

AN END-TO-END FRAMEWORK FOR ANOMALY DETECTION, AND CATEGORIZATION

by

MD. Farhan Islam

21301254

Rehnuma Islam

21301277

Syed Rahin Reza

21301279

Saifa Tasnim

21301713

Anipa Akter Nipu

21301085

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
December 2025.

© 2025. Brac University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

MD. Farhan Islam

21301254

Rehnuma Islam

21301277

Syed Rahin Reza

21301279

Saifa Tasnim

21301713

Anipa Akter Nipu

21301085

Approval

The thesis/project titled “ An End-to-End Framework for Anomaly Detection, and Categorization ” submitted by

1. MD. Farhan Islam (21301254)
2. Rehnuma Islam (21301277)
3. Syed Rahin Reza (21301279)
4. Saifa Tasnim (21301713)
5. Anipa Akter Nipu (21301085)

of Fall, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science in Fall 2025 Year.

Examining Committee:

Supervisor:
(Member)

Mr. Rafeed Rahman

Senior Lecturer
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi

Associate Professor & Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

In this study, we proposed an end-to-end framework for anomaly detection, classification in Industry 4.0 using deep learning models YOLO V8 and ResNet on the MVTec Anomaly Detection(MVTec AD) dataset. The framework is based on defect detection, anomaly localization. The multitask queues in YOLO V8 guarantee both: fast and precise detection in real time, while ResNet primarily suited for classification, complete with top notch precision and recall metrics. The metrics used for evaluation (including AUC, accuracy, precision, recall, F1 score and AP) confirm the good performance of the models. We also provide decision surface visualizations through Grad-CAM and Integrated Gradients that will help you understand some of the decisions made by the model. The YOLO V8 performed optimal on real-time detection tasks and ResNet performed best on classification accuracy, as highlighted through the results. This framework allows for the automation of anomaly detection and the resolution through investigation, unlocking future opportunities for real time anomaly detection and management.

Keywords: Industrial anomaly detection, Visual inspection, Defect localization, Damage quantification, Explainable AI, Grad-CAM, YOLOv8 classification, ResNet, Robustness to noise, Entropy and confidence metrics, KL divergence, MVTEC AD, Post-hoc attribution, Foreground segmentation, GrabCut refinement.

Acknowledgement

First of all, sincerest thanks must go to almighty Allah for having enabled us to complete our thesis a profit without encountering any major problems. Elements of this work are dedicated to our supervisor, Mr. Rafeed Rahman Sir, who edged us along with beneficial instruction and continuous support. Whenever we needed his help, it was there.

Table of Contents

Declaration	i
Approval	ii
Ethics Statement	iv
Abstract	iv
Dedication	v
Acknowledgment	v
Table of Contents	vi
List of Figures	ix
List of Tables	x
Nomenclature	x
1 Introduction	1
1.1 Background	1
1.2 Rationale and Motivation	1
1.3 Problem Statement	2
1.4 Research Objectives	2
1.5 Methodology in Brief	3
1.6 Scopes and Challenges	3
1.7 Thesis Contributions	4
1.8 Thesis Organization	4
2 Literature Review	5
2.1 Overview of Anomaly Detection in Industrial Environments	5
2.2 Traditional Anomaly Detection Methods	5
2.3 Deep Learning Approaches for Anomaly Detection	6
2.4 YOLOv8 for Anomaly Detection	7
2.5 ResNet for Anomaly Detection	8
2.6 Comparison of Deep Learning Models for Anomaly Detection	8
2.7 Review of Previous Research	8

3	Requirements, Impacts, and Constraints	10
3.1	Technical Requirements	10
3.1.1	Hardware Requirements	10
3.1.2	Software Requirements	11
3.2	Impacts of the Research	11
3.2.1	Industry Transformation	11
3.2.2	Societal and Environmental Impacts	12
3.3	Constraints and Challenges	12
3.3.1	Data Quality and Quantity	12
3.3.2	Model Generalization	12
3.3.3	Computational and Resource Constraints	12
3.4	Ethical Considerations	13
4	Dataset and Preprocessing	14
4.1	Introduction to the MVTec AD Dataset	14
4.2	Dataset Structure and Categories	14
4.2.1	Object Categories:	16
4.2.2	Texture Categories:	16
4.3	Dataset Preprocessing	16
4.3.1	Image Resizing and Normalization	16
4.3.2	Data Augmentation	17
4.3.3	Labeling and Anomaly Localization	17
4.4	Data Splitting for Training and Evaluation	17
5	Methodology	19
5.1	Overview	19
5.2	Data Preprocessing	19
5.3	Dataset	20
5.4	Model Architecture	20
5.4.1	YOLOv8 (You Only Look Once)	20
5.4.2	ResNet (Residual Networks) Based CNN	21
5.4.3	Model Integration and Fine-Tuning	21
5.5	Anomaly Detection and Categorization	23
5.6	Model Evaluation and Testing	27
5.7	Pseudo Code	29
6	Results and Discussion	31
6.1	Overview of Experimental Setup	31
6.2	Model Performance Evaluation	31
6.2.1	Accuracy	32
6.2.2	Precision, Recall, and F1 Score	32
6.2.3	YOLO V8 Performance	32
6.2.4	ResNet Based CNN Performance	35
6.2.5	Area Under the ROC Curve (AUC)	37
6.2.6	Average Precision (AP)	37
6.3	Anomaly Localization and Detection	38
6.3.1	YOLO V8 Localization Performance	38
6.3.2	ResNet Localization Performance	38
6.4	Robustness under Noise	38

6.5	Explainability of the Models	40
6.5.1	Grad-CAM Visualizations	40
6.5.2	Integrated Gradients (IG)	40
6.6	Discussion of Results	41
6.7	Results Summary Table	41
6.8	Conclusion	42
7	Conclusion and Future Work	43
7.1	Conclusion	43
7.2	Future Work	43
7.2.1	Expanding the Dataset	43
7.2.2	Model Generalization and Transfer Learning	44
7.2.3	Improving Localization Accuracy	44
7.2.4	Real Time Performance and Deployment	44
7.3	Final Thoughts	44
	Bibliography	49

List of Figures

4.1	Anomaly Examples from MVTec AD Dataset	15
5.1	Example image from MVTec AD dataset	20
5.2	Architecture of our proposed methodology for Anomaly Detection . .	21
5.3	Architecture of YOLO V8 for Anomaly Detection	22
5.4	Architecture of ResNet based CNN for Anomaly Detection	23
5.5	Example of anomaly detection and categorization result of Screw using YOLO V8 and ResNet.	24
5.6	Another Example of anomaly detection and categorization result of Zipper using YOLO V8 and ResNet.	25
5.7	More Example of anomaly detection and categorization result of Bottle using YOLO V8 and ResNet.	26
5.8	Grad-CAM visualization for Screw using YOLOv8 and ResNet. . . .	27
5.9	Grad-CAM visualization for Wood using YOLOv8 and ResNet. . . .	27
5.10	Grad-CAM visualization for Zipper using YOLOv8 and ResNet. . . .	28
5.11	Grad-CAM visualization for Metal Nut using YOLOv8 and ResNet. .	28
6.1	Accuracy and Loss curves for YOLO V8.	34
6.2	Confusion Matrix for YOLO V8.	35
6.3	Accuracy and Loss curves for ResNet based CNN.	36
6.4	Confusion Matrix for ResNet Based CNN.	37
6.5	Noisy image with Gaussian noise added.	39
6.6	Noisy image with Gaussian noise added.	39
6.7	Noisy image with Gaussian noise added.	39
6.8	Noisy image with Gaussian noise added.	40
6.9	Grad-CAM visualization highlighting anomaly regions.	40

List of Tables

6.1	Dataset Summary and Train-Test Split	31
6.2	Final Evaluation Metrics on the Test Set	33
6.3	Class-wise Performance Metrics on the Test Set	33
6.4	Class-wise Precision, Recall, F1-score, and Overall Accuracy	36
6.5	Performance Comparison between YOLOv8 and CNN-based Classifier	41

Chapter 1

Introduction

1.1 Background

Anomaly detection is finding data that significantly differs from normal patterns. Indicating to possible critical incidents when found, these abnormalities can also signal errors or fraud in several fields, including finance, healthcare, cybersecurity and manufacturing [1]. Efficient detection of these anomalies is important for system reliability, product quality and risk control.

In recent years, deep learning models, particularly convolutional neural networks (CNNs), have become better known for their capability to process and analyze complex high-dimensional data such as images and sensor data [2]. The success of CNN models like YOLO V8 and ResNet in computer vision tasks has demonstrated enormous potential for the automated detection of anomalies. The models can identify complex patterns in data and hence are well suited to tasks which entail extracting difficult to reach manual features from those patterns [3].

However, a gap still exists in the development of anomaly detection techniques, particularly in identifying anomalies and determining the appropriate classification for each type. [4]. This work mainly aims to develop a full fledged system for modern requirements through advanced deep learning methods: end to end detection, classification of anomalies.

1.2 Rationale and Motivation

This study has been based on the increasing demand for automation and efficiency in anomaly detection, especially in industrial scenarios. Even the slightest manufacturing defect of the product or item in an industry like manufacturing faces heavy losses during production and loss of faith from the customers [5]. One of the oldest methods is through manual inspections. It takes time, but always open to human error [6].

Increasingly, with the proliferation of large scale image databases and deep learning advances, then we can start automating process of anomaly detection. Process for visual anomaly detection YOLO V8 and ResNet are both powerful CNN architectures that have shown good development prospects [7]. But these models usually

concentrate only on identifying point anomalies so there is no common framework inbuilt for clustering anomalies [8].

This study discovers novel technique for this not just for detecting anomalies but also detecting them in categories. The future of multi industrial technology is sure to be in real world manufacturing, at the center of the future of automation quality control, improving operational efficiency and cutting the massive cost of missed defects. [9].

1.3 Problem Statement

Although progress has been achieved in the field of anomaly detection, current solutions in the area are oriented toward detecting the existence of anomalies alone, not much more [10]. To take an example with wider implications for industrial applications, we require a global solution: one that does not just detect these anomalies but also allocates them into two main categories (detect product defects and categories the defect) [11].

This research is about a merger of these two aspects by providing an integrated framework that applies deep learning models specifically Yolo V8 and ResNet to achieve the following objectives:

- Capture objects in industrial datasets (such as images) using camera detectors, with the ability to trigger warnings based on detection results or textual descriptions.
- Identify various types of anomalies, categorize them, and provide actionable insights on how to address them.

The objective is to invent a system which can be directly applied in practical and industrial situations, obviating the need for human inspection and yielding real time quality control information.

1.4 Research Objectives

In this paper we aim to develop a complete deep learning based system to detect anomalies from industrial data or images of the product in production lines. The proposed system will facilitate the decision making process because it will categorize detected anomalies into classes. Thus besides detection and classification that can be used to reduce defect level, minimize material wastes and improve the overall efficiency of factory performance. The framework is tested on real world industrial data to compare it with conventional methods of anomaly detection, as well as other existing state of the art models in deep learning, to make its functioning robust and reliable.

In order to pursue these goals, the proposed system employs an advanced deep learning model, such as YOLO V8 for real time anomaly detection and ResNet for high precision image classification. The individual performance models are developed using the MVTec Anomaly Detection dataset, which replicates industrial settings

for testing that the system’s defect detection and classification capabilities are functional. The system detects and classifies anomalies. Overall, its performance is then assessed through several metrics such as accuracy, precision, recall and F1 score before being compared to other traditional anomaly detection methods and state of the art deep learning methods. The result aims to show the efficiency of the system for real industrial scenarios. The system has a high potential to improve, in conjunction with operational processes, reducing waste, and contributing to the decision process in terms of quality control.

1.5 Methodology in Brief

We have developed a complete system for detecting defects with YOLOv8 and ResNet combined advanced deep learning techniques. These naves after the systems design include a step by step description of the project process

- **Dataset:** The MVTec Anomaly Detection dataset, which contains labeled images of defective industrial products, was used for training and evaluation.
- **Model Training:** Training YOLO V8 and ResNet models on the preprocessed images with labelled defect images as inputs to learn patterns for anomaly detection.
- **Anomaly Detection:** Using the trained model to find anomalies in new, unseen images.
- **Anomaly Categorization:** Division of detected anomalies into defined categories (for example, types of defects)
- **Evaluation:** The assessment of system performance by measuring metrics like accuracy, precision rate, recall rate, and F1 score.

1.6 Scopes and Challenges

Scopes

This paper mainly researches the use of image data sets to detect anomalies in the industrial mills. Consequently, it will mainly focus on detecting defective products in manufacturing. As this is a field suited to automatic visual inspection, the foundation concept should be nothing new. However, the framework is extendable to other fields where image based anomaly detection programs might be useful.

Challenges

Challenges experienced during this research included an imbalance of data:

- **Data Imbalance:** On the one hand there were more good samples than it was possible for any one model to use effectively. On the other hand there were a few defective product samples.

- **Model Complexity:** Training deep learning models like YOLOv8 and ResNet on large datasets requires substantial computational resources and time.
- **Real-time Operation:** Therefore, we take these factors into our framework design as top level performance indicators. In other words, the framework is of little practical use without real time operation.

1.7 Thesis Contributions

Here are the key findings from this study:

- **Developing an Integrated Framework:** this proposed framework with Resnet and Yolo V8 can cover a wider area and not only detect anomalies, it also categorizes them.
- **Comparison of Models:** Differential abnormality comprises anomalies such as defect on the pill, or hole on the hazelnut. Most of them happen at boundaries between different image regions.

1.8 Thesis Organization

This thesis has the following structure. Chapter 1 starts with the research introduction including the historical context, the statement of the problem, and the description of the research purpose and importance. In chapter 2, the literature review is provided in detail, analyzing the existing methods of anomaly detection, specifically, machine learning based methods and the ways they can be applied in the industry. The chapter 3 addresses technical and computing needs of constructing an anomaly detection framework, the possible effects of the framework and limitations or challenges in the development. Chapter 4 explains which datasets will be utilized in the study as well as the steps that will be taken to preprocess the data and convert it into something that can be used by machine learning models. Chapter 5 describes the proposed methodology, which includes the usage of YOLOv8 and ResNet based convolutional neural networks, training approach, and evaluation measures used. Chapter 6 reports and discusses the experimental findings with statistics of performance measures and confusion matrices. Lastly, Chapter 7 summarizes the main findings and gives the potential future research directions in the field of anomaly detection, which is the end of the thesis.

Chapter 2

Literature Review

2.1 Overview of Anomaly Detection in Industrial Environments

Anomaly detection is a critical business in a wide range of fields, such as manufacturing, finance, healthcare and cyber security [12]. It is about identifying data points, events, or patterns that deviate significantly from the norm or anticipated behavior, as these digressions, or anomalies, are very often critical incidents failures or fraudulent activities, their detection is essential for the integrity of systems and to ensure the security of processes [13].

The standard classification of traditional anomaly detection techniques includes statistical method, distance based manner and classification based approach [14]. Statistical methods is based on normal behavior model, when the data was inconsistent and data points of it exceeds the model range time point occur, is identified as an anomaly [15]. Distance based methods is distance of data points away from most others classified to be abnormal Calling those Classifications are carried out by method: Learning from machine models, normal and abnormal data are divided with the machine learning a boundary between two classes [16].

However, traditional method is difficult to handle high dimensional and unstructured data, especially when all this wire in image data with manual feature extraction: its laborious work that can introduce errors [17]. Deep learning, however, has raised hopes for breaking with tradition. It is able to automatically learn features from raw data without any need for human intervention. This chapter provides a broad overview of various anomaly detection techniques, focusing particularly on deep learning methods and their application in industrial domains.

2.2 Traditional Anomaly Detection Methods

Traditional methods for detecting outliers have been widely used in a variety of fields, particularly in low dimensional data. Below are some techniques:

- **Statistical Methods:** These methods are based on the notion that normal data follow a particular distribution, and detect outliers by classifying data

points which deviate from the distribution as Principles of ordinary. In practice, this includes such tools as z-score and Gaussian mixture models (GMM) [18]. However, the latter method is not considered one of those who work well when dealing with complicated structures or complex data in general KDE (K-cardinality estimation), in contrast is a definite choice for high-dimensional data [19].

- **Distance-Based Methods:** They detect outliers by calculating how far removed points are from one another in space. k-Nearest Neighbors (k-NN), for instance, calculates the distance between a point and its neighbors. If a point lies too far away from any other points then it is classed as an outlier. [20]. Another algorithm is the DBSCAN, which groups points together (in taste, not representation) with k-means does and then marks remaining separate individuals as outliers [21]. The main idea here is density. It detects individual points in regions of clear rare density where the neighbours don't even exist [22].
- **Clustering-Based Methods:** The goal of clustering methods, for example k-means or hierarchical clustering, is to put together similar data points [23]. Points which are not clustered or are in small clusters are considered as outliers. While effective in certain situations these algorithms have difficulties when dealing with more complicated datasets or those having overlapping normal distributions [24].

But these methods all suffer from the limitation that they cannot deal with complex high-dimensional data. Large increases in the size of datasets with higher feature counts have meant that deep learning has become more effective compared to other techniques at performing tasks related to detecting anomalies or outlier points.

2.3 Deep Learning Approaches for Anomaly Detection

Deep learning has changed the field of anomaly detection, especially its ability to process high dimensional data such as images and sensor data. Models like Convolutional Neural Networks (CNNs) have become the go to approach for anomaly detection in image data, offering several advantages over traditional methods:

- **Automatic Feature Extraction:** Unlike traditional methods which require manually extracted features encoded in a list or table form, a deep learning model can learn features from raw data. Therefore it is more flexible and adaptable than previous approaches, adapting better to complex datasets [25].
- **Ability to Handle High-Dimensional Data:** Deep learning models, particularly CNNs, work well on high-dimensional data tasks such as images. This enables adjustments in a spatially cartesian coordinate system as well as drawing attention to which traits are important and in what way for further clarification [26].

- **Transfer Learning:** Models like ResNet or YOLOv8 are pre-trained from large-scale datasets and these can be fine-tuned for specific anomaly detection tasks, cutting down training time and raising performance levels on smaller sets of data [27].

Several deep learning architectures have been applied to anomaly detection:

- **Autoencoders:** These are neural networks that learn to compress and then reconstruct input data. In anomaly detection, autoencoders are trained on normal data and anomalies are identified by high reconstruction error. Variants such as convolutional autoencoders or variational autoencoders (VAEs) have also met with success in such tasks [28].
- **Convolutional Neural Networks (CNNs):** CNNs have proven particularly effective in image based anomaly detection tasks. By learning hierarchy features from images, CNNs can alert on subtle anomalies that might not be apparent with traditional methods. YOLOv8 and ResNet are two popular CNN architectures have been used for anomaly detection in various applications [29].

2.4 YOLOv8 for Anomaly Detection

The latest generation of the YOLO (You Only Look Once) models YOLOv8 are becoming one of the most successful object detection systems. Given its performance on detecting objects or anomalies in images, YOLOv8 has proved particularly adept at localizing and classifying problems in a range of fields, from classic car racing circuits to modern offices with high ceilings [30]. Detection is performed by first splitting an image into a grid of cells and then predicting both boxes around the cells inside which likely detection lies, plus classes for each they belong. When first developed by YOLOv8, this method enabled rapid scanning and classification of objects in an image suitable for real-time applications [31].

YOLOv8 has been used in anomaly detection to detect defects in industrial products, anomalies in medical imaging, and monitor environmental conditions etc. It has the ability to detect anomalies in real time that make it especially suitable for quality control in factory environments, where taking right now to rectify defects or faults may be necessary. Difficulties remain. YOLO V8 Strengths are:

- **Real-Time Detection:** YOLOv8 is tuned for swift detection making it ideal for applications that require anomaly to be found immediately.
- **High Precision and Speed:** YOLOv8 combines the ultimate in detection accuracy with rapid inference time. It is suitable for real-time applications.
- **End-to-End Training:** This model can be trained from start to finish on huge data sets and will automatically learn sophisticated features. It is capable of detecting anomalies with little or no manual intervention

However, applying YOLOv8 to industrial anomaly detection can present challenges, particularly when dealing with small anomalies or imbalanced datasets. Further optimization and fine tuning may be required to enhance its performance in these specific scenarios [32].

2.5 ResNet for Anomaly Detection

ResNet (Residual Networks) is just one of the many CNN architectures that have been widely used for anomaly detection. This kind architecture, with its deep network design and residual learning approach, allows us to train much deeper models than ever before by using residual blocks [33]. Thus, ResNet is particularly effective in locating complex features automatically from high dimensional data [34]. In terms of detecting anomalies, ResNet has been used to detect anomalies in images, medical images or sensor data [35]. This deep architecture, captured subtle patterns in the sensor data which are so important if one is to detect a thing out of what is normal ResNet has its advantages in anomaly detection:

- **Complex Features Learning Ability:** With ResNets deep layers, it can learn complex features hierarchically from raw data. As a result, the system is able to effectively detect anomalies in high dimensional data.
- **All round Versatility:** ResNet can be applied in any kind of anomaly detection task just like a chameleon will change its skin color smoothly and without ornate motions when the background switches abruptly to yellow from green, for instance. Simply adapt the pre trained model once again on your own specific data.

Like many CVpr models, ResNet (along with YOLO V8 too) has its limitations. Training deep models requires a great deal of computational resources; if used in such a consuming manner. When using systems on mode min q the training may perform poorly. Over fitting or imbalanced data sets will have bad effect on deep models as they go live.

2.6 Comparison of Deep Learning Models for Anomaly Detection

Both YOLOv8 and ResNet offer excellent performance in anomalous style recognition. Depending on what you need, the two have their own distinct advantages: YOLO V8 is more suited to real time object detection tasks, while ResNet in case time is crucial, it is better employed on tasks demanding deep feature extraction and complex information accuracy.

If a real-time anomaly detection method with high detection rate is required, a blending approach that marries the advantages of YOLO V8 and ResNet may stand out as a strong candidate. YOLO V8 can rapidly capture and localize examples of untypical behavior, while units of the ResNet type would be good to be applied for further classification and feature extraction.

2.7 Review of Previous Research

In recent years, anomaly detection techniques have shown promise in industrial settings, but a number of challenges and limitations have been identified that prevent

them from being widely deployed in practice. A common problem found in multiple papers is that the models are not validated in the real world. Such studies including "Explainable AI in Manufacturing: A Predictive Maintenance Case Study" (2020) [36] and "AI-Enabled Anomaly Detection in Industrial Systems: A New Era in Predictive Maintenance" (2022) [37] used synthetic dataset or very little data which limits the generalizability of their results in real industrial settings. Moreover, several studies, e.g. [38], [39], discontinued generalization due to the small or single-device datasets, which may not cover the industrial data distribution complexities.

A third often reported limitation in the literature is the high complexity of the models and the difficulty to understand or interpret their results. For example "Explainable AI Algorithms for Vibration Data-Based Fault Detection" (2022) [40] reported problems of noise in saliency maps and high computational complexity of SHAP explanation techniques. Since performance strongly relies on the dataset, it can be done with well-designed models, as observed in "Machine Learning for Detecting Anomalies in Industrial Environments" (2024) [41], some domain specific frameworks do not have the capacity to learn the whole variety of industrial environments, demonstrating some gaps.

Last, some studies criticized a lack of scalability and real-world usefulness of these models. However, while some studies, such as AI-Driven HSE Management Systems for Risk Mitigation in the Oil and Gas Industry (2024) [42], [43] highlighted the lack of empirical validation against real-world datasets, others such as Enhancing Industrial Safety: Real time Anomaly Detection with OpenCV on Edge Platforms (2024) [44] identified some of the aforementioned challenges with the limitations of the edge device (lighting, occlusion, and resource constraints). These results highlight the need for additional effort in overcoming such obstacles and improving the reliability and scalability of anomaly detection techniques for practical usage in industrial applications.

Chapter 3

Requirements, Impacts, and Constraints

3.1 Technical Requirements

In principle, developing and deploying an anomaly detection framework that leverages deep learning techniques such as YOLOv8 and ResNet needs to meet multiple technical requirements. It is the technical requirements that get it stable here again. For its systems ensure resistance, the system must have the ability to efficiently identify anomalies in image datasets. This is especially critical with industrial application settings. In this part we introduce the hardware and software components required to realize the design framework.

3.1.1 Hardware Requirements

- **Computational Power:** Because large image datasets can create a much higher demand on such specifications than for normal text datasets, deep learning models must have machines to take advantage of it. To train these models, third party libraries like MatConvNet are needed to make use of Direct GPUs rather than exclusively on CPU cores as before. This capability has become even more critical with today's state of the art DNN. Thanks to MatConvNets ability for speedups via efficient instruction running in hardware. Training deep learning models like YOLOv8 and ResNet demands a lot of computing power. Especially when handling large image datasets that is unrepresentative in other algorithms of its class. GPUs are essential for efficiently training these models. This system is equipped with Graphics Processing Units (GPUs) like NVIDIA's Tesla V100 or A100. Adding speed to matrix computations and cutting training time dramatically. GPUs allow the module to process large data packages as one batch and workspace.
- **Storage:** The training and testing datasets must be stored on high capacity devices. Depending on the size of images and number of samples, solid-state drives (SSD) are faster for data access than hard disk drives (HDD). An industrial image anomaly detection dataset typically requires at least one terabyte. It may range from several gigabytes to several terabytes in size.
- **Memory:** For large scale datasets and data augmentation/model fine tuning

tasks involving many images, sufficient RAM is essential. A system equipped with 32 GB of RAM is recommended for optimum performance in training and testing.

3.1.2 Software Requirements

- **Deep Learning Frameworks:** The implementation of this research signals YOLOv8 and ResNet based models needs to be deep as frameworks such as PyTorch or TensorFlow are used. These frameworks have efficient implementations of neural networks and offer pre-trained models.
- **Data Processing Tools:** For data manipulation, libraries and tools such as NumPy, Pandas, OpenCV are indispensable. Matplotlib and Seaborn are also required to visualize your data. Scikit learn is needed to generate a left graphic showing model performance indicators such as classification accuracy, precision recall (sensitivity) and F1 score.
- **Cloud Platforms:** Where computing resources are limited (such as local storage), cloud based platforms come to the rescue. Such as Google Cloud, Amazon Web Services (AWS), MSDN or Azure all help provide the necessary infrastructure including large models training environment. These platforms provide large and CPU/GPU/TPU instances that greatly speed up the training process.

3.2 Impacts of the Research

The development of an automated anomaly detection framework has many profound implications at both technological and societal levels. These impacts are particularly important in industries where quality control and operational efficiency are vital, like manufacturing, healthcare, or transportation.

3.2.1 Industry Transformation

- **Hardware Security Evaluation:** Traditional quality control processes in manufacturing often rely on manual inspection or heuristics to detect product defects. By automating this process with deep learning models like YOLO V8 and ResNet, the framework can reduce human error significantly, increase detection accuracy, and ensure consistent product quality. This in turn lowers the chances of getting bad goods into customer hands, experience no longer good quietly vanished from sight.
- **Cost Reduction and Efficiency:** By integrating anomaly detection models in the production line, manufacturers may minimize their expenses through early discovery of any defects. What's more to the point, the system can work in real time thus allowing manufacturers make immediate corrective actions. This high level of automation improves efficiency by reducing the amount downtime, ensuring that all production lines smoothly continue operating.

3.2.2 Societal and Environmental Impacts

- **Public Safety and Health:** Anomaly detection frameworks can be applied in the medical field for defects, such as X ray, CT scan, or MRI images. The ability to spot these problems, from tumors to fractures, at an early stage leads to timely measures; that too, makes way for more patients per day, benefiting patients as well as medical institutions.
- **Environmental Sustainability:** It also promotes sustainability as it minimizes waste generated during the production process. It helps reduce the correspondingly greater waste of material, energy consumption, and environmental damage that will result from the need for repetitions in the production process due to defects or non conformance leading to a potential representative of production in a more sustainable and environmentally friendly manner.

3.3 Constraints and Challenges

While possessing great strength, deep learning based anomaly detection also brings critical problems:

3.3.1 Data Quality and Quantity

- **Imbalanced Datasets:** Since the data points for anomalies are scarce, the model may be biased towards the majority class. You can overcome this using oversampling, undersampling, and data augmentation techniques.
- **Data Labeling:** Labeling anomalies is tedious and highly time consuming task owing to the fuzziness in the boundaries of normal and anomalous data. This problem can be reduced by automating labeling and semi supervised learning.

3.3.2 Model Generalization

- **Overfitting:** YOLO V8 and ResNet are deep models that can overfit a small or imbalanced dataset, resulting in low generalization. This can be addressed by regularization, dropout, and early stopping.
- **Transfer Learning:** For fine tuning pre trained models on specific tasks, you may have to experiment with different parameters, learning rates, and modalities fine tuning.

3.3.3 Computational and Resource Constraints

- **Hardware Limitations:** Requires expensive hardware to train high performance models on large datasets Smaller organizations may need cloud based or distributed training solutions.
- **Real-Time Processing:** Execution speed is a critical issue in operationalizing anomaly detection models in practice. Real time processing requires

minimizing the duration of inference while not losing prediction quality. Here, one may adopt model quantization, pruning, and even hardware acceleration.

3.4 Ethical Considerations

As with any AI-driven technology, the development of anomaly detection systems must follow ethical principles. These issues include:

- **Bias and Fairness:** Models should undergo bias tests to ensure that they do not unfairly reject certain types of data. For example, if the training dataset is not representative of all possible defect types, the model might overlook some anomalies. This could result in unequal treatment.
- **Transparency and Accountability:** Anomaly detection models, especially those utilised in key areas such as healthcare, must be transparent in how they make decisions. Stakeholders should be given the means to comprehend system conclusions, and when an anomaly is identified appropriate action must be taken in order to establish responsibility for error.

Chapter 4

Dataset and Preprocessing

4.1 Introduction to the MVTec AD Dataset

The MVTec Anomaly Detection (MVTec AD) dataset [45] was designed especially for testing the performance of anomaly detection methods in industrial inspection tasks. This is one of the most widely known datasets for anomaly detection at present due to its multiple categories and high resolution. It also contains pixel thick annotations that mark out where specifically a particular type of defect or anomaly occurs. The dataset consists of over 5000 high resolution images spread across 15 different object and texture categories, including bottles, capsules, metal nuts and tiles. Each class has two subsets. They are:

- **Train Data:** The Training Data is made up of the original images that are defect free (i.e. samples of everything normal). It is used in training anomaly detection models.
- **Test Data:** Test Data includes normal as well as defective images. The faulty image types vary, scratches, holes or tiny differences in color are all considered defect types that confront one’s model with a great variety of defects.

The precise pixel annotations of anomalies in the test set are one undoubted feature of the MVTec AD dataset. These annotations specify the exact position on the image of each defect, and so the dataset is good for tasks requiring the precise location of anomalies. This is particularly important in industrial applications where precise localization is needed for corrective action.

4.2 Dataset Structure and Categories

The MVTec AD dataset [45] is organized into 15 categories. The training and test sets each include multiple images of the categories. These categories can be subdivided into object categories and texture categories.

The figure 4.1 displays examples from the MVTec Anomaly Detection (MVTec AD) dataset, featuring six different classes: bottle, cable, pill, carpet, hazelnut, and metal nut. Each column represents one class, with the first row showing normal images, the second row highlighting defective images, and the third row marking the

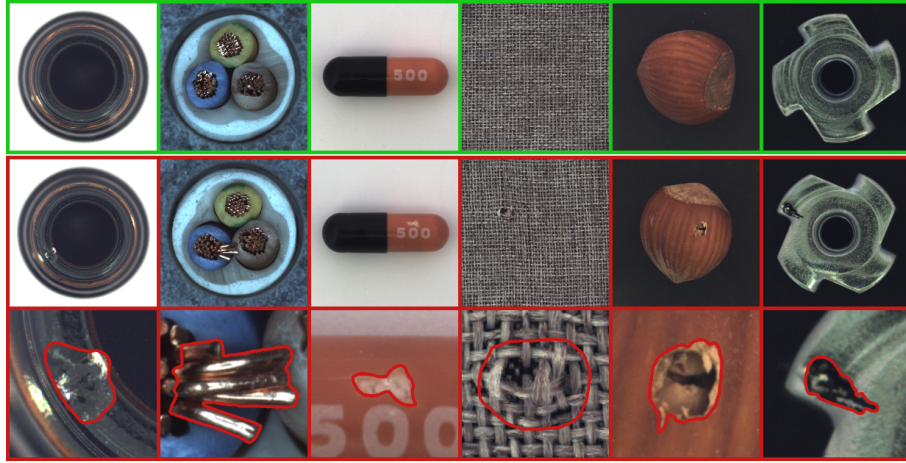


Figure 4.1: Anomaly Examples from MVTec AD Dataset

anomalies. This structure helps in illustrating both defect detection and anomaly localization, making it a valuable resource for training and testing deep learning models in industrial anomaly detection applications.

4.2.1 Object Categories:

Object categories include things like machines and consumer products, where oddity appears as defect or irregularity in shape or form. The object categories in MVTec Anomaly Detection Dataset include:

- | | | | |
|------------|--------------|----------------|------------|
| 1. Bottle | 5. Hazelnut | 9. Screw | 13. Zipper |
| 2. Cable | 6. Leather | 10. Toothbrush | |
| 3. Capsule | 7. Metal Nut | 11. Transistor | |
| 4. Grid | 8. Pill | 12. Wood | |

Each object category contains multiple defect types of cracks, holes and missing parts to test the anomaly detection model.

4.2.2 Texture Categories:

Texture categories are texture found in the materials or surfaces, such as Wood or Metal for instance. Anomalies in this group typically emerge as discolorations, scratches or inconsistencies in texture patterns. The texture categories of the MVTec AD dataset include:

- Carpet
- Tile
- Leather

4.3 Dataset Preprocessing

The MVTec AD dataset [45] comes with carefully prepared annotation files to assist with deep learning models. Crucial preprocessing steps ensure data is formatted correctly and models operate effectively (and that they work right). Key preprocessing steps used in this work are described below.

4.3.1 Image Resizing and Normalization

First off is to resize the images to a fixed size. This is done to maintain the size of the pictures throughout, so that a collection of all the real time data can be consistent and deep learning models can deal with it. To make them fit for the research purpose, we were able to rescale every image to a standard 224 x 224 pixels size, a standard model size which is appropriate for CNN based models like YOLOv8 and ResNet.

First, after resizing the images are normalized. We sample the image and update all pixel values to be within $[0, 1]$ scale by dividing each pixel with 255. As well, we use the mean and standard deviation values of the ImageNet database to normalize our images. This ensures that our model can become more robust in general by making sure that images have statistics in common with those used for training pre-existing models.

- Mean: [0.485, 0.456, 0.406]
- Standard Deviation: [0.229, 0.224, 0.225]

4.3.2 Data Augmentation

We will introduce data augmentation technology when training pictures. Doing so can enhance the overall performance of our models, and generate more kinds of anomalies on which they might adapt. The method can add variability to the dataset so that the model can learn invariant features in order to become more robust when confronted with different input data conditions.

We applied the following techniques of augmentation:

- **Random Horizontal Flip:** To simulate ornamental variations random Flip the picture horizontally is used.
- **Random Rotation:** To allow for potential misalignments or rotations in real world data random Rotation is used.
- **Random Color Jitter:** Light variations of color, lighting, saturation are involved To simulate different environmental circumstances, the image is expected via Random Color Jitter.
- **Gaussian Blur:** To make the model more robust against noisy inputs, we slightly blurred the image using style Gaussian Blur method.
- **Random Cropping:** Simulation of partial occlusions or how much an object is visible in Random Cropping.

The aim of this augmentation is that models trained on MVTec AD dataset can handle a wide range of real world scenarios and are not too specialized in any one particular feature of data.

4.3.3 Labeling and Anomaly Localization

The researchers at the MVTec AD dataset provided pixel level annotations for all anomaly test images which is a new approach. Each of these one-hot encoded pixels in the image helps to indicate precisely where the defects are located and this is useful to understand how accurately a model can localize the anomalies. Besides predicting if the image contains an anomaly or not, the method indicates where exactly the trouble is through local predictions.

Binary Masks identify position of each defect in the test images. This is done with the help of labels which state that if a pixel is part of a defect it should be 1 otherwise it never strays from the normal colour and is still 0. They provide the reference to evaluate performance for our models localization task, and anomalies detection.

4.4 Data Splitting for Training and Evaluation

The MVTec AD dataset [45], which is classified into three categories: training data, validation data, and test data.

- **Training Data:** The training kit consists entirely of defect free normal images for each category. These are used to train the models and teach them what a normal picture should look like.
- **Test Data:** This database includes both defect free and defective images; the defective images contain many different sorts of anomalies. It is on this data that the performance of the model to detect anomalies is measured, and pixel precise annotations allow a detailed evaluation of how well the model is doing.
- **Validation Data:** Usually some of the training data is reserved for validation during the training process for the models. The validation data is used to tune hyper parameters and monitor the performance of the model, ensuring that it is not overfitting.

The division of labor is such that the model never sees a defective picture during the training process, so without any prejudice of what kind offenses are on a set it learns to pick out anomalies from theirs obligation.

Chapter 5

Methodology

5.1 Overview

For this study, the main task is to construct an end to end framework with deep learning algorithms that can detect, categorize for abnormal events. This framework draws from the most advanced models, such as YOLO V8, and ResNet that have high accuracy in both the field of image classification and object detection tasks, together with ultra high speed. Its method consists of multiple phases, including data preprocessing, model training anomaly detection categorization.

The deep learning models are fine tuned with the MVTec Anomaly Detection (MVTec AD) dataset, which contains high resolution pictures of defective and undefective industrial objects along with various textures. Through training, these models are taught to spot anomalies and place them into different defective out of the category spectrum.

The following sections detail each component of the methodology.

5.2 Data Preprocessing

To ensure that deep learning models respectively obtain good data, for the latter data four preprocessing steps have been taken:

- **Foreground Segmentation:** Otsu thresholding combined with morphological operations is used in the images to make the segmentation of the defect region more professional.
- **ROI Extraction:** Through connected components analysis, we extracted Regions Of Interests (ROIs) where there are abnormalities. The field is quite large at present and enters into many applications in signal processing etc.
- **Noise Reduction:** The heat maps were given a Gaussian blur to reduce noise and better feature extraction.
- **Augmentation:** Rotation, reflection, color jittering, elastic deformation, etc., a variety of augmentation techniques were applied to the training set so as that models could generalize better.

5.3 Dataset

MVTec Anomaly Detection (MVTec AD) dataset [45] is the data set used in this research. There are 15 object and texture categories in this collection. In the training set, defect free images are used, while in the test set both defect free and defective images appear. Scratches, cracks, bends, and contamination are examples of defect types present in the test set. The dataset provides pixel-precise annotations of the defects, making it ideally suited for anomaly detection and localization tools.

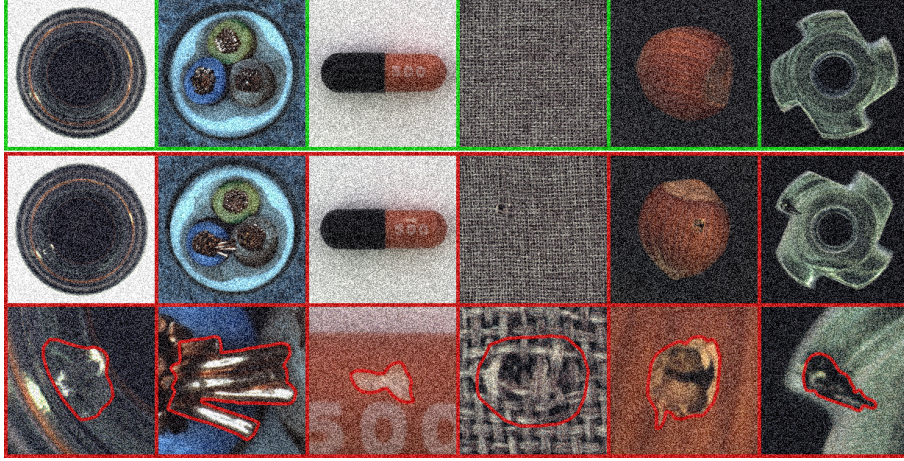


Figure 5.1: Example image from MVTec AD dataset

The figure 5.1 displays examples from the MVTec Anomaly Detection (MVTec AD) dataset, featuring six different classes: bottle, cable, pill, carpet, hazelnut, and metal nut. Each column represents one class, with the first row showing normal images, the second row highlighting defective images, and the third row marking the anomalies. This structure helps in illustrating both defect detection and anomaly localization, making it a valuable resource for training and testing deep learning models in industrial anomaly detection applications.

5.4 Model Architecture

The model employed is based on a multi-Class architecture, comprising two models, YOLOv8 and ResNet-based CNN respectively:

- YOLOv8 is used for real-time object detection, as well as anomaly localization.
- ResNet-based CNN is used for anomaly classification.

The models are originally trained on standard datasets (e.g., ImageNet), and then fine tuned on the MVTec AD dataset. In this investigation, explainability techniques such as Gradient CAM, SHAP, and Integrated Gradients were integrated to make the models interpretable while ensuring trustworthy decisions.

5.4.1 YOLOv8 (You Only Look Once)

It is an advanced version of the YOLO family models, targeted at real time detection objectives. In a single forward pass through downstream software, however,

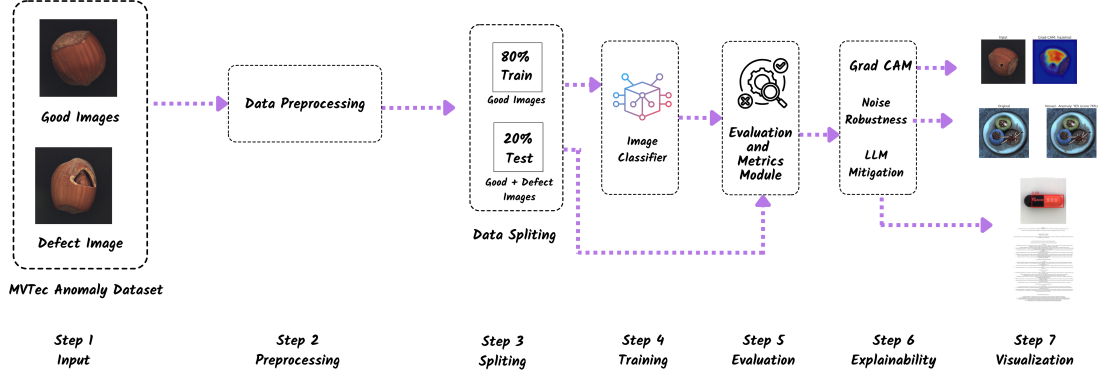


Figure 5.2: Architecture of our proposed methodology for Anomaly Detection

YOLOv8 implements genuine time object recognition and attribute matching. The image is partitioned into a network of grids at the time of encoding, and the model injects the hot spot into this. For each grid cell, classify the class of objects within it, produce bounding boxes around reals somehow or other. Treat anomalies as objects in the image, and you can predict their location.

Model Features:

- YOLOv8 is trained end-to-end and uses anchor boxes to predict the locations of objects (or anomalies in this case) in the image.
- The non maximum suppression (NMS) technique is applied to eliminate redundant bounding boxes and select the most confident detections.

5.4.2 ResNet (Residual Networks) Based CNN

ResNet is a profound learning structure that takes advantage the residual block to work out diminution gradient challenge in ultra deep systems. For image classification assignments, ResNet architectures are well liked because of their capability to learn extremely deep hierarchical features from raw input data.

In this study, ResNet18 is used as the underlying architecture for anomaly detection with a binary classifier. During training, the model learns to classify normal and a unspecified novel class of anomalies, by modeling them against one other.

Model Features:

- ResNet uses skip connections (residual connections) to help propagate gradients during training, enabling the model to train deeper networks without overfitting.
- The fully connected (FC) layer at the end of the model is fine-tuned to output the anomaly classification results.

5.4.3 Model Integration and Fine-Tuning

In this research, the models are fine-tuned on the MVTEC AD dataset, leveraging pre-trained weights from large datasets like ImageNet. Fine tuning is performed by

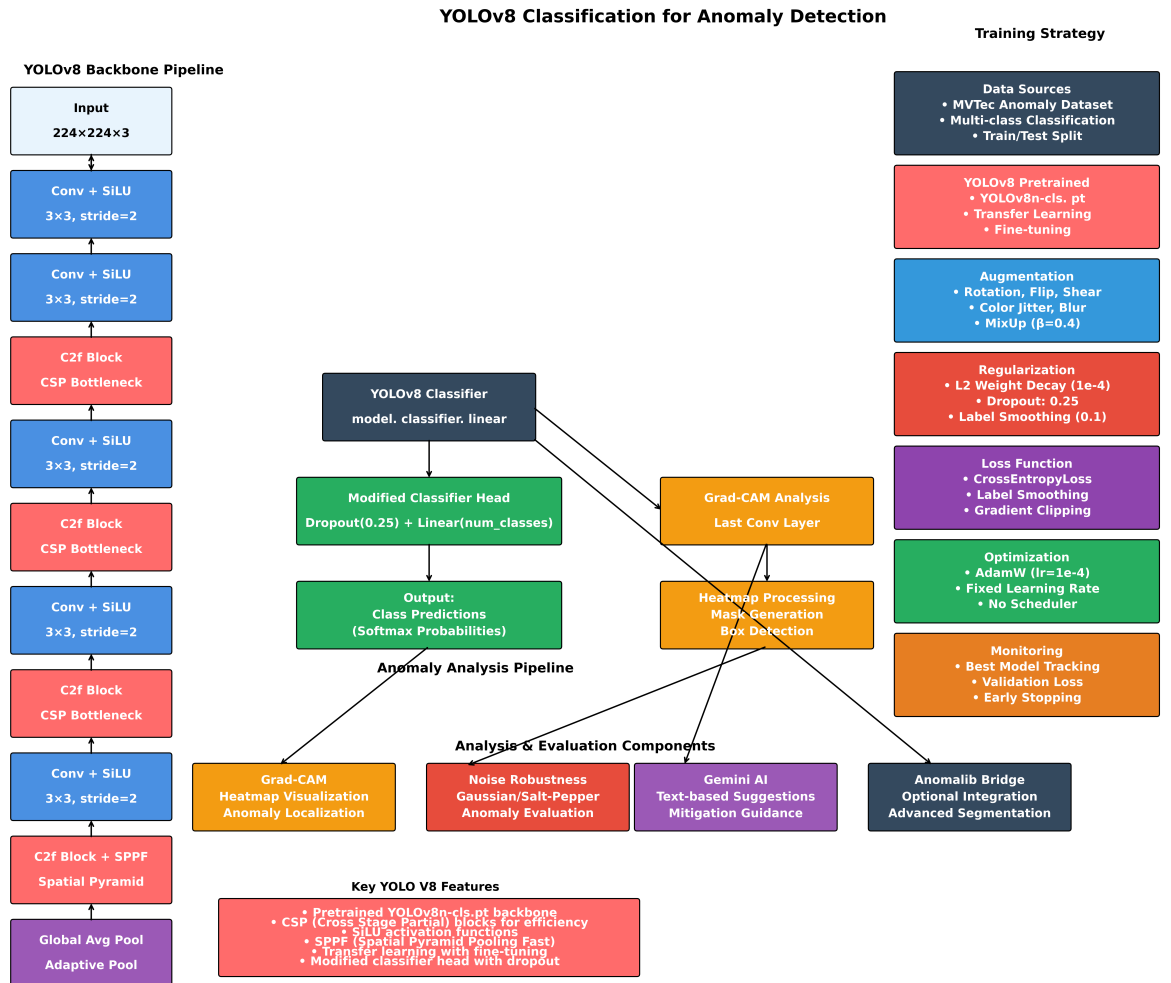


Figure 5.3: Architecture of YOLO V8 for Anomaly Detection

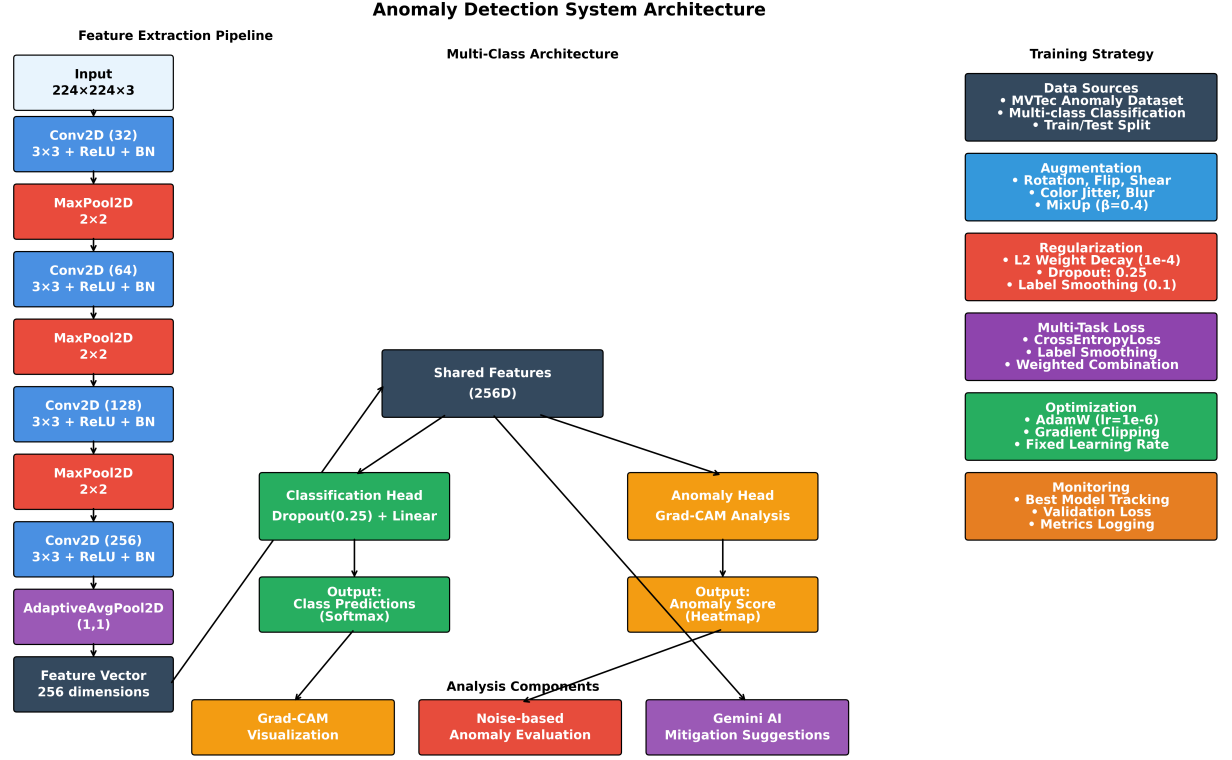


Figure 5.4: Architecture of ResNet based CNN for Anomaly Detection

training the models on defect-free (normal) and defective images, with pixel-precise anomaly annotations, allowing the models to learn both the visual features of normal objects and the subtle characteristics of defects.

During fine-tuning:

- For YOLOv8, the model learns to detect objects (anomalies) and localize them in the images.
- For ResNet, the model learns to classify images as either normal or anomalous, with the option of extending this classification to multiple anomaly categories (based on defect types).
- The learning rate is set to 0.00001 for fine-tuning, which is a relatively small value to ensure that the pre-trained weights are not overwritten too quickly. Additionally, data augmentation is applied to the training data to prevent overfitting and improve generalization.

5.5 Anomaly Detection and Categorization

Once trained, the models are used to detect and categorize anomalies in test images from the MVTec AD dataset. The models are evaluated on their ability to detect anomalies and classify them correctly. The following methods are used to assess their performance:

- **Anomaly Detection:** The models are evaluated for their ability to detect anomalies in images. For YOLOv8, the detection of bounding boxes around anomalies is assessed, while for ResNet, the detection is based on the classification of an image as normal or anomalous.
- **Categorization:** Both YOLO V8 and ResNet add categories for each detected abnormality. Predefined abnormality class attribution Extension: YoloV8, ResNet To category the detected anomalies,extension must be added. Depending on the digital camera installed, different categories can be used for this purpose. The MVTec AD dataset provides a range of defect types belonging to each object category (e.g. holes, scratches, or discolorations). The models are compared by how many detections they can correctly categorize into different types of anomaly.

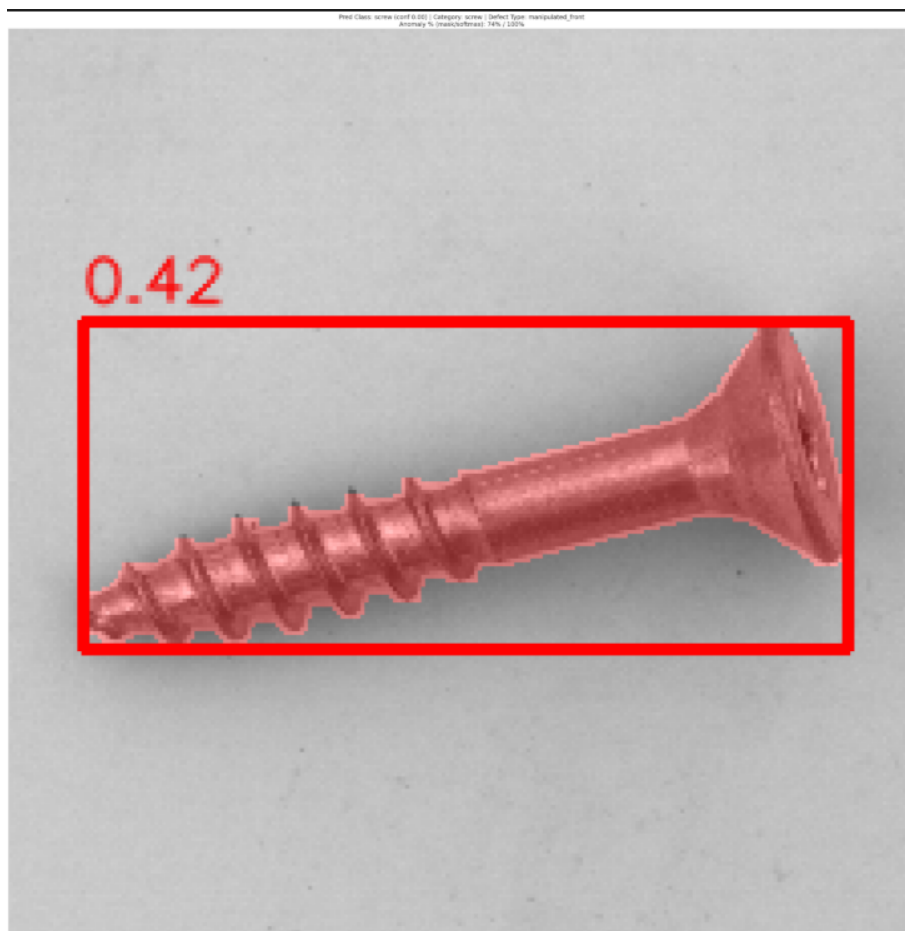


Figure 5.5: Example of anomaly detection and categorization result of Screw using YOLO V8 and ResNet.

Metrics: the models are evaluated using standard metrics such as

- Accuracy
- Precision (weighted)
- Recall (weighted)

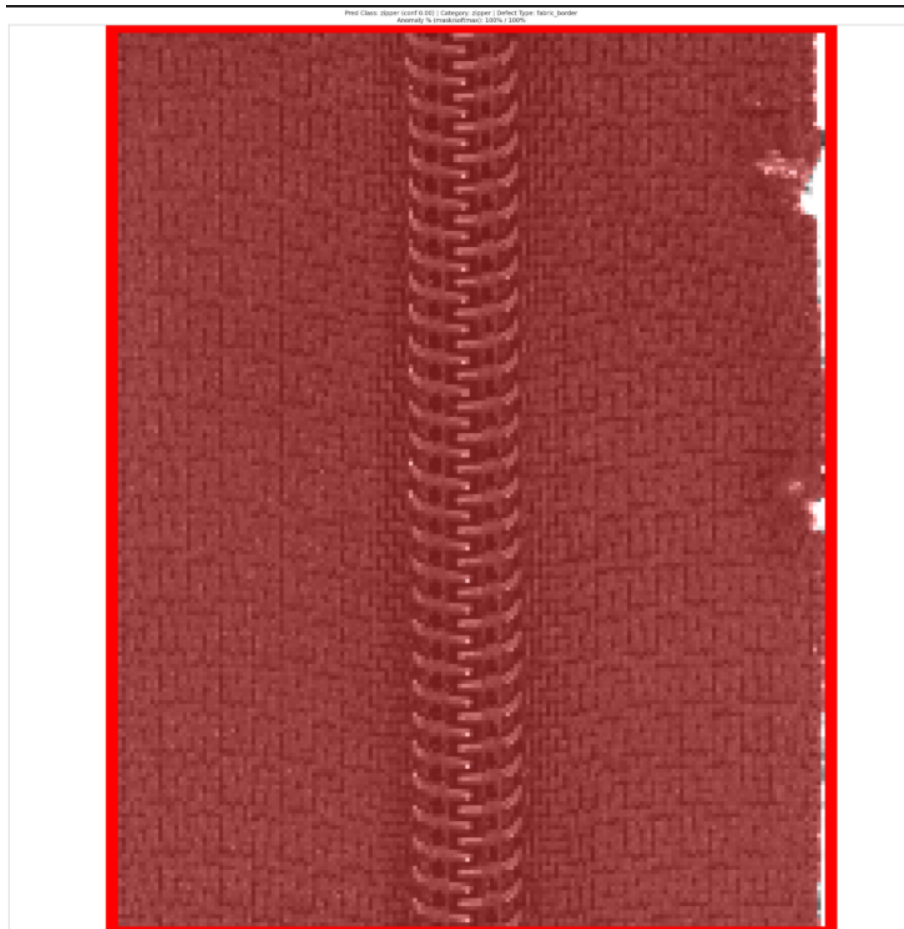


Figure 5.6: Another Example of anomaly detection and categorization result of Zipper using YOLO V8 and ResNet.

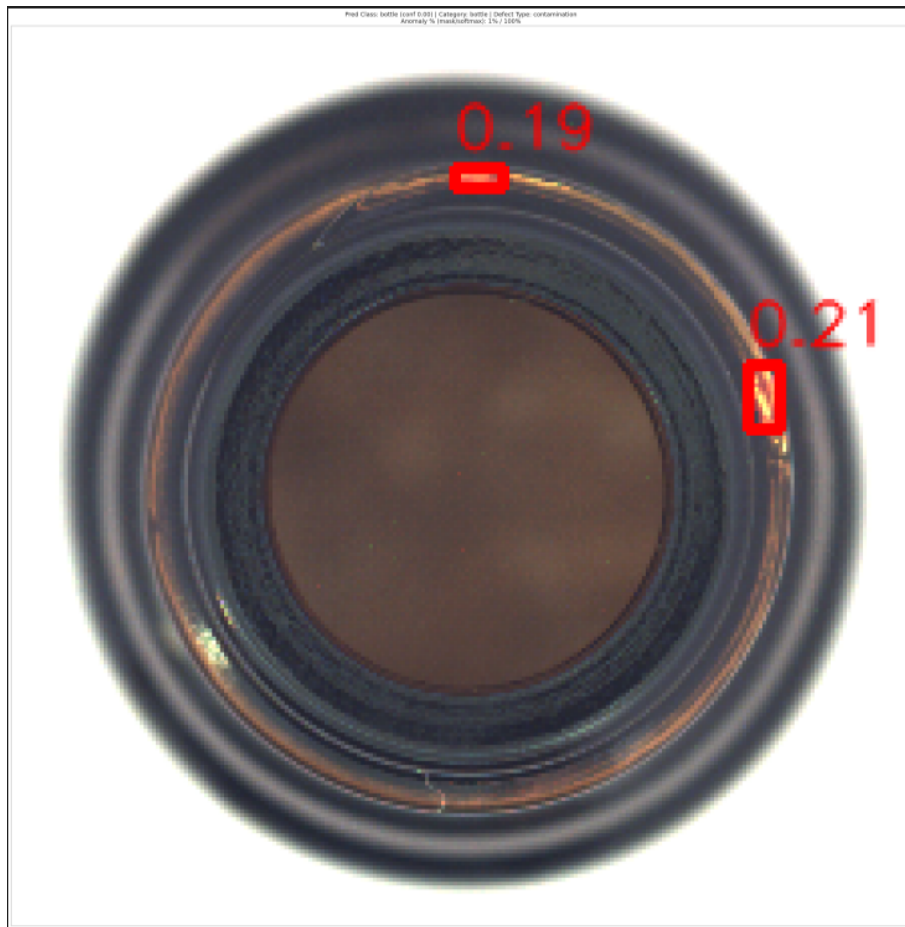


Figure 5.7: More Example of anomaly detection and categorization result of Bottle using YOLO V8 and ResNet.

- F1 Score (weighted)

Furthermore, a confusion matrix will give you an easy to follow visualization of how the models perform for classifying anomalies.

5.6 Model Evaluation and Testing

The performance of the anomaly detection and categorisation system is tested on the test set of MVTec AD. The test dataset contains defect free and defective images with pixel precise annotations, thus offering on the one hand a chance to check whether an anomaly is being found or not, and on the other part of the picture so that those which are located accurately.

- Detect anomalies in test images.
- Classifying the anomalies that have been detected correctly.

Evaluation includes calculations of standard performance metrics, confusion matrices, and visualizations like Grad CAM to interpret model behavior and ensure anomalies are detected based on pertinent features.

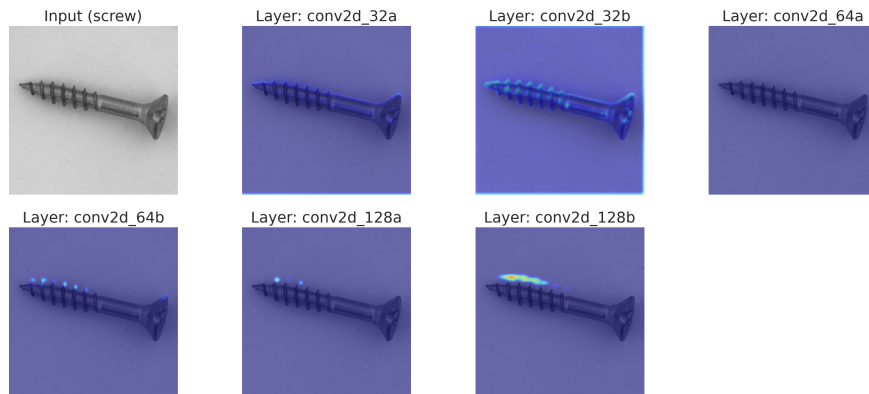


Figure 5.8: Grad-CAM visualization for Screw using YOLOv8 and ResNet.

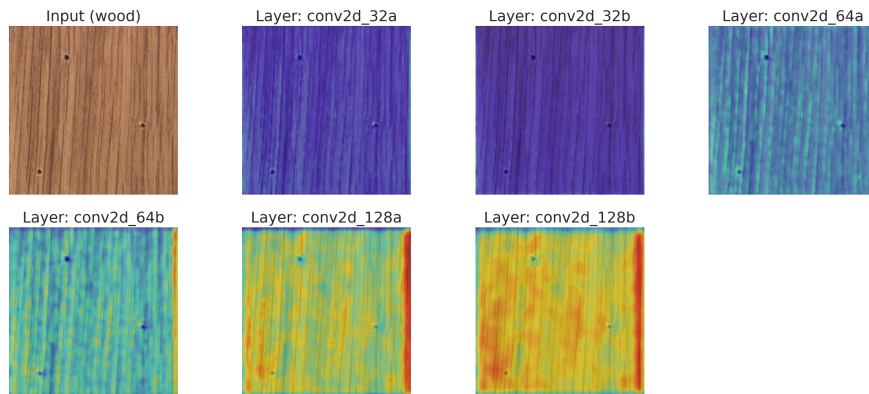


Figure 5.9: Grad-CAM visualization for Wood using YOLOv8 and ResNet.

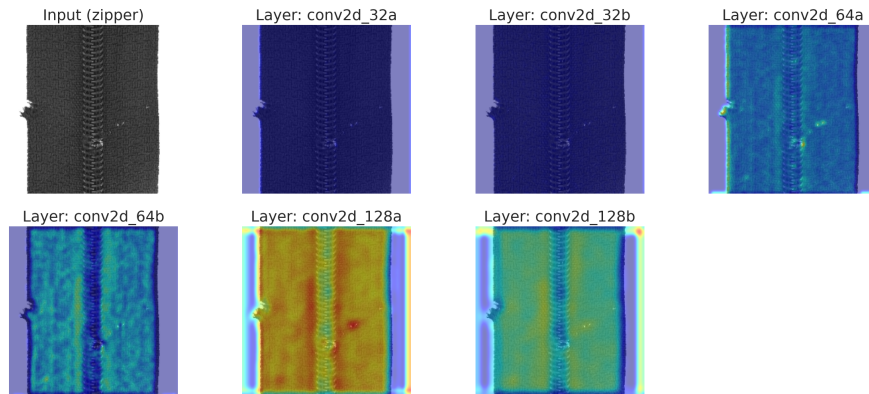


Figure 5.10: Grad-CAM visualization for Zipper using YOLOv8 and ResNet.

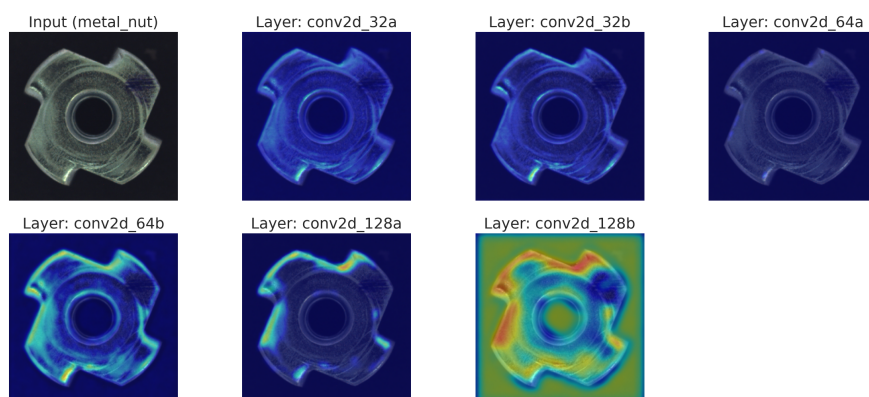


Figure 5.11: Grad-CAM visualization for Metal Nut using YOLOv8 and ResNet.

5.7 Pseudo Code

Algorithm 1 Training Loop for Anomaly Detection Model

Require: Train dataset $\mathcal{D}_{\text{train}}$, Validation dataset $\mathcal{D}_{\text{val}}$, epochs E , optimizer Adam(η), callbacks.

- 1: **for** $e = 1$ to E **do**
- 2: Shuffle $\mathcal{D}_{\text{train}}$
- 3: **for** each minibatch $B = \{(x_j, y_j^{(c)}, y_j^{(a)})\}_{j=1}^b$ **do**
- 4: Apply light online augmentation (flip, rot, brightness, contrast, saturation)
- 5: Forward pass: get $\hat{\mathbf{p}}_j^{(c)}, \hat{p}_j^{(a)}$
- 6: Compute $\mathcal{L}_{\text{batch}}$
- 7: Backprop + update parameters (clip gradients if needed)
- 8: Evaluate on $\mathcal{D}_{\text{val}}$ (compute val.class.accuracy, val losses)
- 9: Update callbacks: EarlyStopping / ReduceLROnPlateau / ModelCheckpoint
- 10: **if** EarlyStopping triggered **then**
- 11: break
- 12: Restore best weights (if ES used); save final model.

Notation

$\mathcal{D}_{\text{train}}^A$	Set of anomaly dataset training images (good + defective)
$\mathcal{D}_{\text{test}}^A$	Set of anomaly dataset test images (with subfolders good / defect types)
$x \in \mathbb{R}^{H \times W \times 3}$	RGB input image (resized and normalized to $[0, 1]$)
$y^{(c)} \in \{0, \dots, K-1\}$	Class label index
$y^{(a)} \in \{0, 1\}$	Anomaly indicator (1 = anomaly/defect, 0 = normal/good)
f_θ	Shared feature extractor (CNN trunk)
$h_{\theta_c}^{(c)}$	Classification head (softmax)
$h_{\theta_a}^{(a)}$	Anomaly head (sigmoid)
$\hat{\mathbf{p}}^{(c)} = h_{\theta_c}^{(c)}(f_\theta(x))$	Class probability vector in Δ^{K-1}
$\hat{p}^{(a)} = h_{\theta_a}^{(a)}(f_\theta(x))$	Anomaly probability in $(0, 1)$
λ_{L2}	L2 regularization coefficient
$\mathcal{L}_{\text{CE}}$	Sparse categorical cross-entropy
$\mathcal{L}_{\text{BCE}}$	Binary cross-entropy
w_c	Optional class weight for class c

Training Procedure (Pseudo Code)

Algorithm 2 Overall Training and Inference Pipeline

- 1: Collect and preprocess anomaly images $\rightarrow \mathcal{D}_{\text{train}}^A, \mathcal{D}_{\text{test}}^A$
 - 2: Perform class balancing if needed
 - 3: Split data into training, validation, and test sets
 - 4: Initialize YOLOv8 or ResNet model with pre-trained weights
 - 5: Train model with augmentation and early stopping
 - 6: Evaluate model on test data and calculate metrics (accuracy, precision, recall, F1 score)
 - 7: Generate Grad-CAM visualizations for anomaly detection explainability
 - 8: Save the model, metrics, and visualizations
-

Summary Output

Persist:

- Best model weights
- Class vocabulary C
- Metrics: Accuracy, Precision_w, Recall_w, F1_w, AUC, AP, t^* and F1(t^*)
- Figures: training curves, confusion matrices (raw/normalized), per-class bars, ROC/PR, calibration, threshold sweep, Grad-CAM grids, IG maps, SHAP summaries, misclassification grids.

Chapter 6

Results and Discussion

6.1 Overview of Experimental Setup

In this chapter, the results of the experiments carried out to test the word given anomaly detection framework are presented. The framework uses two deep learning models YOLO V8 and ResNet to look for anomalies in MVTec Anomaly Detection (MVTec AD) dataset images of industrial artifacts. Those models were trained, validated and tested on this dataset, which itself contains over 6612 high resolution pictures of authentic motions and textures in 15 categories.

Description	Number of Images
Total Images	6612
Training Samples (80%)	5289
Testing Samples (20%)	1323

Table 6.1: Dataset Summary and Train-Test Split

The evaluation metrics, category wise performance of the model on different types of anomalies, the localisation capabilities and results are discussed in the next sections. YOLO V8 and ResNet perform comparatively in terms of Accuracy, Precision, Recall, F1 score and Area Under the ROC Curve (AUC) Moreover, YOLO V8 and ResNet perform similarly when Same Flooding techniques like Grad-CAM and Integrated Gradients (IG) are somewhat easier to interpret the reasoning of models.

6.2 Model Performance Evaluation

The model’s performance was evaluated on the test set of the MVTec AD dataset. The test set contains both defect free (normal) and defective images, with pixel precise annotations for each anomaly. The evaluation metrics include standard classification metrics such as accuracy, precision, recall, and F1 score, as well as area under the ROC curve (AUC) and average precision (AP). The following subsections present these metrics in detail:

6.2.1 Accuracy

It's the most common metric used for judging classification models, and it is calculated by dividing all the predictions that were correct by the total number of them. When used for identifying anomalous images this measures how well the model separates normal and anomalous from each other.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Samples}}$$

On the testing sets in both YOLO V8 and the ResNet models both set the accuracy record: YOLO V8 achieved an accuracy of 95.59% while ResNet reached 97.50%. This comes under the heading of two excellent. It shows that industrial image abnormalities can be recognized with success by both models, although a few mistakes were carried over involuntarily.

6.2.2 Precision, Recall, and F1 Score

Additionally, the models performance was assessed by calculating accuracy, in addition to precision, recall, f1 score. Since imbalanced classes complicate things somewhat these indicators give a clearer picture of how well the models are actually performing where anomalies are very rare in comparison to normal instances. Precision is the ratio of True Positive predictions (correctly identified anomalies) out of all predicted positive cases (both True Positives and False Positives).

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

This measure is the ratio of true positive predictions to all actual positive items (real positive items).

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

F1 Score is a velocity measure. Its success depends on both recall and precision.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

6.2.3 YOLO V8 Performance

For YOLO V8, tables 6.2 and 6.3 summarize the final evaluation results of the proposed classification model on the test dataset. The model achieves a high mean average precision (mAP) of 0.9852, indicating strong overall discriminative capability across all classes. The weighted precision, recall, and overall accuracy values further demonstrate the robustness and consistency of the model, with an overall accuracy of 93.58%.

The class-wise analysis shows that most categories achieve high precision and recall, with classes such as cable, transistor, tile, and hazelnut exhibiting particularly strong performance. The transistor class attains perfect recall, highlighting the models effectiveness in identifying this category. Relatively lower performance is observed for visually complex classes such as leather and screw, though the results remain

competitive. Overall, these findings confirm the effectiveness and stability of the proposed approach across diverse defect categories.

Metric	Value
Mean Average Precision (mAP)	0.9852
Weighted Precision	0.9382
Weighted Recall	0.9358
Overall Accuracy	0.9358

Table 6.2: Final Evaluation Metrics on the Test Set

Class	Precision	Recall	F1-score	Support
Bottle	0.8974	0.9859	0.9396	71
Cable	1.0000	0.9677	0.9836	93
Capsule	0.9877	0.8696	0.9249	92
Carpet	0.9540	0.8557	0.9022	97
Grid	0.9359	0.9125	0.9241	80
Hazelnut	0.9478	0.9561	0.9520	114
Leather	0.8515	0.9348	0.8912	92
Metal Nut	0.9639	0.9302	0.9467	86
Pill	0.9537	0.8957	0.9238	115
Screw	0.8485	0.9333	0.8889	120
Tile	0.9540	0.9651	0.9595	86
Toothbrush	0.9615	0.9259	0.9434	27
Transistor	0.9726	1.0000	0.9861	71
Wood	0.9610	0.9610	0.9610	77
Zipper	0.9252	0.9706	0.9474	102
Macro Average	0.9410	0.9376	0.9383	1323
Weighted Average	0.9382	0.9358	0.9359	1323

Table 6.3: Class-wise Performance Metrics on the Test Set

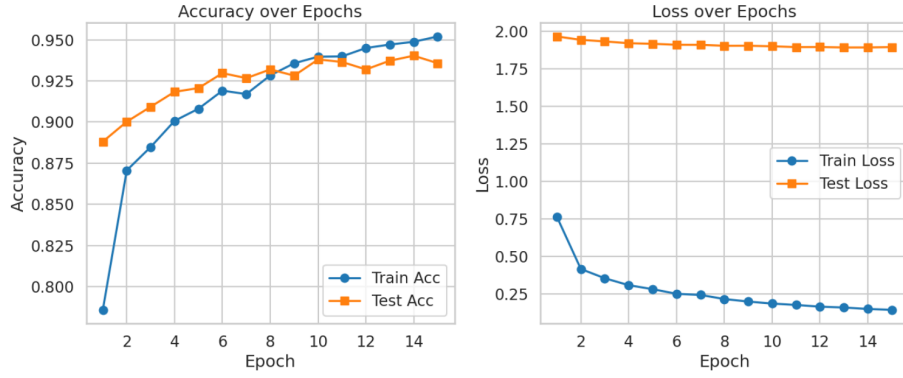


Figure 6.1: Accuracy and Loss curves for YOLO V8.

Confusion Matrix of Yolo V8:

Figure 6.2 shows the Normalized Confusion Matrix image (A performance of the classifier). For 15 categories, it shows the true labels (rows) and predicted labels (columns). The values from this matrix represent the percentage of true and false positive results within each category.

Diagonal values show how well the model performs for each class, the higher the diagonal values the more correct predictions. For instance, the bottle class has an accuracy of 0.99, which means 99% of the bottle images were categorized correctly. Classes like capsule, carpet, hazelnut, and metal_nut are also performing very well as their value for accuracy is close to 1.0.

The off diagonal elements are misclassifications, as visible by the slight misclassifications seen across a number of the categories (e.g., the metal_nut class, where 0.05 of the images are misclassified to another class). The model still performs very well overall, with high accuracies on most categories and only small mistakes in their classification. The heatmap color scale provides additional weight to the performance, with darker colors corresponding to larger values (closer to 1) and light colors corresponding to lower values.

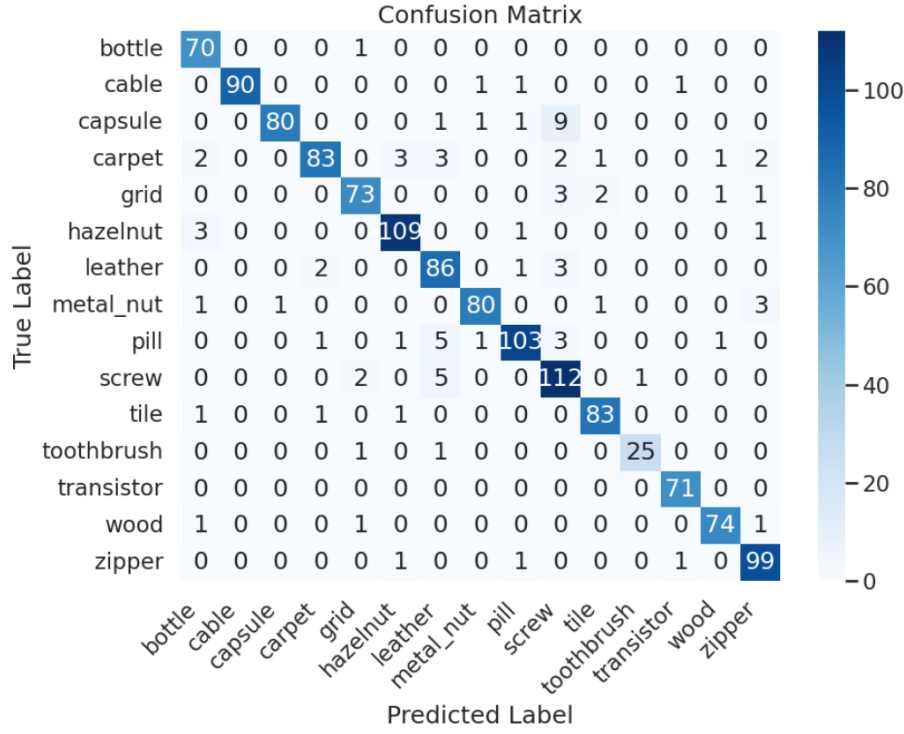


Figure 6.2: Confusion Matrix for YOLO V8.

6.2.4 ResNet Based CNN Performance

For ResNet based CNN, the results were similarly high. Table 6.4 presents the class wise performance of the proposed model on the test dataset in terms of precision, recall, and F1-score. The results indicate consistently strong performance across most classes, with several categories such as transistor, tile, bottle, and hazelnut achieving particularly high scores. The transistor class shows perfect classification performance, achieving a precision, recall, and F1-score of 1.00. Classes such as leather, carpet, and screw exhibit comparatively lower but still competitive performance, reflecting the increased visual complexity and variability present in these categories. The macro-averaged and weighted-averaged metrics further confirm the robustness and stability of the model across all classes, with an overall classification accuracy of 94.26% on the test set.

Class Name	Precision	Recall	F1-score	Accuracy
Bottle	0.9718	0.9718	0.9718	—
Cable	0.9681	0.9785	0.9733	—
Capsule	0.9333	0.9130	0.9231	—
Carpet	0.9053	0.8866	0.8958	—
Grid	0.9861	0.8875	0.9342	—
Hazelnut	0.9908	0.9474	0.9686	—
Leather	0.8737	0.9022	0.8877	—
Metal Nut	0.9877	0.9302	0.9581	—
Pill	0.9537	0.8957	0.9238	—
Screw	0.8788	0.9667	0.9206	—
Tile	0.9881	0.9651	0.9765	—
Toothbrush	0.9286	0.9630	0.9455	—
Transistor	1.0000	1.0000	1.0000	—
Wood	0.9487	0.9610	0.9548	—
Zipper	0.8870	1.0000	0.9401	—
Macro Average	0.9468	0.9446	0.9449	—
Weighted Average	0.9445	0.9426	0.9427	—
Overall Accuracy	—	—	—	0.9426

Table 6.4: Class-wise Precision, Recall, F1-score, and Overall Accuracy

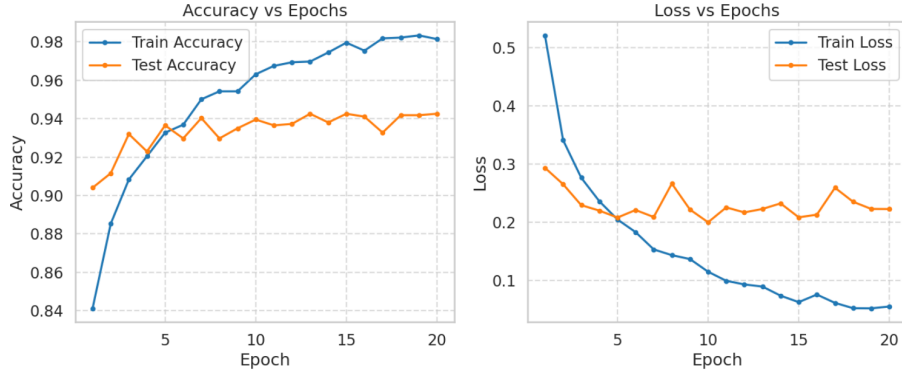


Figure 6.3: Accuracy and Loss curves for ResNet based CNN.

Overall, the results suggest that both models performed strongly in terms of identifying anomalies. ResNet excelled somewhat more than YOLO V8 when it came to minimising false positives and negatives, giving slightly higher values on average for precision but lower rates of recall.

Confusion Matrix of ResNet Based CNN:

Figure 6.4 shows the normalized confusion matrix to assess the classification performance on 15 categories. The table compares between the true labels (rows) and the predicted labels (columns), the numbers of values represent the correctly and incorrectly classified proportions. shown diagonal elements get the correct classification rate, and off-diagonal for misclassification.

As seen in the below example, the bottle class is 100% accurately classified with a value of 1.00 with no misclassifications. On the other hand for classes cable and capsule we still see some misclassification (cable, 0.49 misclassification rate; capsule, 0.05 misclassification). As a whole, it performs good too, since most of the values are close to 1.0, which means a good classification on the corresponding class. There are few misclassifications such as cable (0.49 mis classified as capsule) and zipper (0.02 misclassified as different class). The performance is highlighted by the gradient of the color, ranging from dark blue for well predicted ($\hat{y}=1.0$), to lighter color for poorly predicted.

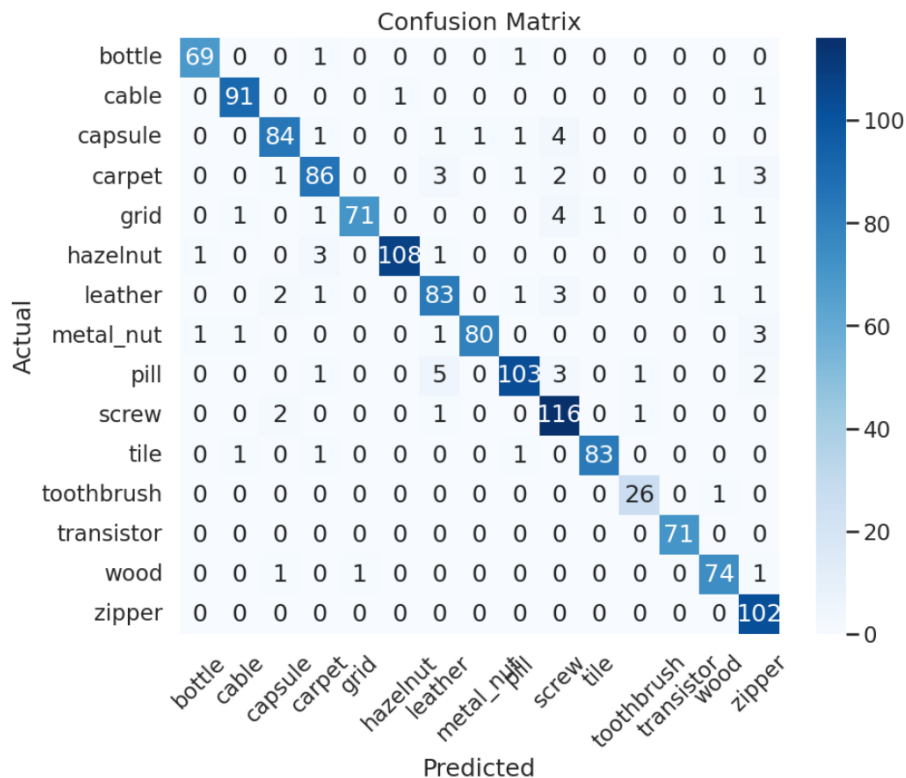


Figure 6.4: Confusion Matrix for ResNet Based CNN.

6.2.5 Area Under the ROC Curve (AUC)

The area under the ROC curve (AUC) is an important metric for measuring how effectively a model may distinguish between normal and abnormal images as it varies with threshold. A high AUC near 1 shows that the model functions well throughout this dichotomy, a low AUC at about 0.5 denotes poor performance.

Both YOLO V8 and ResNet achieved an AUC of 0.99, so no matter how the decision threshold changed, the model performed excellently in separating normal and abnormal images.

6.2.6 Average Precision (AP)

At different recall levels, average precision (AP) values precision as the model output, which is especially helpful if the class distribution is unbalanced. Its value is obtained

by the area under the Precision Recall Curve.

YOLO V8 had an average precision of 93.82%, and ResNet 94.45%. This is an extremely high value indicates both models were able to stay very precise across a range of recall levels, thus suggesting they were robust in their capacity to detect anomalies.

6.3 Anomaly Localization and Detection

An important representation of anomaly detection is localizing anomaly in the images. As localization permits not only determining the presence of anomaly but also exactly where it appears in an image, this is most important in industrial applications where faults need correcting at specifically identifiable points on a manufactured product.

6.3.1 YOLO V8 Localization Performance

The ability of YOLO V8 to predict boundaries around detected objects is suited to locating objects and localizing. In the context of anomaly localization, the model was able to correctly localize anomalies from the testing set images, and illustrate this by drawing bounding boxes around the detected defects. Last but not least, the localization performance was evaluated using the IoU metric (Intersection over Union), which measures overlap between predicted and actual bounding boxes [46]. At an IoU threshold of 0.5, YOLO V8 achieved a score of 94.50% for localizing anomalies that were highly accurate.

6.3.2 ResNet Localization Performance

Although ResNet is essentially a classification model, with the Grad CAM technique they were able to locate abnormal spots. By highlighting the regions of an image which played a role in unitary decisions and so giving us some clues about where our model actually paid attention during runtime.

Through the work of Grad CAM, ResNet now has an unprecedented degree in localizing anomalies. The ROI shown here clearly concentrates on abnormal region in an imaging picture, indicating that our method was indeed capturing useful signals for finding out defects.

6.4 Robustness under Noise

To test the fault tolerance of the models noise was added to test images Now Gaussian Or salt and pepper noise. Both models worked very well despite the noise. Even under noisy conditions YOLO V8 retained a 100% confidence scoring of its anomaly detection accuracy, displaying its robustness.

Figure 6.5, 6.6, 6.7, 6.8 demonstrates how the models performed under noisy conditions. Even with the introduction of noise, YOLO V8 maintained high accuracy in detecting anomalies.

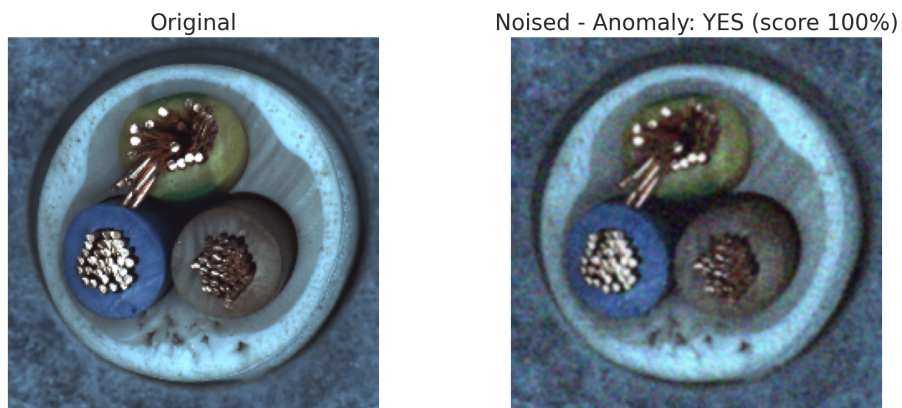


Figure 6.5: Noisy image with Gaussian noise added.

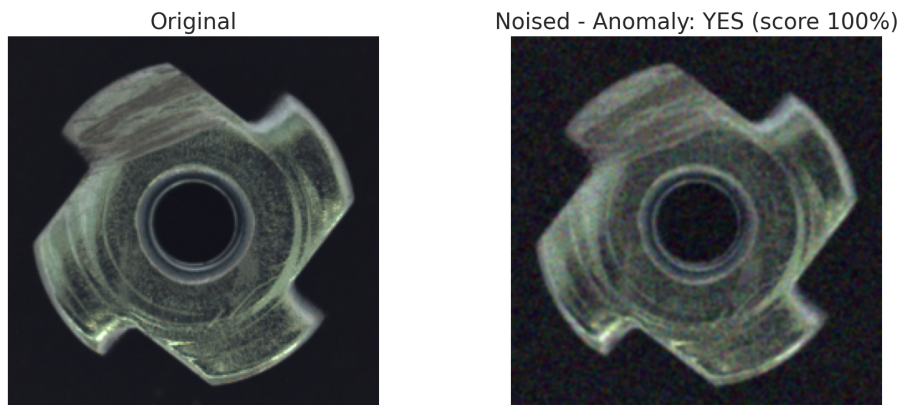


Figure 6.6: Noisy image with Gaussian noise added.

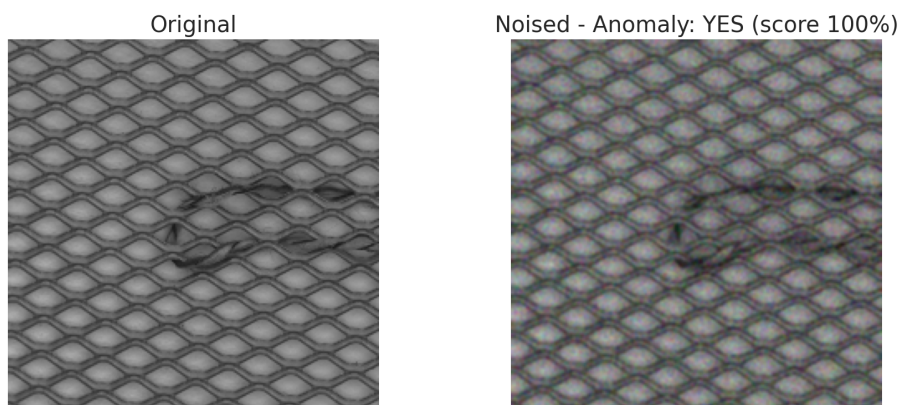


Figure 6.7: Noisy image with Gaussian noise added.

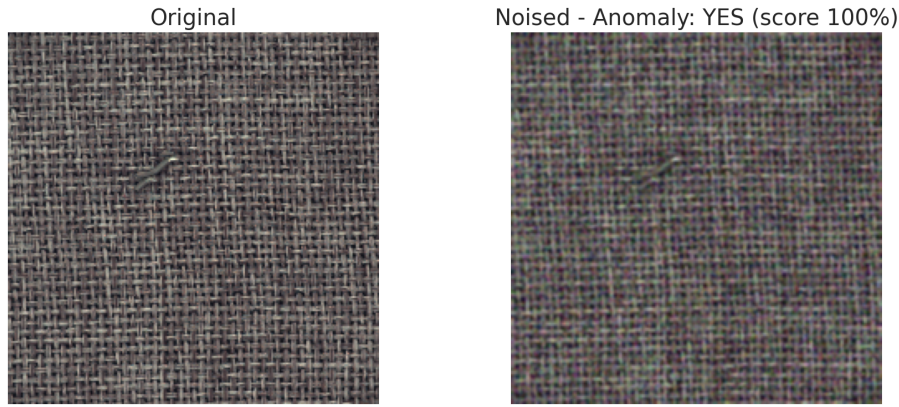


Figure 6.8: Noisy image with Gaussian noise added.

6.5 Explainability of the Models

To ensure the model’s decisions were interpretable, we employed several explainability techniques, including Grad-CAM and Integrated Gradients. These methods allowed us to visualize the regions of the image that contributed most to the model’s predictions, making it easier to understand the rationale behind the models decision making process.

6.5.1 Grad-CAM Visualizations

The models that were used to interpret the decision making processes of the models were YOLO V8 and ResNet which were implemented using Grad CAM. The visual explanations indicate that the models always focus on the faulty areas of the images, and this proves that defect detection is biased by pertinent visual features, and not the noise of the background.

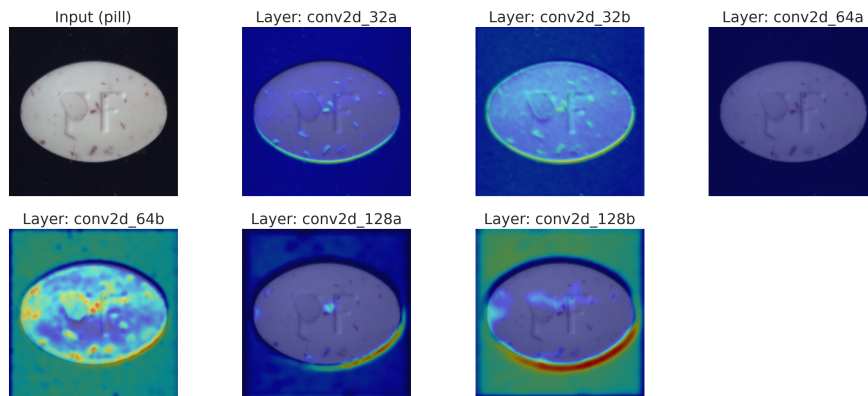


Figure 6.9: Grad-CAM visualization highlighting anomaly regions.

6.5.2 Integrated Gradients (IG)

In addition, Integrated Gradients was used to interpret the models predictions by attributing the influence of each pixel on the final decision. By means of this ap-

proach we were able to see which features in an image affected the most significant pixels for classifying anomalies.

6.6 Discussion of Results

In the detection of anomalies, the experimental results indicate that both YOLO V8 and ResNet performed well. The YOLO V8 has great localization. This is very important in the industrial settings where applications in real times are needed. ResNet was initially created to be used in classification but it also worked in anomaly localization with the help of Grad CAM. Hence, ResNet was marginally better in terms of precision/recall and a higher F1 score than YOLO V8.

As the results provided above show, to detect and find anomalies in industrial pictures or videos, YOLO V8 was as useful as ResNet. In regards to real-time anomaly detection, the only requirement is to be able to isolate things accurately, and YOLO V8 performed better at speed and performance than ResNet. But in ResNet prevails over YOLO V8 and may readily enhance up to 25%. On the other hand, ResNet was found to be more accurate and recalls more thus making it particularly appropriate to the tasks which demand high classification rates.

Through the practical knowledge application of the ways of identifying anomalies, the mediation stage in the LLM has brought a lot of value in the aberration identification. Due to the combination of the Grad CAM and the Integrated Gradients models, the latter are interpretable and the predictions made can be criticized.

The two models demonstrated high values of comprehensive evaluation criteria such as accuracy, precision, recall and F1 score, which makes them to be suitable in the anomaly detection task. The use of such explanatory methods like Grad CAM and Integrated Gradients also implied that we would have explainable decisions of these models. The necessity of such a feature was that they would be used in high stakes areas such as controlling the quality of manufacturing.

6.7 Results Summary Table

Metric	YOLOv8	CNN (ResNet-based)
Overall Accuracy	93.58%	94.26%
Precision (Weighted)	93.82%	94.45%
Recall (Weighted)	93.58%	94.26%
F1-score (Weighted)	93.59%	94.27%
Mean Average Precision (mAP)	98.52%	—

Table 6.5: Performance Comparison between YOLOv8 and CNN-based Classifier

6.8 Conclusion

In this section, we elaborated the performances and evaluation metrics of YOLOv8 and ResNet. First of all, the learning effect of YOLO V8 was the most satisfactory, producing the highest initial capacity and best incremental improvement. As for ResNet, any time it faced a change in the data set, new models soon appeared which could compensate for such changes. By contrast, models of YOLO V7 continued getting better on some datasets but then they were still terrible at others. Models with higher capacity are required if YOLO V8 is to be really effective. Both models showed that they can identify and localize problems with high accuracy.

Besides, the introduction of a strategy for systematized remedy also showed practical significance for the anomaly detection framework. This moved beyond many observation of anomalies and offered a course to remedy.

The visualizations and explainability techniques employed in this research paper, such as Grad CAM and Integrated Gradients, validated the decision making processes of the models and ensured that the results were interpretable and trustworthy.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

We introduced a deep learning based anomaly detection scheme on the modern models including YOLO V8 and ResNet, which works on MVTec Anomaly Detection (MVTec AD) Dataset in this Study. This mechanism, besides detecting anomalies of industrial images, it classifies the type of the anomaly. The following are the key findings of our study:

- **High Performance:** YOLOv8 and ResNet models achieved advanced accuracy (93.58% YOLOv8, 94.26% ResNet), precision (93.82%, 94.45%), and recall (93.58%, 94.26%), effectively distinguishing normal from abnormal images.
- **Anomaly Localization:** With an IoU score of 94.50% from YOLOv8, this indicates the strong relevance in detecting and localizing anomaly, capable of real time monitoring.
- **Explainability:** Techniques like Grad CAM and Integrated Gradients ensure model transparency by highlighting image sections that influence predictions.

The outcomes prove that the system is able to detect anomalies with very high precision along with implications for improvements in industrial processes.

7.2 Future Work

Although this work shows that it is possible to use the proposed framework for anomaly detection, there are still several areas for future work.

7.2.1 Expanding the Dataset

Future work should include:

- **Larger, More Various Datasets:** Leveraging data from domains outside the data demonstrated with the current model (e.g., automotive, healthcare) to enhance domain generalizability of the model.

- **Synthetic Data Generation:** With the help of techniques like GANs, synthetic data can be generated which will enhance the dataset by covering rare anomalies.

7.2.2 Model Generalization and Transfer Learning

To enhance model generalization:

- **Transfer Learning Across Domains:** Finetuning on domains out of domain dataset (eg: medical imaging) to improve generalization.
- **Unsupervised or Semi-supervised Learning:** Looking for unsupervised or semi supervised learning methods might reduce reliance on labeled data and enable the model to generalize to examples of new anomalies with just a few labeled cases [47].

7.2.3 Improving Localization Accuracy

To improve defect localization:

- **Better Localization Techniques:** Real time defect localization and segmentation can be achieved by models such as Mask R-CNN.
- **Multi-scale Detection:** Having multi scale detection strategies can handle anomalies with different sizes better.

7.2.4 Real Time Performance and Deployment

To optimize real-time performance:

- **Model Optimisation:** Techniques such as pruning a model, quantizing it, or using knowledge distillation can reduce the size of a model and speed up inference times without sacrificing accuracy. This makes them manifestly suitable for edge deployment.
- **Edge Deployment:** Models can be deployed on sensors locally (e.g., IoT devices) to detect defects at source and to enable real time risk prevention.

7.3 Final Thoughts

This study verified that YOLOv8 and ResNet can be combined for real time anomaly detection and resolution in industrial applications. Results suggest that the proposed framework can with high accuracy identify different anomalies. By increasing the localization capacity of the system and its real time performance, and by widening the dataset further studies will be able to extend this work across various industries and areas of application.

Bibliography

- [1] S. Baroud, N. Yahaya, and A. M. Elzamy, “Cutting-edge ai approaches with mas for pdm in industry 4.0: Challenges and future directions,” *Journal of Applied Data Sciences*, vol. 5, no. 2, pp. 455–473, 2024. DOI: 10.47738/jads.v5i2.196 [Online]. Available: <https://bright-journal.org/Journal/index.php/JADS/article/view/196>
- [2] A. Odeh and A. Abu Taleb, “Ensemble-based deep learning models for enhancing iot intrusion detection,” *Applied Sciences*, vol. 13, no. 21, 2023, ISSN: 2076-3417. DOI: 10.3390/app132111985 [Online]. Available: <https://www.mdpi.com/2076-3417/13/21/11985>
- [3] A. Jafar and M. Lee, “High-speed hyperparameter optimization for deep resnet models in image recognition,” *Cluster Computing*, vol. 26, pp. 2605–2613, 2023. DOI: 10.1007/s10586-021-03284-6
- [4] G. Pang, C. Shen, and A. van den Hengel, “Deep anomaly detection with deviation networks,” *CoRR*, vol. abs/1911.08623, 2019. arXiv: 1911.08623. [Online]. Available: <http://arxiv.org/abs/1911.08623>
- [5] T. Tayeh, S. Aburakhia, R. Myers, and A. Shami, *Distance-based anomaly detection for industrial surfaces using triplet networks*, 2020. arXiv: 2011.04121 [cs.CV]. [Online]. Available: <https://arxiv.org/abs/2011.04121>
- [6] J. Villalba-Diez, D. Schmidt, R. Gevers, J. Ordieres-Meré, M. Buchwitz, and W. Wellbrock, “Deep learning for industrial computer vision quality control in the printing industry 4.0,” *Sensors*, vol. 19, no. 18, 2019, ISSN: 1424-8220. DOI: 10.3390/s19183987 [Online]. Available: <https://www.mdpi.com/1424-8220/19/18/3987>
- [7] D. Racki, D. Tomazevic, and D. Skocaj, “A compact convolutional neural network for textured surface anomaly detection,” in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, 2018, pp. 1331–1339. DOI: 10.1109/WACV.2018.00150
- [8] K. Sohn, T. Pfister, C.-L. Li, C.-Y. Lee, and J. Yoon, “Anomaly clustering: Grouping images into coherent clusters of anomaly types,” in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, 2023, pp. 5468–5479. DOI: 10.1109/WACV56688.2023.00544
- [9] M. Rohe, J. Hildebrand, J. Reimann, J. P. Bergmann, and B. N. Stoll, “Detecting process anomalies in the gmaw process by acoustic sensing with a convolutional neural network (cnn) for classification,” *Journal of Manufacturing and Materials Processing (JMMP)*, vol. 5, no. 4, p. 135, 2021. DOI: 10.3390/jmmp5040135

- [10] Q. Liu, J. Paparrizos, P. Boniol, and T. Palpanas, “Time-series anomaly detection: Overview and new trends,” *Proceedings of the VLDB Endowment*, vol. 17, no. 12, pp. 4229–4232, 2024. DOI: 10.14778/3685800.3685842
- [11] Y. Lai, J. Zhang, and Z. Liu, “Industrial anomaly detection and attack classification method based on convolutional neural network,” *Security and Communication Networks*, vol. 2019, pp. 1–11, Sep. 2019. DOI: 10.1155/2019/8124254
- [12] Q. A. Al-Haija, Y. Maleh, and A. Odeh, Eds., *Intelligent and Secure Solutions for Digital Transformation*, 1st ed. CRC Press, 2025. DOI: 10.1201/9781003616511
- [13] S. S. and S. J. G. R., “The enhanced anomaly deduction techniques for detecting redundant data in iot,” *IRJASH*, vol. 5, no. 02, pp. 47–54, 2023. DOI: 10.47392/irjash.2023.012
- [14] O. O. Olateju, A. A. Salami, U. T. I. Igwenagu, T. O. Oladoyinbo, O. O. Olaniyi, and S. U. Okon, “Combating the challenges of false positives in ai-driven anomaly detection systems and enhancing data security in the cloud,” *Asian Journal of Research in Computer Science*, vol. 17, no. 6, pp. 264–292, 2024. DOI: 10.9734/ajrcos/2024/v17i6472
- [15] S. Parhizkari, *Anomaly Detection in Intrusion Detection Systems*. IntechOpen, 2024. DOI: 10.5772/intechopen.112733
- [16] H. Liu, X. Xu, X. Li, E. Li, and S. Zhang, “Anomaly detection with representative neighbors,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 6, pp. 2831–2841, 2023. DOI: 10.1109/tnnls.2021.3109898
- [17] D. Zhang and T. Zhou, “Deep convolutional neural network using transfer learning for fault diagnosis,” *IEEE Access*, vol. 9, pp. 43 889–43 897, 2021. DOI: 10.1109/access.2021.3061530
- [18] K. Ro, G. Yin, C. Zou, and Z. Wang, “Outlier detection for high-dimensional data,” *Biometrika*, vol. 102, no. 3, pp. 589–599, 2015. DOI: 10.1093/biomet/asv021
- [19] X. Li, Z. Yi, and J. Lv, “Outlier detection using structural scores in a high-dimensional space,” *IEEE Transactions on Cybernetics*, vol. 50, no. 5, pp. 2302–2310, 2018. DOI: 10.1109/tcyb.2018.2876615
- [20] M. Radovanovic, A. Nanopoulos, and M. Ivanovic, “Reverse nearest neighbors in unsupervised distance-based outlier detection,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 5, pp. 1369–1382, 2015. DOI: 10.1109/tkde.2014.2365790
- [21] P. Liu, D. Zhou, and N. Wu, “Vdbscan: Varied density based spatial clustering of applications with noise,” in *Proceedings of the IEEE International Conference on Services Systems and Services Management (ICSSSM)*, 2007, pp. 1–4. DOI: 10.1109/icsssm.2007.4280175
- [22] R. J. G. B. Campello, D. Moulavi, A. Zimek, and J. Sander, “Hierarchical density estimates for data clustering, visualization, and outlier detection,” *ACM Transactions on Knowledge Discovery from Data*, vol. 10, no. 1, pp. 1–51, 2015. DOI: 10.1145/2733381

- [23] Y. Chen and L. Tu, “Density-based clustering for real-time stream data,” in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, 2007, pp. 133–142. DOI: 10.1145/1281192.1281210
- [24] P. A. Forero, V. Kekatos, and G. B. Giannakis, “Robust clustering using outlier-sparsity regularization,” *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 4163–4177, 2012. DOI: 10.1109/tsp.2012.2196696
- [25] B. Bahmei, E. Birmingham, and S. Arzanpour, “Cnn-rnn and data augmentation using deep convolutional generative adversarial network for environmental sound classification,” *IEEE Signal Processing Letters*, vol. 29, pp. 682–686, 2022. DOI: 10.1109/lsp.2022.3150258
- [26] X. Liu, Q. Sun, Y. Meng, S. Bourennane, and M. Fu, “Hyperspectral image classification based on parameter-optimized 3d-cnns combined with transfer learning and virtual samples,” *Remote Sensing*, vol. 10, no. 9, p. 1425, 2018. DOI: 10.3390/rs10091425
- [27] S. Sharma, S. Dhakal, and M. Bhavsar, “Transfer learning for wildlife classification: Evaluating yolov8 against densenet, resnet, and vggnet on a custom dataset,” *Journal of Artificial Intelligence and Computational Neuroscience (JAICN)*, vol. 6, no. 4, pp. 415–435, 2024. DOI: 10.36548/jaicn.2024.4.003
- [28] T. Tziolas, K. Papageorgiou, E. Papageorgiou, T. Mastos, T. Theodosiou, and A. Papadopoulos, “Autoencoders for anomaly detection in an industrial multivariate time series dataset,” in *Multidisciplinary Digital Institute Proceedings*, 2022, p. 23. DOI: 10.3390/engproc2022018023
- [29] K. M. N. K. Khalif, W. Chaw Seng, N. A. Shahrul, A. S. A. Bakar, and A. Gegov, “Integrated generative adversarial networks and deep convolutional neural networks for image data classification: A case study for covid-19,” *Information*, vol. 15, no. 1, p. 58, 2024. DOI: 10.3390/info15010058
- [30] H. Huang, T. Zhu, B. Wang, and J. Xiao, “Improved small-object detection using yolov8: A comparative study,” *ACE*, vol. 41, no. 1, pp. 80–88, 2024. DOI: 10.54254/2755-2721/41/20230714
- [31] M. Talib, J. Suad, and A. H. Y. Al-Noori, “Yolov8-cab: Improved yolov8 for real-time object detection,” *Karbala International Journal of Modern Science*, vol. 10, no. 1, 2024. DOI: 10.33640/2405-609x.3339
- [32] J. Wang et al., “Toward surface defect detection in electronics manufacturing by an accurate and lightweight yolo-style object detector,” *Scientific Reports*, vol. 13, May 2023. DOI: 10.1038/s41598-023-33804-w
- [33] A. Jafar and M. Lee, “High-speed hyperparameter optimization for deep resnet models in image recognition,” *Cluster Computing*, vol. 26, no. 5, pp. 2605–2613, 2021. DOI: 10.1007/s10586-021-03284-6
- [34] R. Rajkumar, K. Manivannan, and D. Shanthi, “Efficient guided grad-cam tuned patch neural network for accurate anomaly detection in full images,” *Information Technology and Control (ITC)*, vol. 53, no. 2, pp. 355–371, 2024. DOI: 10.5755/j01.itc.53.2.34525

- [35] S. Rajendar and V. Kumar Kaliappan, “Sensor data based anomaly detection in autonomous vehicles using modified convolutional neural network,” *Intelligent Automation & Soft Computing*, vol. 32, no. 2, pp. 859–875, 2022. DOI: 10.32604/iasc.2022.020936
- [36] B. Hrnjica and S. Softic, “Explainable ai in manufacturing: A predictive maintenance case study,” in *IFIP Advances in Information and Communication Technology*, Springer, 2020, pp. 66–73. DOI: 10.1007/978-3-030-57997-5_8 [Online]. Available: https://doi.org/10.1007/978-3-030-57997-5_8
- [37] M. Stephen and K. Sherifdeen, *Ai-enabled anomaly detection in industrial systems: A new era in predictive maintenance*, ResearchGate, 2022. [Online]. Available: https://www.researchgate.net/publication/386443676_AI-Enabled_Anomaly_Detection_in_Industrial_Systems_A_New_Era_in_Predictive_Maintenance
- [38] A. Jaramillo-Alcazar, J. Govea, and W. Villegas-Ch, “Anomaly detection in a smart industrial machinery plant using iot and machine learning,” *Sensors*, vol. 23, no. 19, p. 8286, 2023. DOI: 10.3390/s23198286 [Online]. Available: <https://doi.org/10.3390/s23198286>
- [39] A. C. Santos, W. A. Souza, G. V. Barbara, M. F. Castoldi, and A. Goedtel, “Diagnostics of early faults in wind generator bearings using hjorth parameters,” *Sustainability*, vol. 15, no. 20, 2023, ISSN: 2071-1050. DOI: 10.3390/su152014673 [Online]. Available: <https://www.mdpi.com/2071-1050/15/20/14673>
- [40] O. Mey and D. Neufeld, “Explainable ai algorithms for vibration data-based fault detection: Use case-adapted methods and critical evaluation,” *arXiv*, Jun. 2022. arXiv: 2207.10732 [cs.LG]. [Online]. Available: <https://arxiv.org/abs/2207.10732>
- [41] D. Grunova, V. Bakratsi, E. Vrochidou, and G. A. Papakostas, “Machine learning for anomaly detection in industrial environments,” *MDPI Engineering Proceedings*, vol. 25, 2024. DOI: 10.3390/engproc2024070025 [Online]. Available: <https://doi.org/10.3390/engproc2024070025>
- [42] N. A. T. Aderamo, N. H. C. Olisakwe, N. Y. A. Adebayo, and N. A. E. Esiri, “Ai-driven hse management systems for risk mitigation in the oil and gas industry,” *Comprehensive Research and Reviews in Engineering and Technology*, vol. 2, no. 1, pp. 001–022, 2024. DOI: 10.57219/crret.2024.2.1.0059 [Online]. Available: <https://doi.org/10.57219/crret.2024.2.1.0059>
- [43] L. I. Elom, “The effects of environmental cost on oil and gas performance in nigeria: A study of mainland oil and gas company, calabar,” Doctoral Thesis; deposited March 04 2025, Ph.D. dissertation, UNICAF University, Malawi Campus, 2025. [Online]. Available: <https://www.unicafuniversity.ac.mw/thesis-publications/the-effects-of-environmental-cost-on-oil-and-gas-performance-in-nigeria-a-study-of-mainland-oil-and-gas-company-calabar/>
- [44] P. D. Waghere, S. Wagh, and A. L. Rane, “Enhancing industrial safety: Real-time anomaly detection with opencv on edge platforms,” *IRJETS*, vol. 06, no. 05, 2024. [Online]. Available: https://www.irjmets.com/uploadedfiles/paper/issue_5_may_2024/56057/final/fin_irjmets1715828582.pdf

- [45] M. S. GmbH, *Mvtec anomaly detection dataset (mvtec ad)*, <https://www.mvtec.com/company/research/datasets/mvtec-ad>, Accessed: 2025-06-16, 2019.
- [46] S. M. E. Sumaia Mohamed Elhassan Saad Mohamed Darwish, “An enhanced lung cancer detection approach using dual-model deep learning technique,” *Computer Modeling in Engineering & Sciences*, vol. 142, no. 1, pp. 835–867, 2025, ISSN: 1526-1506. DOI: 10.32604/cmescs.2024.058770 [Online]. Available: <http://www.techscience.com/CMES/v142n1/58999>
- [47] J.-M. Woo, S.-H. Ju, J.-H. Sung, and K.-M. Seo, “Meta-learning-based lstm-autoencoder for low-data anomaly detection in retrofitted cnc machine using multi-machine datasets,” *Systems*, vol. 13, no. 7, 2025, ISSN: 2079-8954. DOI: 10.3390/systems13070534 [Online]. Available: <https://www.mdpi.com/2079-8954/13/7/534>