. 35, p. 101217, Oct. 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2214860420305893>
- [29] H. Liang, L. Song, J. Wang, L. Guo, X. Li, and J. Liang, "Robust unsupervised anomaly detection via multi-time scale DCGANs with forgetting mechanism for industrial multivariate time series," *Neurocomputing*, vol. 423, pp. 444–462, Jan. 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0925231220316970>
- [30] S. Mao, D. Rajan, and L. T. Chia, "Deep residual pooling network for texture recognition," *Pattern Recognition*, vol. 112, p. 107817, Apr. 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0031320321000042>
- [31] A. L. Ellefsen, S. Ushakov, V. Æsøy, and H. Zhang, "Validation of Data-Driven Labeling Approaches Using a Novel Deep Network Structure for Remaining Useful Life Predictions," *IEEE Access*, vol. 7, pp. 71 563–71 575, 2019, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/8727535>

- [32] W. Mao, L. Ding, Y. Liu, S. S. Afshari, and X. Liang, "A new deep domain adaptation method with joint adversarial training for online detection of bearing early fault," *ISA Transactions*, vol. 122, pp. 444–458, Mar. 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0019057821002238>
- [33] S. Fathizadan, F. Ju, and Y. Lu, "Deep representation learning for process variation management in laser powder bed fusion," *Additive Manufacturing*, vol. 42, p. 101961, Jun. 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2214860421001263>
- [34] B. Lu, D. Xu, and B. Huang, "Deep-learning-based anomaly detection for lace defect inspection employing videos in production line," *Advanced Engineering Informatics*, vol. 51, p. 101471, Jan. 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1474034621002214>
- [35] S. M. H. Rizvi, S. K. Sadanandan, and A. K. Srivastava, "Data-Driven Short-Term Voltage Stability Assessment Using Convolutional Neural Networks Considering Data Anomalies and Localization," *IEEE Access*, vol. 9, pp. 128 345–128 358, 2021, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9521499>
- [36] J. Ye and N. Toyama, "Benchmarking Deep Learning Models for Automatic Ultrasonic Imaging Inspection," *IEEE Access*, vol. 9, pp. 36 986–36 994, 2021, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9366507>
- [37] J. Jakubowski, P. Stanisz, S. Bobek, and G. J. Nalepa, "Anomaly Detection in Asset Degradation Process Using Variational Autoencoder and Explanations," *Sensors*, vol. 22, no. 1, p. 291, Jan. 2022, number: 1 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/22/1/291>
- [38] F. Arellano-Espitia, M. Delgado-Prieto, A.-D. Gonzalez-Abreu, J. J. Saucedo-Dorantes, and R. A. Osornio-Rios, "Deep-Compact-Clustering Based Anomaly Detection Applied to Electromechanical Industrial Systems," *Sensors*, vol. 21, no. 17, p. 5830, Jan. 2021, number: 17 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/21/17/5830>
- [39] S. Ho, W. Zhang, W. Young, M. Buchholz, S. A. Jufout, K. Dajani, L. Bian, and M. Mozumdar, "DLAM: Deep Learning Based Real-Time Porosity Prediction for Additive Manufacturing Using Thermal Images of the Melt Pool," *IEEE Access*, vol. 9, pp. 115 100–115 114, 2021, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9514863>

- [40] D. Fährmann, N. Damer, F. Kirchbuchner, and A. Kuijper, “Lightweight Long Short-Term Memory Variational Auto-Encoder for Multivariate Time Series Anomaly Detection in Industrial Control Systems,” *Sensors*, vol. 22, no. 8, p. 2886, Jan. 2022, number: 8 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/22/8/2886>
- [41] C. Wang and J. Liu, “An Efficient Anomaly Detection for High-Speed Train Braking System Using Broad Learning System,” *IEEE Access*, vol. 9, pp. 63 825–63 832, 2021, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9410630>
- [42] S. Abbasi, M. Famouri, M. J. Shafiee, and A. Wong, “OutlierNets: Highly Compact Deep Autoencoder Network Architectures for On-Device Acoustic Anomaly Detection,” *Sensors*, vol. 21, no. 14, p. 4805, Jan. 2021, number: 14 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/21/14/4805>
- [43] T. Tran, N. T. Pham, and J. Lundgren, “A deep learning approach for detecting drill bit failures from a small sound dataset,” *Scientific Reports*, vol. 12, no. 1, p. 9623, Jun. 2022, number: 1 Publisher: Nature Publishing Group. [Online]. Available: <https://www.nature.com/articles/s41598-022-13237-7>
- [44] M. Rusanovsky, O. Beerli, and G. Oren, “An end-to-end computer vision methodology for quantitative metallography,” *Scientific Reports*, vol. 12, no. 1, p. 4776, Mar. 2022, number: 1 Publisher: Nature Publishing Group. [Online]. Available: <https://www.nature.com/articles/s41598-022-08651-w>
- [45] H. Yan, Z. Liu, J. Chen, Y. Feng, and J. Wang, “Memory-augmented skip-connected autoencoder for unsupervised anomaly detection of rocket engines with multi-source fusion,” *ISA Transactions*, vol. 133, pp. 53–65, Feb. 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0019057822003688>
- [46] H. Kim, H. Lee, and S. W. Kim, “Current Only-Based Fault Diagnosis Method for Industrial Robot Control Cables,” *Sensors*, vol. 22, no. 5, p. 1917, Jan. 2022, number: 5 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/22/5/1917>
- [47] A.-E. R. Abd-Elhay, W. A. Murtada, and M. I. Youssef, “A Reliable Deep Learning Approach for Time-Varying Faults Identification: Spacecraft Reaction Wheel Case Study,” *IEEE Access*, vol. 10, pp. 75 495–75 512, 2022, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9830739>

- [48] A. Almudévar, P. Sevillano, L. Vicente, J. Preciado-Garbayo, and A. Ortega, “Unsupervised Anomaly Detection Applied to Φ -OTDR,” *Sensors*, vol. 22, no. 17, p. 6515, Jan. 2022, number: 17 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/22/17/6515>
- [49] M.-A. Tnani, P. Subarnaduti, and K. Diepold, “Efficient Feature Learning Approach for Raw Industrial Vibration Data Using Two-Stage Learning Framework,” *Sensors*, vol. 22, no. 13, p. 4813, Jan. 2022, number: 13 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/22/13/4813>
- [50] A. Albanese, M. Nardello, G. Fiacco, and D. Brunelli, “Tiny Machine Learning for High Accuracy Product Quality Inspection,” *IEEE Sensors Journal*, vol. 23, no. 2, pp. 1575–1583, Jan. 2023, conference Name: IEEE Sensors Journal. [Online]. Available: <https://ieeexplore.ieee.org/document/9969601>
- [51] L. Kou, J. Chen, Y. Qin, and W. Mao, “The Robust Multi-Scale Deep-SVDD Model for Anomaly Online Detection of Rolling Bearings,” *Sensors*, vol. 22, no. 15, p. 5681, Jan. 2022, number: 15 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/22/15/5681>
- [52] K.-J. Jeong, J.-D. Park, K. Hwang, S.-L. Kim, and W.-Y. Shin, “Two-Stage Deep Anomaly Detection With Heterogeneous Time Series Data,” *IEEE Access*, vol. 10, pp. 13 704–13 714, 2022, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9695481>
- [53] C. Qu, Z. Zhou, Z. Liu, and S. Jia, “Predictive anomaly detection for marine diesel engine based on echo state network and autoencoder,” *Energy Reports*, vol. 8, pp. 998–1003, Jul. 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2352484722002256>
- [54] S. Lee, S. Lee, and B. C. Song, “CFA: Coupled-Hypersphere-Based Feature Adaptation for Target-Oriented Anomaly Localization,” *IEEE Access*, vol. 10, pp. 78 446–78 454, 2022, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9839549>
- [55] B. Kim, M. A. Alawami, E. Kim, S. Oh, J. Park, and H. Kim, “A Comparative Study of Time Series Anomaly Detection Models for Industrial Control Systems,” *Sensors*, vol. 23, no. 3, p. 1310, Jan. 2023, number: 3 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/23/3/1310>
- [56] D. Candido de Oliveira, B. T. Nassu, and M. A. Wehrmeister, “Image-Based Detection of Modifications in Assembled PCBs with Deep Convolutional Autoencoders,” *Sensors*, vol. 23,

- no. 3, p. 1353, Jan. 2023, number: 3 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/23/3/1353>
- [57] J. K. Chow, Z. Su, J. Wu, P. S. Tan, X. Mao, and Y. H. Wang, "Anomaly detection of defects on concrete structures with the convolutional autoencoder," *Advanced Engineering Informatics*, vol. 45, p. 101105, Aug. 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1474034620300744>
- [58] M. Mohtasham Khani, S. Vahidnia, L. Ghasemzadeh, Y. E. Ozturk, M. Yuvalaklioglu, S. Akin, and N. K. Ure, "Deep-learning-based crack detection with applications for the structural health monitoring of gas turbines," *Structural Health Monitoring*, vol. 19, no. 5, pp. 1440–1452, Sep. 2020, publisher: SAGE Publications. [Online]. Available: <https://doi.org/10.1177/1475921719883202>
- [59] J. Yang, F. Yang, L. Zhang, R. Li, S. Jiang, G. Wang, L. Zhang, and Z. Zeng, "Bridge health anomaly detection using deep support vector data description," *Neurocomputing*, vol. 444, pp. 170–178, Jul. 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0925231221001211>
- [60] H. Son, Y. Jang, S.-E. Kim, D. Kim, and J.-W. Park, "Deep Learning-Based Anomaly Detection to Classify Inaccurate Data and Damaged Condition of a Cable-Stayed Bridge," *IEEE Access*, vol. 9, pp. 124 549–124 559, 2021, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9497115>
- [61] Y. Wang, H. Qin, and F. Miao, "A Multi-Path Encoder Network for GPR Data Inversion to Improve Defect Detection in Reinforced Concrete," *Remote Sensing*, vol. 14, no. 22, p. 5871, Jan. 2022, number: 22 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/2072-4292/14/22/5871>
- [62] F. Hake, L. Göttert, I. Neumann, and H. Alkhatib, "Using Machine-Learning for the Damage Detection of Harbour Structures," *Remote Sensing*, vol. 14, no. 11, p. 2518, Jan. 2022, number: 11 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/2072-4292/14/11/2518>
- [63] M. Shimizu, S. Perinpanayagam, and B. Namoano, "A Real-Time Fault Detection Framework Based on Unsupervised Deep Learning for Prognostics and Health Management of Railway Assets," *IEEE Access*, vol. 10, pp. 96 442–96 458, 2022, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9882101>
- [64] S. Gong, X. Feng, G. Zhang, and F. Ansari, "Distributed detection of internal cavities in concrete-filled steel tube arch bridge elements," *Structural Health Monitoring*, vol. 22,

- no. 1, pp. 657–671, Jan. 2023, publisher: SAGE Publications. [Online]. Available: <https://doi.org/10.1177/14759217221088457>
- [65] P. Ribalta Lorenzo, L. Tulczyjew, M. Marcinkiewicz, and J. Nalepa, “Hyperspectral Band Selection Using Attention-Based Convolutional Neural Networks,” *IEEE Access*, vol. 8, pp. 42 384–42 403, 2020, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9019632>
- [66] G. Xu, C. Cheng, W. Yang, W. Xie, L. Kong, R. Hang, F. Ma, C. Dong, and J. Yang, “Oceanic Eddy Identification Using an AI Scheme,” *Remote Sensing*, vol. 11, no. 11, p. 1349, Jan. 2019, number: 11 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/2072-4292/11/11/1349>
- [67] F. Harrou, A. Dairi, Y. Sun, and F. Kadri, “Detecting Abnormal Ozone Measurements With a Deep Learning-Based Strategy,” *IEEE Sensors Journal*, vol. 18, no. 17, pp. 7222–7232, Sep. 2018, conference Name: IEEE Sensors Journal. [Online]. Available: <https://ieeexplore.ieee.org/document/8401490>
- [68] X. Xu, L. Huang, X. Liu, and G. Fang, “DeepMAD: Deep Learning for Magnetic Anomaly Detection and Denoising,” *IEEE Access*, vol. 8, pp. 121 257–121 266, 2020, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9133058>
- [69] S. Tilon, F. Nex, N. Kerle, and G. Vosselman, “Post-Disaster Building Damage Detection from Earth Observation Imagery Using Unsupervised and Transferable Anomaly Detecting Generative Adversarial Networks,” *Remote Sensing*, vol. 12, no. 24, p. 4193, Jan. 2020, number: 24 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/2072-4292/12/24/4193>
- [70] S. Miao and W.-H. Hung, “River Flooding Forecasting and Anomaly Detection Based on Deep Learning,” *IEEE Access*, vol. 8, pp. 198 384–198 402, 2020, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9245480>
- [71] I. T. Nicholas, J. R. Park, K. Jung, J. S. Lee, and D.-K. Kang, “Anomaly Detection of Water Level Using Deep Autoencoder,” *Sensors*, vol. 21, no. 19, p. 6679, Jan. 2021, number: 19 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/21/19/6679>
- [72] J. Chen and Y. K. Cho, “CrackEmbed: Point feature embedding for crack segmentation from disaster site point clouds with anomaly detection,” *Advanced Engineering Informatics*, vol. 52,

- p. 101550, Apr. 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1474034622000258>
- [73] A. Dairi, F. Harrou, Y. Sun, and M. Senouci, "Obstacle Detection for Intelligent Transportation Systems Using Deep Stacked Autoencoder and k -Nearest Neighbor Scheme," *IEEE Sensors Journal*, vol. 18, no. 12, pp. 5122–5132, Jun. 2018, conference Name: IEEE Sensors Journal. [Online]. Available: <https://ieeexplore.ieee.org/document/8352801>
- [74] C. Huang, X. Wang, J. Cao, S. Wang, and Y. Zhang, "HCF: A Hybrid CNN Framework for Behavior Detection of Distracted Drivers," *IEEE Access*, vol. 8, pp. 109 335–109 349, 2020, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9113267>
- [75] A. Mur, L. Travé-Massuyès, E. Chanthery, R. Pons, and P. Ribot, "A Neural Algorithm for the Detection and Correction of Anomalies: Application to the Landing of an Airplane," *Sensors*, vol. 22, no. 6, p. 2334, Jan. 2022, number: 6 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/22/6/2334>
- [76] B. Peralta, R. Soria, O. Nicolis, F. Ruggeri, L. Caro, and A. Bronfman, "Outlier Vehicle Trajectory Detection Using Deep Autoencoders in Santiago, Chile," *Sensors*, vol. 23, no. 3, p. 1440, Jan. 2023, number: 3 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/23/3/1440>
- [77] D. Cheng, Y. Fan, S. Fang, M. Wang, and H. Liu, "ResNet-AE for Radar Signal Anomaly Detection," *Sensors*, vol. 22, no. 16, p. 6249, Jan. 2022, number: 16 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: <https://www.mdpi.com/1424-8220/22/16/6249>
- [78] S. Bouchenak, R. Merzougui, F. Harrou, A. Dairi, and Y. Sun, "A Semi-Supervised Modulation Identification in MIMO Systems: A Deep Learning Strategy," *IEEE Access*, vol. 10, pp. 76 622–76 635, 2022, conference Name: IEEE Access. [Online]. Available: <https://ieeexplore.ieee.org/document/9832883>
- [79] W. Xie, Y. Li, J. Lei, J. Yang, J. Li, X. Jia, and Z. Li, "Unsupervised spectral mapping and feature selection for hyperspectral anomaly detection," *Neural Networks*, vol. 132, pp. 144–154, Dec. 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0893608020302999>
- [80] S. Plakias and Y. S. Boutalis, "Exploiting the generative adversarial framework for one-class multi-dimensional fault detection," *Neurocomputing*, vol. 332, pp. 396–405, Mar. 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S092523121831498X>

- [81] M. Ullah, S. Akbar, A. Raza, and Q. Zou, "Deepavp-tppred: identification of antiviral peptides using transformed image-based localized descriptors and binary tree growth algorithm," *Bioinformatics*, p. btae305, 2024.
- [82] S. Akbar, Q. Zou, A. Raza, and F. K. Alarfaj, "iafps-mv-bitcn: Predicting antifungal peptides using self-attention transformer embedding and transform evolutionary based multi-view features with bidirectional temporal convolutional networks," *Artificial Intelligence in Medicine*, vol. 151, p. 102860, 2024.
- [83] S. Akbar, A. Raza, and Q. Zou, "Deepstacked-avps: predicting antiviral peptides using tri-segment evolutionary profile and word embedding based multi-perspective features with deep stacking model," *BMC bioinformatics*, vol. 25, no. 1, p. 102, 2024.
- [84] D. Moher, L. Shamseer, M. Clarke, D. Gherzi, A. Liberati, M. Petticrew, P. Shekelle, L. A. Stewart, and PRISMA-P Group, "Preferred reporting items for systematic review and meta-analysis protocols (PRISMA-P) 2015 statement," *Systematic Reviews*, vol. 4, no. 1, p. 1, Jan. 2015. [Online]. Available: <https://doi.org/10.1186/2046-4053-4-1>
- [85] Y. Zhou, B. Li, J. Wang, E. Rocco, and Q. Meng, "Discovering unknowns: Context-enhanced anomaly detection for curiosity-driven autonomous underwater exploration," *Pattern Recognition*, vol. 131, p. 108860, Nov. 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0031320322003417>
- [86] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010: 19th International Conference on Computational Statistics Paris France, August 22-27, 2010 Keynote, Invited and Contributed Papers*. Springer, 2010, pp. 177–186.
- [87] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [88] A. Cardellicchio, M. Nitti, C. Patruno, N. Mosca, M. di Summa, E. Stella, and V. Renò, "Automatic quality control of aluminium parts welds based on 3D data and artificial intelligence," *Journal of Intelligent Manufacturing*, Apr. 2023. [Online]. Available: <https://doi.org/10.1007/s10845-023-02124-1>
- [89] C. Jiang, S. Chen, Z. Zhang, and R. Li, "Energy meter patch resistance and welding spot anomaly detection method based on machine vision," in *Journal of Physics: Conference Series*, vol. 2428, no. 1. IOP Publishing, 2023, p. 012045.

- [90] R. Miao, Z. Jiang, Q. Zhou, Y. Wu, Y. Gao, J. Zhang, and Z. Jiang, "Online inspection of narrow overlap weld quality using two-stage convolution neural network image recognition," *Machine Vision and Applications*, vol. 32, pp. 1–14, 2021.
- [91] P. Tripicchio, G. Camacho-Gonzalez, and S. D'Avella, "Welding defect detection: Coping with artifacts in the production line," *The International Journal of Advanced Manufacturing Technology*, vol. 111, pp. 1659–1669, 2020.
- [92] S. Zhou, Y. Chen, D. Zhang, J. Xie, Y. Zhou *et al.*, "Classification of surface defects on steel sheet using convolutional neural networks," *Mater. Technol*, vol. 51, no. 1, pp. 123–131, 2017.
- [93] "Reducing the Dimensionality of Data with Neural Networks | Science." [Online]. Available: <https://www.science.org/doi/full/10.1126/science.1127647>
- [94] K. Meyer and V. Mahalec, "Anomaly detection methods for infrequent failures in resistive steel welding," *Journal of Manufacturing Processes*, vol. 75, pp. 497–513, 2022.
- [95] G. Yang, Y. Zhong, L. Yang, J. Li, Y. Liang, and R. Du, "Anomaly detection of automobile welding workstation based on stacked autoencoders and long short-term memory network," in *2021 China Automation Congress (CAC)*. IEEE, 2021, pp. 1433–1437.
- [96] A. Savitzky and M. J. E. Golay, "Smoothing and Differentiation of Data by Simplified Least Squares Procedures." *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, Jul. 1964, publisher: American Chemical Society. [Online]. Available: <https://doi.org/10.1021/ac60214a047>
- [97] F. Chollet *et al.*, "Keras," <https://keras.io>, 2015.
- [98] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," Jan. 2017, arXiv:1412.6980 [cs]. [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [99] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814.
- [100] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *Artificial Neural Networks and Machine Learning–ICANN 2011: 21st International Conference on Artificial Neural Networks, Espoo, Finland, June 14-17, 2011, Proceedings, Part I 21*. Springer, 2011, pp. 52–59.
- [101] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using lstms," in *International conference on machine learning*. PMLR, 2015, pp. 843–852.

- [102] L. Chen, X. Yao, P. Xu, S. K. Moon, and G. Bi, "Rapid surface defect identification for additive manufacturing with in-situ point cloud processing and machine learning," *Virtual and Physical Prototyping*, vol. 16, no. 1, pp. 50–67, 2021. [Online]. Available: <https://doi.org/10.1080/17452759.2020.1832695>
- [103] E. Ergin, S. Stumm, and S. Brell-Cokcan, "Robotic iot-enabled 1d line scanner integration for 3d point cloud data processing," in *Proceedings of the 41st International Symposium on Automation and Robotics in Construction*, V. Gonzalez-Moret, J. Zhang, B. García de Soto, and I. Brilakis, Eds. Lille, France: International Association for Automation and Robotics in Construction (IAARC), June 2024, pp. 1168–1175.
- [104] A. Venezia, I. Romata, G. Nicola, R. Fausti, M. Nitti, A. Liso, V. Renò, N. Pedrocchi, M. Beschi, N. Longo, G. Di Stefano, and S. Panicucci, "Enabling cognitive robotics through autonomous motion planning and model-driven intuitive programming," in *European Robotics Forum 2024*, C. Secchi and L. Marconi, Eds. Cham: Springer Nature Switzerland, 2024, pp. 224–229.
- [105] Z. He, H. Cao, P. Zhao, Q. Zhu, X. Wu, W. Feng, and S. Guo, "Noncontact measurement of elastic constants of anisotropic cfrp laminate using orthogonal cross-scan point source-point receiver laser ultrasound," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–12, 2024.
- [106] A. Salazar, A. Rodriguez, N. Vargas, and L. Vergara, "Improving classification of composite materials measured by ultrasounds using classifier combination," in *2021 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2021, pp. 1582–1587.
- [107] M. Meng, Y. J. Chua, E. Wouterson, and C. P. K. Ong, "Ultrasonic signal classification and imaging system for composite materials via deep convolutional neural networks," *Neurocomputing*, vol. 257, pp. 128–135, 2017, machine Learning and Signal Processing for Big Multimedia Analysis. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0925231217301522>
- [108] Y. Guo, Z. Xiao, L. Geng, J. Wu, F. Zhang, Y. Liu, and W. Wang, "Fully convolutional neural network with gru for 3d braided composite material flaw detection," *IEEE Access*, vol. 7, pp. 151 180–151 188, 2019.
- [109] S. McKnight, S. G. Pierce, E. Mohseni, C. MacKinnon, C. MacLeod, T. O'Hare, and C. Loukas, "A comparison of methods for generating synthetic training data for domain adaption of deep learning models in ultrasonic non-destructive evaluation," *NDT & E International*, vol. 141, p. 102978, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0963869523001937>

- [110] V. Tunukovic, S. McKnight, R. Pyle, Z. Wang, E. Mohseni, S. Gareth Pierce, R. K. W. Vithanage, G. Dobie, C. N. MacLeod, S. Cochran, and T. O'Hare, "Unsupervised machine learning for flaw detection in automated ultrasonic testing of carbon fibre reinforced plastic composites," *Ultrasonics*, vol. 140, p. 107313, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0041624X24000751>
- [111] F. Graziano, C. Tortora, V. Vespini, M. Rippa, V. Dentico, F. Leone, N. Gallo, E. Stella, P. Russo, S. Coppola, and P. Ferraro, "Non-destructive techniques for quality control of composite materials," in *2023 IEEE 10th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, 2023, pp. 529–533.
- [112] I. Lucarini, F. Maita, L. Maiolo, and A. Savoia, "Flexible cmut as smart tool for nondestructive testing of aircraft composite structures," in *2021 IEEE 8th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, 2021, pp. 397–401.
- [113] C.-A. Holst, V. Lohweg, K. Röckemann, and A. Steinmetz, "Lamb wave-based quality inspection of repaired carbon fibre reinforced polymers for on-site aircraft maintenance," in *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2019, pp. 1643–1646.
- [114] A. Cardellicchio, M. Nitti, C. Patruno, N. Mosca, M. di Summa, E. Stella, and V. Renò, "Automatic quality control of aluminium parts welds based on 3d data and artificial intelligence," *Journal of Intelligent Manufacturing*, vol. 35, no. 4, pp. 1629–1648, 2024.
- [115] N. Mosca, C. Patruno, V. Renò, M. Nitti, and E. Stella, "Qualitative comparison of methodologies for detecting surface defects in aircraft interiors," in *2021 IEEE 8th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*. IEEE, 2021, pp. 215–220.
- [116] S. Timilsina, S. M. Jang, C. W. Jo, Y. N. Kwon, K.-S. Sohn, K. H. Lee, and J. S. Kim, "Identification of location and geometry of invisible internal defects in structures using deep learning and surface deformation field," *Advanced Intelligent Systems*, vol. 5, no. 12, p. 2300314, 2023.
- [117] F. Kähler, O. Schmedemann, and T. Schüppstuhl, "Anomaly detection for industrial surface inspection: Application in maintenance of aircraft components," *Procedia CIRP*, vol. 107, pp. 246–251, 2022.
- [118] G. Vesala, V. Ghali, D. R. Sastry, and R. Naik, "Deep anomaly detection model for composite inspection in quadratic frequency modulated thermal wave imaging," *NDT & E International*, vol. 132, p. 102710, 2022.

- [119] A. Ghamisi, T. Charter, L. Ji, M. Rivard, G. Lund, and H. Najjaran, "Anomaly detection in automated fibre placement: learning with data limitations," *Frontiers in Manufacturing Technology*, vol. 4, p. 1277152, 2024.
- [120] Y. M. Tang, A. W. Ip, and W. Li, "Artificial intelligence approach for aerospace defect detection using single-shot multibox detector network in phased array ultrasonic," in *IoT and Spacecraft Informatics*. Elsevier, 2022, pp. 1–27.
- [121] S. Yoon, W. J. Cantwell, C. Y. Yeun, C.-S. Cho, Y.-J. Byon, T.-Y. Kim *et al.*, "Defect detection in composites by deep learning using solitary waves," *International Journal of Mechanical Sciences*, vol. 239, p. 107882, 2023.
- [122] G. Guaragnella, D. Morelli, T. D'Orazio, U. Galietti, B. Trentadue, and R. Marani, "Defect detection by a deep learning approach with active ir thermography," in *2022 8th International Conference on Control, Decision and Information Technologies (CoDIT)*, vol. 1, 2022, pp. 27–32.
- [123] XARION, "Xarion's laser-excited acoustics." [Online]. Available: <https://xarion.com/en/applications/non-destructive-testing/quality-control-of-composite-materials>
- [124] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.
- [125] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.
- [126] C. Patruno, A. Liso, V. Renò, V. Vespini, S. Coppola, P. Ferraro, N. Gallo, G. Del Prete, V. Dentico, D. Chirico *et al.*, "A wavelet-based method for defect detection in aircraft components by using ultrasonic technology," in *2024 11th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*. IEEE, 2024, pp. 513–518.
- [127] V. James, D. Carswell, J. Riise, P. Ian Nicholson, N. Graf, N. Huber, M. Gärtner, B. Reitingner, E. Scherleitner, P. Burgholzer, J. Phipps, D. Burns, and C. Incarnato, "Robot deployed laser-ultrasonic ndt system for inspection of large aircraft structures," vol. 1024, no. 1, 2021.
- [128] P. Steduto, J. Faurès, J. Hoogeveen, J. Winpenny, and J. Burke, *Coping with Water Scarcity: An Action Framework for Agriculture and Food Security*, ser. Water reports. Food and Agriculture Organization of the United Nations, 2012. [Online]. Available: <https://books.google.it/books?id=3ikwIAEACAAJ>

- [129] J. Wallace, "Increasing agricultural water use efficiency to meet future food production," *Agriculture, Ecosystems & Environment*, vol. 82, no. 1, pp. 105–119, 2000. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167880900002206>
- [130] I. Lunt, S. Hubbard, and Y. Rubin, "Soil moisture content estimation using ground-penetrating radar reflection data," *Journal of Hydrology*, vol. 307, no. 1, pp. 254–269, 2005. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0022169404005165>
- [131] D. Comite, A. Galli, S. Lauro, E. Mattei, and E. Pettinelli, "Analysis of gpr early-time signal features for the evaluation of soil permittivity through numerical and experimental surveys," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, pp. 1–10, 09 2015.
- [132] N. Peplinski, F. Ulaby, and M. Dobson, "Dielectric properties of soils in the 0.3-1.3-ghz range," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 33, no. 3, pp. 803–807, 1995.
- [133] C. Warren, A. Giannopoulos, and I. Giannakis, "gprmax: Open source software to simulate electromagnetic wave propagation for ground penetrating radar," *Computer Physics Communications*, vol. 209, pp. 163–170, 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0010465516302533>
- [134] H. Namdari, M. Moradikia, S. Zekavat, R. Askari, O. Mangoubi, and D. Petkie, "Advancing precision agriculture: Machine learning-enhanced gpr analysis for root-zone soil moisture assessment in mega farms," *IEEE Transactions on AgriFood Electronics*, vol. 3, no. 1, pp. 98–109, 2025.
- [135] J. Zheng, X. Teng, J. Liu, and X. Qiao, "Convolutional neural networks for water content classification and prediction with ground penetrating radar," *IEEE Access*, vol. 7, pp. 185 385–185 392, 2019.
- [136] G. C. Topp, J. L. Davis, and A. P. Annan, "Electromagnetic determination of soil water content: Measurements in coaxial transmission lines," *Water Resources Research*, vol. 16, no. 3, pp. 574–582, 1980. [Online]. Available: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/WR016i003p00574>
- [137] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017. [Online]. Available: <https://arxiv.org/abs/1707.06347>
- [138] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," 2018. [Online]. Available: <https://arxiv.org/abs/1801.01290>

- [139] K. Perlin, “An image synthesizer,” in *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '85. New York, NY, USA: Association for Computing Machinery, 1985, p. 287–296. [Online]. Available: <https://doi.org/10.1145/325334.325247>

- [140] R. Rettore, “Leveraging ground penetrating radar with deep reinforcement learning for subsurface-informed navigation in autonomous agricultural robotics,” Master’s thesis, University of Padua, 2025.