Many-to-One: Transformer-based Unsupervised Anomaly Detection and Localization on Industrial Images

By

Naga Jyothirmayee Dodda

A Thesis
Submitted to the Faculty of Graduate Studies
through the School of Computer Science
in Partial Fulfillment of the Requirements for
the Degree of Master of Science
at the University of Windsor

Windsor, Ontario, Canada

2023

Many-to-One: Transformer-based Unsupervised Anomaly Detection and Localization on Industrial Images

by

Naga Jyothirmayee Dodda

APPROVED BY:

J. Urbanic
Department of Mechanical, Automotive and Materials Engineering

B. Boufama
School of Computer Science

Z. Kobti, Advisor
School of Computer Science

August 31, 2023

DECLARATION OF CO-AUTHORSHIP/PREVIOUS PUBLICATION

1. Co-Authorship

   I hereby declare that this thesis incorporates material that is the result of research conducted under the supervision of Dr. Ziad Kobti. In all cases, the key ideas, primary contributions, experimental designs, data analysis, and interpretation were performed by the author, and the contribution of co-authors was primarily through the proofreading of the published manuscripts.

   I am aware of the University of Windsor Senate Policy on Authorship, and I certify that I have properly acknowledged the contribution of other researchers to my thesis and have obtained written permission from each of the co-author(s) to include the above material(s) in my thesis.

   I certify that, with the above qualification, this thesis, and the research to which it refers, is a product of my own work.

2. Previous Publication

   This thesis includes the paper that has been submitted in a peer-reviewed conference, as follows:

Publication Title/Full Citation	Publication Status
N.J. Dodda and Z. Kobti. Many-to-One: Transformer-based Unsupervised Anomaly Detection and Localization on Industrial Images. In 2023 International Conference on Machine Learning and Applications, ICMLA 2023.	Accepted (To appear in the proceedings of the ICMLA conference, 15-17 December 2023)

   I certify that I have obtained written permission from the copyright owner(s) to include the above-published material(s) in my thesis. I certify that the above material describes work completed during my registration as a graduate student at the University of Windsor.

3. General

I declare that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained written permission from the copyright owner(s) to include such material(s) in my thesis. I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

Anomaly detection is of utmost importance in the realm of industrial defect identification, particularly when employing computer vision-based inspection mechanisms within quality control systems. This research introduces the Many-to-One (M2O) framework, which relies on a multi-level transformer encoder combined with a single transformer decoder, which forms many-to-one relation in the framework for detecting and localizing anomalies. The rise of Industry 4.0 and electric vehicles has increased interest in this area. Although previous research has made significant contributions, challenges still exist in this area. It is crucial to develop models that can generalize well and overcome time complexity problems that affect model performance. The proposed M2O framework aims to address these challenges and improve the robustness and efficiency of anomaly detection and localization in this domain. M2O is a reconstruction framework that utilizes transformer-based architecture and employs a novel module called Multi-Level Feature Fuse to address these challenges. In order to establish a benchmark for industrial electrical connectors, we have introduced a novel dataset named ECAD, which contains real-world anomalies. This dataset has the potential to inspire further research in this field. Through evaluation against MVtec AD, BTAD, and ECAD, we have demonstrated that M2O outperforms existing methods. Extensive comparisons have established M2O's ability to overcome previous limitations, making it a robust solution for detecting anomalies in industrial environments.

DEDICATION

This thesis is sincerely dedicated to my loving parents, Aruna Dodda and Prasad Dodda, whose unwavering belief in the value of education has served as a guiding light throughout my academic journey. Their constant support and encouragement have been instrumental in shaping my aspirations and fostering my self-belief, even during moments of doubt. I am immensely grateful to them for being my pillars of strength.

In addition, I extend my heartfelt appreciation to my beloved sister, Prathyusha Dodda, whose humour and companionship have been a source of solace during times of adversity and challenges.

Furthermore, I am deeply indebted to my maternal family, particularly my uncles, whose profound influence has laid a strong foundation for the person I am today. Their continuous guidance and unwavering support have played a pivotal role in my personal and academic growth.

I humbly dedicate this accomplishment to my entire family, for their boundless love and encouragement have been the driving force behind my achievements. Their faith in me, even when faced with seemingly challenging obstacles, has propelled me forward. I acknowledge their sacrifices, which have enabled me to pursue my education and reach this significant milestone in my academic journey. This momentous achievement would not have been possible without their presence and unwavering belief in me.

With utmost gratitude, I honour and dedicate this thesis to my beloved family.

ACKNOWLEDGEMENTS

I sincerely thank my esteemed advisor, Dr. Ziad Kobti, whose exceptional research acumen and unwavering mentorship have profoundly influenced my academic journey. Dr. Kobti provided invaluable opportunities to explore my research domain, offering continuous encouragement and guidance that elevated the quality of my work. Their dedication to my growth as both a researcher and an individual has been truly inspiring. The constructive feedback they provided played a pivotal role in refining my thesis and instilling in me a profound sense of self-esteem and accomplishment. Collaborating with Dr. Kobti has been a privilege, and I am deeply grateful for the invaluable opportunities and support throughout this academic pursuit.

I also extend my sincere appreciation to my thesis committee members, Dr. Jill Urbanic and Dr. Boubakeur Boufama, whose invaluable insights and inspiration played a significant role in the successful completion of this thesis.

My association with the University of Windsor has been a great honor, and I express my gratitude to SHARCNET for their provision of essential computational resources. Moreover, I am thankful for the guidance offered by other esteemed professors at UWindsor during the course of my MSc program.

My heartfelt thanks go to Mrs. Gloria Mensah, Mrs. Melissa Robinet, and Mrs. Christine Weisener for their unwavering support and invaluable assistance in resolving various academic matters.

Above all, I owe a profound debt of gratitude to my family, whose constant inspiration and unwavering encouragement have been a pillar of strength throughout my journey. Their unwavering support has been immeasurable, and I am forever indebted to them.

Lastly, I wish to express my acknowledgment to my friends, whose continuous encouragement and care have made living away from home more manageable. Words cannot adequately convey the depth of my appreciation for the unwavering support from my esteemed advisor, devoted family, and cherished friends, all of whom have played an integral role in every aspect of my academic and personal growth.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

TP	True Positive
FN	False Negative
FP	False Positive
TN	True Negative
TPR	True Positive Rate
FPR	False Positive Rate
FNR	False Negative Rate
TNR	True Negative Rate
MSE	Mean Squared Error
AL	Anomaly localization
AD	Anomaly Detection
MB	Mega Byte
AUROC	Area Under the Receiver Operating Characteristic curve
CNN	Convolutional Neural Network
GAP	Global Average Pooling
MLP	Multi Layer Perceptron
MLFF	Multi Level Feature Fusion
GPE	Global Positional Encoding
LPE	Local Positional Encoding

Chapter 1

Introduction

1.1 Background

Ensuring the safety and efficiency of industrial operations has always relied on industrial inspection as a crucial component of the manufacturing process. With the ever-evolving manufacturing sector, particularly in the era of Industry 4.0, the approach to inspections has undergone a significant transformation. Industry 4.0, also known as the fourth industrial revolution, aims to enhance productivity, quality, and safety in smart factory environments. As a result, the implementation of automated inspection systems has become essential to meet the growing demands for improved quality and reduced manufacturing costs. These intelligent inspection systems aim to achieve a consistent and accurate inspection process while also contending with the inherent uncertainties and maintaining precision. However, effectively controlling the level of uncertainty in automated inspections remains a challenging task.

The absence of efficient quality inspection systems results in the occurrence of product recalls. In the present day, these recalls impose substantial financial burdens, regardless of the industry involved. Recalls have both financial and non-financial implications for companies, including production downtime and damage to their reputation. This holds true in various industries, including the automotive sector. It is worth noting that the automotive industry, like many other industries, experienced significant market changes during the 2008 financial crisis. Some notable recent incidents in the automotive industry include:

- In February 2023, Nissan was obligated to issue a recall for around 1100 vehicles

owing to the absence of screws in the steering wheel assembly [1].

- In May 2023, a recall was issued for several models of golf carts, including the Advent 4F, Advent 4FL, Advent 6 and Advent 6L. Approximately 2,500 units were affected by this recall, which resulted in a significant financial loss of $31 million [65].
- In June 2023, Tesla was forced to issue a recall for 7,696 vehicles because of a malfunction in the seat belts [17].

In order to underscore the economic ramifications within the automotive industry, it is crucial to note that each minute of downtime on the production line results in an average financial loss of $22,000[48]. These interruptions can have severe consequences on profitability and can cause disruptions throughout the entire supply chain. To effectively address these challenges, it is imperative to prioritize the implementation of Anomaly Detection and Localization systems in conjunction with Quality Control systems. By utilizing these systems, organizations can identify and prevent defects such as missing components or scratches, thereby ensuring that only high-quality products are manufactured. This approach allows for the reduction of unplanned downtime and enhances overall efficiency in production.

Figure 1.1.1 showcases a production line with red-coloured boxes symbolizing defective products and yellow-coloured boxes representing high-quality, anomaly-free items. The system utilizes robots to perform product sorting alongside a process flow control system and a Camera-based computer vision system, which forms a crucial part of the quality control systems. This computer vision system continuously monitors the products, facilitating anomaly detection, which constitutes the primary focus of the presented research. In the next section, we will delve into the concepts of anomaly detection and localization.

1.1.1 Anomaly Detection(AD):

Anomaly detection is the process of identifying unique patterns or behaviours in data that deviate from the expected or normal behaviour [12]. This is relevant in the field

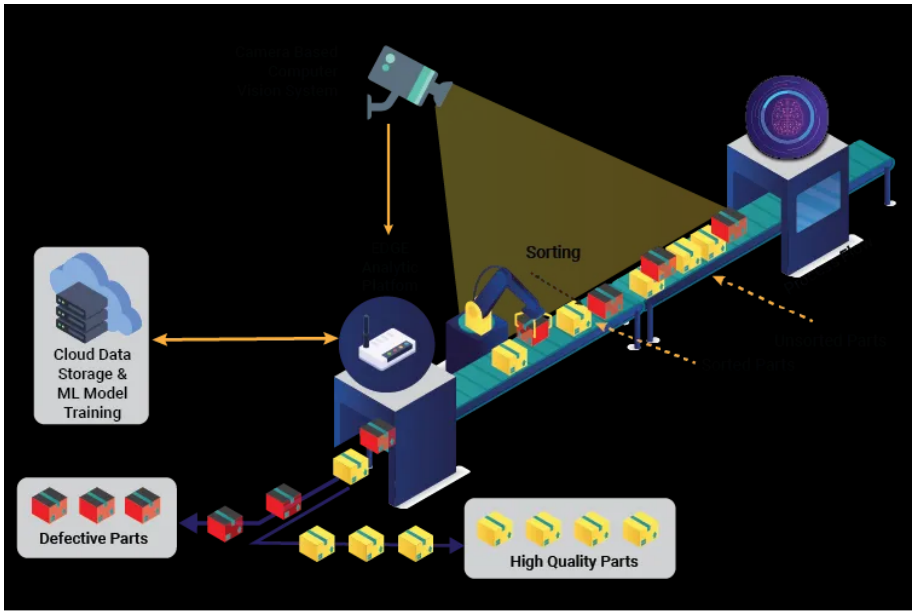

Figure 1.1.1: An illustration of how automation systems work in Industry4.0.[69]

of computer vision, particularly in image classification tasks.

Figure 1.1.2 illustrates the task of anomaly detection. In the image, there are four pictures of connectors. The goal of the anomaly detection algorithm is to classify the three images without any anomalies as normal and consider the fourth image with a spot on the connector as an anomaly. These images are part of a new dataset, which will be introduced later in the section titled "Datasets" (referenced as section 4.3).

1.1.2 Anomaly Localization(AL):

Anomaly localization, on the other hand, refers to the process of pinpointing the specific region or pixel-level location within an image where the anomaly occurs. By localizing the anomaly, it becomes easier to understand and analyze the source of the deviation from normal behaviour [11].

Figure 1.1.3 illustrates the objective of anomaly localization in the context of image analysis. The task entails the classification of three images that exhibit normal characteristics while identifying and categorizing a fourth image that contains an abnormality in the form of a spot on the connector. The anomaly localization algorithm aims to not only differentiate between normal and abnormal images but also pinpoint

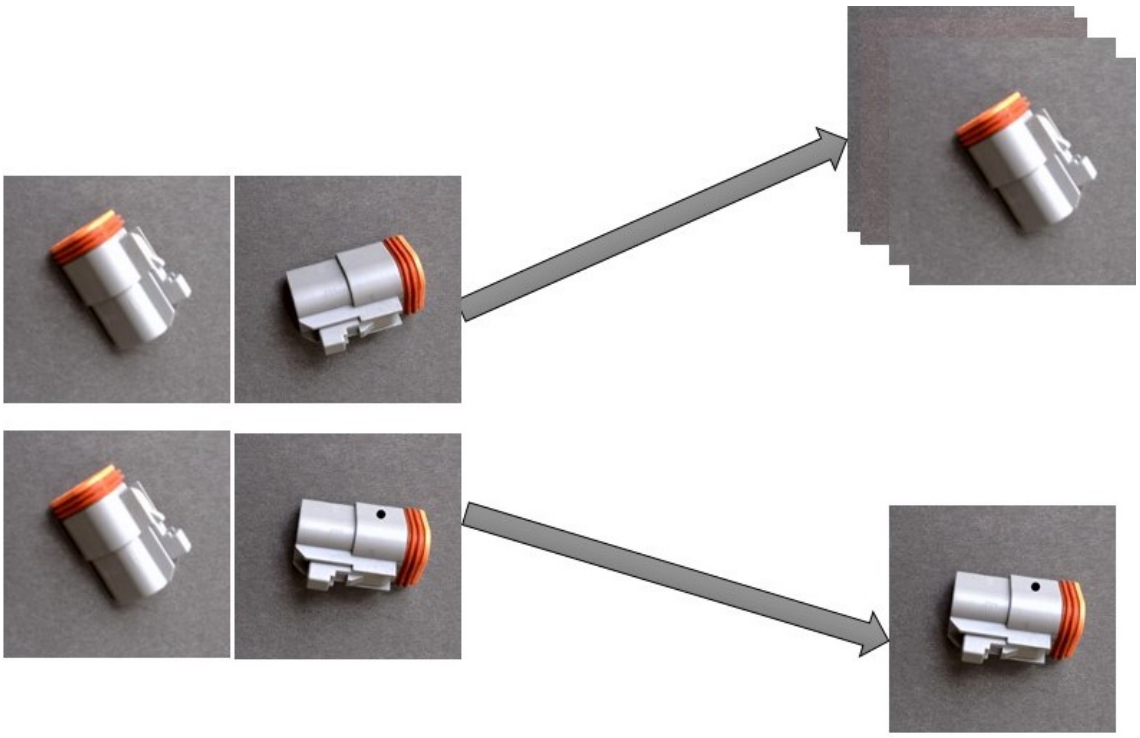

Figure 1.1.2: An illustration Anomaly Detection task

and highlight the specific abnormal region on the connector. This process enables the accurate identification and localization of anomalies within the given images.

In order to gain a deeper understanding of the various types of anomalies that are being examined as part of this study, we have included a representation of some examples in Figure 1.1.4 from the MVTec AD dataset. This dataset contains various categories of images, categorized as either textured or non-textured/object. The top row displays images without any anomalies, while the bottom row showcases images with highlighted anomalies represented by red markings. By utilizing anomaly detection and localization techniques, it becomes possible to identify discrepancies or outliers in image data, allowing for more accurate classification and analysis. Moving forward, let us understand different approaches for detecting and localizing anomalies in images in the following subsection.

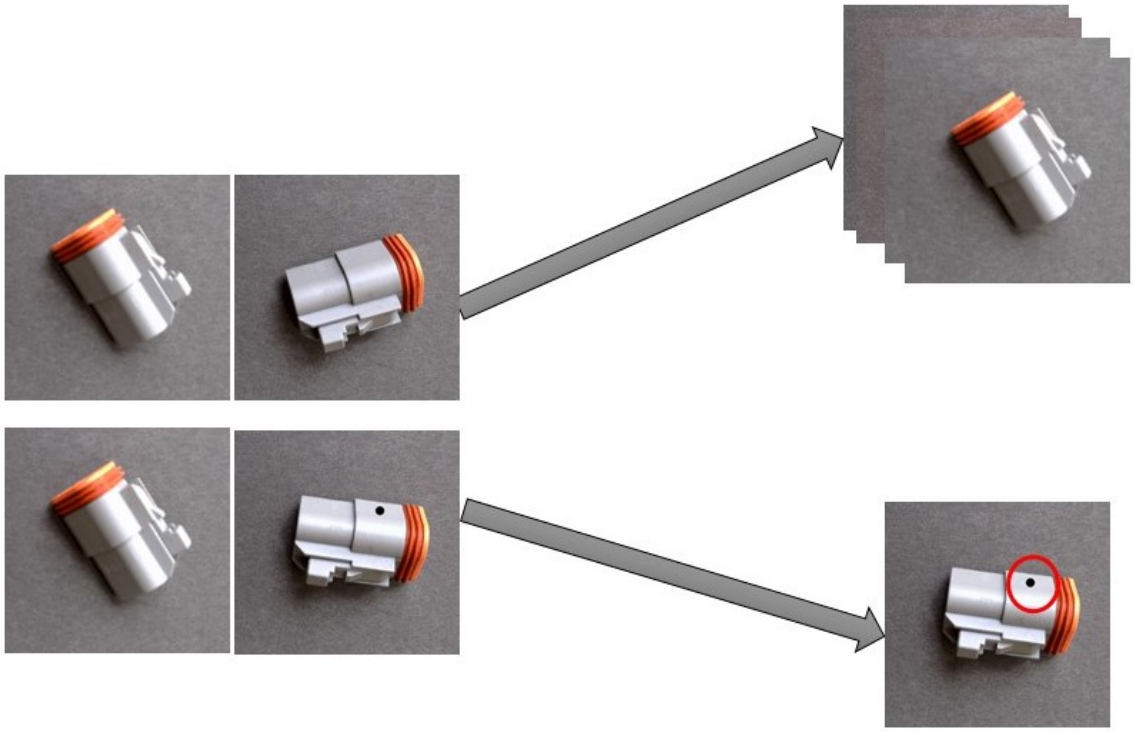

Figure 1.1.3: An illustration Anomaly Localization task

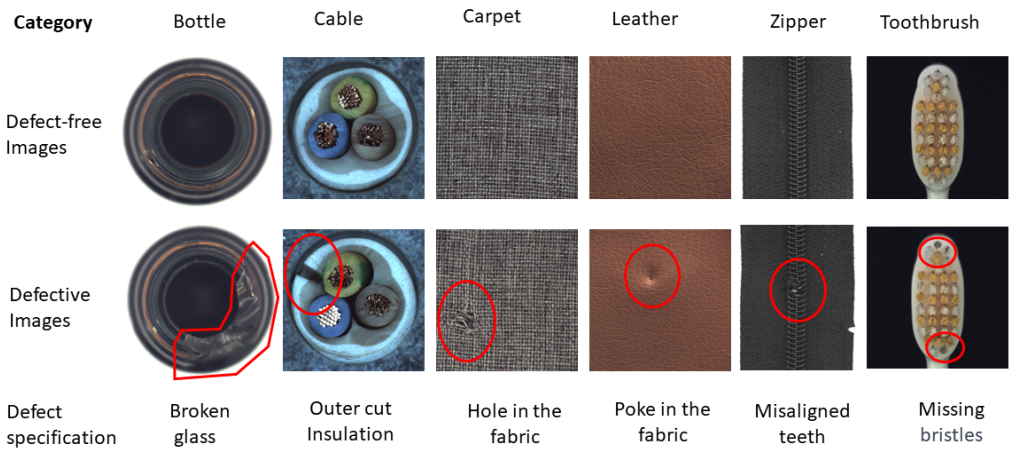

Figure 1.1.4: An illustration of Anomalous and Anomaly free images[9].

1.1.3 Anomaly Detection and Localization Approaches

Anomaly detection and localization can be approached in two main ways: supervised and unsupervised methods.

Supervised methods, methods require training models using both target labels and data. On the other hand, In unsupervised methods, models are trained only using data. Unsupervised methods extract patterns and structures from unlabeled data without any label and automatically group the data according to similarity as a resolution for the challenges mentioned. Here, it extracts patterns and structures from data without explicit labels. Building upon the foundation we have established, on the two primary approaches for anomaly detection and localization

1.1.3.1 Supervised Approach

Supervised methods involve training models using both target labels and data, as depicted in Figure 1.1.5(a). Each instance is classified or predicted as either normal or anomalous.

In the context of anomaly detection, the supervised approach involves assigning a binary label to each instance in the dataset, where 0 represents anomaly-free, and 1 represents an anomaly. Additionally, in the case of anomalous images, the specific regions that are anomalous can be marked or annotated. This annotation process entails marking the anomalous regions in red, as depicted in Figure 1.1.5(c) of the source material. Nevertheless, there are certain difficulties associated with this approach.

- The process of gathering and labelling data can be quite costly and time-consuming.
- It requires the expertise of domain specialists to accurately identify and annotate anomalies.
- Supervised methods necessitate prior knowledge of all potential defect categories, which may restrict their applicability.

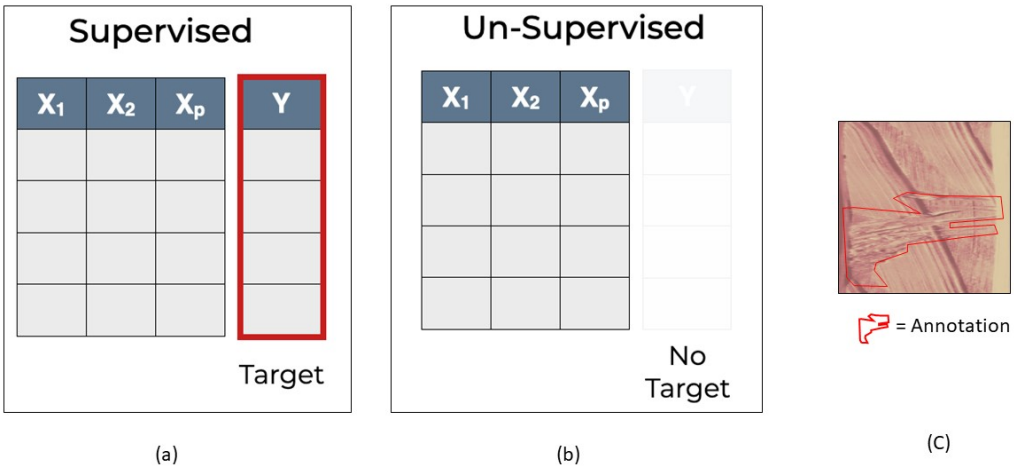

Figure 1.1.5: An illustration of (a) Supervised AD Approach, (b) Unsupervised AD Approach and (c) Annotation .

- The act of labelling data can inadvertently introduce errors or noise in the annotations, which can have a detrimental effect on the model's accuracy.

1.1.3.2 Unsupervised Approach

Unlike supervised methods, unsupervised approaches rely on unclassified data to extract patterns and structures without the need for predefined labels. These techniques automatically group data based on their similarities, making them a valuable solution for addressing the challenges encountered with supervised anomaly detection methods. One of the key advantages of unsupervised methods is that they do not require labelled data, allowing them to detect anomalies in various applications, including the identification of manufacturing defects and other similar scenarios.

In this study, we utilized the MVTec Anomaly Detection and BTAD datasets for our experiments. These datasets consist of a wide range of industrial products such as capsules, pills, wood, leather, and others. Within these datasets, there are images that represent normal products as well as images that exhibit various types of defects, such as cracks, splits, and breaks. Additionally, we have introduced a new

dataset that focuses specifically on electrical connectors used in vehicles for anomaly detection and localization purposes. These datasets serve as crucial benchmarks for evaluating the performance of anomaly detection and localization algorithms.

1.2 Problem Definition

In this section, we provide the problem definition with respect to this research work:

From [42], we define our problem statement for Anomaly Detection and Anomaly Localization as follows:

Given a dataset $X = \{x_1, x_2, ..., x_n\}$, where x_i is a feature vector in a high-dimensional space, the goal of a reconstruction model is to learn a function $f : X \rightarrow R$ that assigns a score to each data point indicating its level of abnormality. The function f is trained to minimize the reconstruction error of a deep reconstruction model, which is defined as:

$$\text{Score}(X, f) = ||X - f(X)||_2 \tag{1}$$

Where X represents the input dataset, f represents the model as a function, $||.||_2$ represents the Euclidian distance between the input and reconstructed image, f (X) represents the reconstructed feature vector of X. To assess the performance of our model in distinguishing between anomalies and non-anomalies, we employ the AUROC (Area Under the Receiver Operating Characteristic) score. A higher AUROC score indicates a higher ability of the model to correctly classify anomalies and non-anomalies, with values closer to 1 indicating a better performance in distinguishing between the two classes.

1.3 Motivation

The motivation for deep learning research in anomaly detection and localization for industrial image inspection stems from the need for improved quality control, cost reduction, and enhanced productivity in manufacturing processes. Traditional meth-

ods of manual inspection are often time-consuming, labour-intensive, and prone to human errors, which can lead to costly defects and safety hazards. By leveraging deep learning techniques, industrial image inspection can be automated, enabling real-time monitoring and early detection of anomalies, thereby preventing faulty products from reaching the market and ensuring consistent quality standards.

Deep learning models can learn complex defect patterns, making them effective in identifying subtle anomalies that may go unnoticed by human inspectors or rule-based methods. Additionally, the data-driven insights derived from analyzing anomalies aid in process optimization and root cause analysis, further boosting efficiency. The scalability and adaptability of deep learning make it suitable for various industrial applications and scenarios, contributing to the advancement of smart manufacturing and Industry 4.0 initiatives. Ultimately, deep learning research in anomaly detection and localization empowers industries to achieve greater operational efficiency, ensure product safety, and maintain their competitive edge in the global market.

In our research, we have conducted an extensive literature review in this particular field. Researchers have employed different statistical distances and comparison methods as criteria for identifying abnormal features[20, 50, 18, 79]. However, these methods need more robustness when dealing with anomalies of varying scales and types. Projection-based approaches have also been utilized to map features into low-dimensional embedding spaces, where normal samples and anomalies can be better differentiated [33]. Early methods used autoencoders and GANs for image reconstruction, assuming a generalization gap between normal and anomalous samples [10, 37, 5, 6, 58]. To address the challenges related to generalization ability, a novel U-shaped reconstruction model based on transformers has been developed. Nevertheless, the existing state-of-the-art U-shaped model faces certain limitations in terms of model size and time complexity. In order to tackle these difficulties, this study presents an innovative strategy for identifying and assessing deviations at various levels as well as determining their underlying origins without any supervision. These problems include its

- Large model size;

- Huge no. of parameters;
- Long inference time;
- Long training time.

The discovery of these drawbacks has motivated us to explore the need for optimizing this model.

1.4 Thesis Statement

Our primary objective of this research is to develop a new framework by modifying State-of-the-art UTRAD [15] architecture and show improvement in terms of computational efficiency without compromising quantitative performance in terms of AUCROC : Computational efficiency:

- Reduce the model size;
- Reduce no. of parameters;
- Reduce inference time;
- Reduce training time.

The secondary objective is to introduce a new dataset that corresponds to the Electrical connectors that are used in vehicles such as Off-Road Vehicles, Agriculture, Motorcycle Wiring, etc., that helps in identifying anomalies such as missing rubber gourmets, missing parts and other anomalies, in the real world.

Additionally, we aim to evaluate the proposed M2O architecture on benchmark datasets as well as the newly proposed dataset and compare its performance against the original architecture. In addition, we have its quantitative performance with other state-of-the-art anomaly detection and localization methods to prove the validity of the work.

We hypothesize that by integrating skip connection aggregation functionality, i.e. the Multi-Level Feature Fuse (MLFF) Layer, and feeding the resulting aggregated

feature map to a single decoder instead of having multiple decoder layers during image reconstruction, we can achieve improvements in terms of:

- Computational efficiency, which is reflected by reduced training time, inference time, and model size and no. of parameters.
- Quantitative performance, which is measured using AUROC.

1.5 Thesis Contribution

This thesis addresses the Anomaly detection and Anomaly Localization problem and proposes a framework called "Many-to-One(M2O): Transformer-based Unsupervised Anomaly Detection and Localization on Industrial Images". The proposed approach optimizes the U-shaped Transformer network of the existing state-of-the-art deep learning framework UTRAD [15] for AD and AL. The use of MLFF, a novel aggregation component, helps in improving computational efficiency without compromising quantitative performance. The following are the key contributions of this work:

- We have proposed a novel framework that exhibits better computational efficiency compared to a U-shaped framework while maintaining excellent quantitative performance as measured by AUROC(Area Under the Receiver Operating Characteristic curve).
- We have introduced a novel Multi-level feature fuse module that effectively combines multi-level features.
- Development of ECAD, a new real-world dataset specifically designed for detecting surface anomalies in electric connectors (further elaborated upon in section 4.3).
- We have compared our proposed framework, M2O, against three real-world datasets - MVtec AD, BTAD and Connector Dataset that we have introduced.

The new component aids in reducing no. of parameters which in turn improves the computational efficiency without affecting the quantitative performance of a model.

1.6 Thesis Organization

The structure of this thesis work is outlined as follows:

In Chapter 2, we present a comprehensive review of prior research on traditional and deep learning approaches in the areas of Anomaly detection, Anomaly Localization, Skip Connection, as well as pre-trained network-based reconstruction.

In Chapter 3 of our study, we have focused on the M2O framework, in which we delve into in detail. Here, we provide a comprehensive overview of each individual component that plays a role within this particular approach.

In Chapter 4, we provide a detailed account of the experimental setup and Evaluation metrics that were used to determine how well our model performed. This encompasses various aspects, such as the specific datasets that were used, including our newly introduced dataset, as well as the hyper-parameters that were employed in our study.

In Chapter 5, we conducted experiments on three different datasets Mvtec AD, BTAD and newly introduced "connectors data set". We compared our framework to several models that are considered to be state-of-the-art.

In Chapter 6, we conclude our research, explain the insights we gained during our research work, and describe some of the opportunities for future work.

Chapter 2

Related Works

2.1 Anomaly detection and localization

The human visual system possesses an inherent ability to detect anomalies, enabling individuals to differentiate between defective and non-defective images even without prior exposure to defective samples. Additionally, humans can pinpoint the location of anomalies. Academic research introduced Anomaly Localization (AL) to impart similar capabilities to machines, teaching them to autonomously "find" anomaly regions without relying on labelled defective samples during the training stage. Unlike the supervised approach, AL methods under the unsupervised paradigm solely use normal images, avoiding the challenges of collecting anomalous or defective samples, reducing labelling costs, and mitigating the influence of labelling deviations.

AD, also known as outlier detection or one-class classification[61], aims to classify images at the image level, distinguishing defective images from the majority of non-defective images. On the other hand, AL, also referred to as anomaly segmentation, aims to produce pixel-level anomaly location results. A few categories of anomaly Localization results from MVTec AD are shown in Figure 2.1.1. The image consists of 8 rows, with alternating anomalous and normal images in each row. For instance, in the first row, we have a Localization map for the pill; the second column of the image reveals the likelihood of locations being anomalous, with the negligible colour shade of yellow indicating higher probabilities of anomalies.

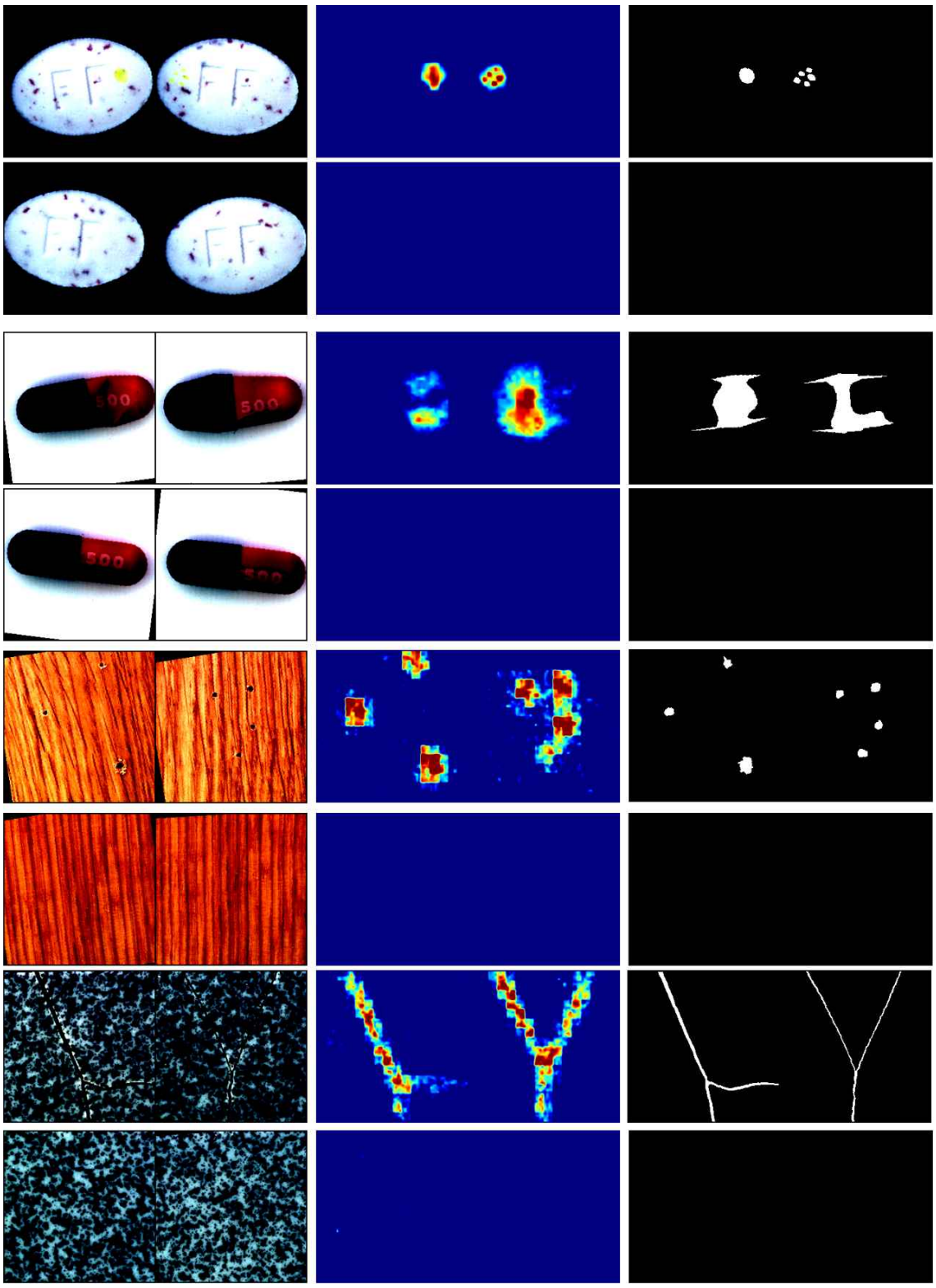

Figure 2.1.1: An illustration Anomaly Localization task in sub-datasets of MVTec AD dataset

2.1.1 Traditional Approaches

In this section, we provide an overview of traditional anomaly detection techniques, such as statistical approach, Filter based approaches. The statistical approach plays a fundamental role in identifying anomalies based on the probability distribution of normal data instances [7]. The underlying principle is that anomalies are observations suspected of being irrelevant as they deviate from the assumed stochastic model. These techniques can be categorized into two main classes: parametric and non-parametric methods.

Parametric techniques assume a specific probability distribution, such as the Gaussian model, for the normal data instances. The model's parameters are estimated from the training data, and the anomaly score of a test instance is determined based on its inverse probability density function. Alternatively, statistical hypothesis tests are employed to compare the test instance's likelihood under the estimated distribution [24]. Various parametric approaches, including Gaussian model-based and regression model-based techniques, have been explored [3].

Non-parametric techniques, on the other hand, make fewer assumptions about the data and estimate the probability density function without specifying a predefined distribution [44]. Notable non-parametric methods include histogram-based and kernel function-based techniques. Histogram-based techniques build a profile of normal data using histograms, enabling anomaly detection based on test instances' positions within the histogram bins. Kernel function-based techniques use kernel functions to approximate the data density and assess anomalies accordingly [22].

Advantages of statistical techniques include their statistical justification, provision of confidence intervals for anomaly scores, and their potential for unsupervised operation when robust to anomalies. However, these techniques heavily rely on the assumption that data follows specific distributions, which may not hold in complex real-world datasets. Choosing appropriate hypothesis tests and capturing interactions between attributes in multivariate data are also challenging aspects.

In summary, statistical anomaly detection techniques encompass a range of para-

metric and non-parametric methods, offering valuable tools for anomaly identification based on probability distributions. Although these techniques offer statistical justification and confidence intervals, the assumptions about data distributions and interactions between attributes present challenges that justify further investigation.

Additionally, In the domain of anomaly recognition, various filter-based methods have been extensively utilized, entailing the application of diverse filter banks on defect images. These methods compute the energy of the responses generated by the applied filters [45, 8, 32, 4, 27, 72, 76, 36, 34, 29, 64, 78, 67]. The repertoire of commonly employed filter-based techniques encompasses the Sobel operator, which is a gradient-based edge detector, the Canny operator, which is a multi-stage algorithm for detecting edges in images, the Gabor operator, which is utilized for texture analysis and feature extraction in image processing, the Laplacian operator, which is used for edge detection and image sharpening, the wavelet transform, which allows multi-resolution analysis of signals and images, and the Fourier transform, which is applied to analyze the frequency components of signals and images.

These filter-based methods can be categorized into spatial domain, frequency domain, and spatial-frequency domain approaches. In the context of vision-based anomaly recognition, filter-based methodologies have gained widespread usage due to their capability to extract essential features invariant to affine transformations and their effectiveness in handling multi-scale defects. However, it is pertinent to acknowledge that these methods may not be ideally suited for analyzing random textured images, and some of them might be influenced by feature correlations and noise.

2.1.2 Deep learning Approaches

In anomaly detection, deep learning approaches in supervised techniques necessitate a training dataset with labelled instances for both normal and anomaly classes. The conventional approach involves constructing a predictive model that discriminates between normal and anomaly classes. Subsequently, any new data instance can be evaluated against the model to determine its class membership. However, two primary

challenges arise in supervised anomaly detection. Firstly, the number of anomalous instances in the training data is typically much smaller than the normal instances, leading to imbalanced class distributions. Researchers in data mining and machine learning have addressed this issue through various methodologies [30, 14, 46, 74, 68]. Secondly, obtaining accurate and representative labels, particularly for the anomaly class, is often problematic. To mitigate this challenge, several techniques have been proposed. involving the injection of artificial anomalies into a normal data set to create a labelled training data set [62, 2, 59]. Given these challenges, the unsupervised anomaly detection technique is the best approach over the supervised approach.

The traditional supervised learning methods require labelled data for both normal and anomaly classes, which may be challenging to obtain in many real-world scenarios [38]. Unsupervised methods do not require labelled data, making them more suitable for practical applications [80, 47]. These methods include reconstruction-based approaches that use encoding and decoding on normal images to detect anomalies based on reconstruction errors [77, 66], normalizing flow (NF)-based methods that learn transformations between data distributions to estimate anomalies [70, 75], representation-based techniques that extract meaningful feature vectors and calculate distances to identify anomalies [39, 53], and data augmentation-based algorithms that create synthetic anomalies for training [25, 54].

The survey [19] presents a comprehensive comparison and analysis of the discussed approaches. It highlights the advantages and limitations of each method, considering factors such as accuracy, generalizability, and computational complexity. While supervised methods tend to achieve higher accuracy, they suffer from the need for labelled data, limiting their practical applicability. On the other hand, unsupervised methods do not require labelled data, making them more scalable and cost-effective for anomaly detection. However, they may have lower accuracy, and their complexity varies depending on the specific approach. Ultimately, the choice of method should be based on the specific requirements of the application and the trade-offs between accuracy and ease of implementation.

With respect to this problem, Anomaly detection and localization methods often

rely on reconstructing images and using reconstruction errors to identify anomalies [5, 6, 10, 37, 58]. Autoencoders (AEs) are commonly used for this purpose, trained on normal samples to learn data representations [10]. Adversarial losses have been employed to enhance the model's ability to detect anomalies [5, 6]. Generative Adversarial Networks (GANs) have also been utilized for anomaly detection by generating images conforming to the training distribution [57, 58].

Pre-trained feature-based methods, on the other hand, extract features using pre-trained CNNs and then use statistical analysis methods or other networks to detect anomalies [56, 81, 20, 50, 79, 18]. However, these methods lack universally accepted criteria or generalization ability for accurate anomaly identification [56, 81, 20, 50, 79, 18].

To address these challenges, a U-shaped transformer-based autoencoder with skip connections was introduced in a previous work [15]. Although effective, this approach suffered from computational inefficiencies due to its large model size and lengthy training time [15].

2.1.3 U-shaped networks:

Researchers have raised concerns about the optimality of U-Net's U-shaped symmetric framework and the key factor influencing its performance. In their investigation, [16] explored different encoder configurations (MiMo, SiMo, MiSo, and SiSo) to assess their impact on U-Net's performance, revealing the prominent role of the divide-and-conquer strategy in the encoder. By simplifying feature fusion while preserving this strategy, they achieved comparable segmentation results with the U-shaped network.

2.1.4 Skip Connections:

Skip connections, first introduced in UNet for bridging the semantic gap, have been successful in recovering fine-grained details in image segmentation tasks [51, 82, 28, 35, 41]. However, recent studies have shown that not all skip connections are beneficial, and their contributions can vary significantly depending on the dataset and

model architecture [71]. UCTransnet identified that UNet without skip connections outperformed the original UNet, and different datasets require different optimal combinations of skip connections [71]. This emphasizes the need for more appropriate feature fusion methods beyond simple copying.

Chapter 3

Proposed Methodology

In this chapter, we introduce and describe the architecture. Specifically, a detailed explanation is given for each component, including the novel Multi-Level Feature Fuse Layer, which is the contribution of this work.

3.1 Proposed Framework

In this section, we discuss overall architecture. This work is based on the unsupervised reconstruction-based network, and we consider it as a binary problem. Using this approach, we can categorize each specific image into two classes: those that exhibit anomalies and those that do not show any signs of abnormalities. Additionally, in this particular technique centred on reconstruction, we exclusively train the model using normal images. By doing so, the model becomes familiar with reconstructing only anomaly-free images and struggles to properly reconstruct anomalous samples. Therefore, we utilize the reconstruction error as a metric for both detecting and localizing anomalies within our framework.

The M2O framework utilizes a pre-trained CNN backbone, specifically Resnet-18, which has been pre-trained on the ImageNet dataset. Research [56] has shown that the samples from these two classes exhibit distinct features in feature space. Therefore, we adopt the use of frozen weights of the Resnet-18 model as the initial component in this framework to extract multi-scale features. We make use of layer 1 -layer 3 of the network. This model consists of 4 layers, and for our purposes, we focus on the mid-layer feature maps from layer 1 to layer 3. These feature maps

are upsampled to the same size and then concatenated along the channel axis to create a multi-scale feature map. The obtained deep feature map, which differs from the 2-D image data, allows for independent processing of each feature vector along with its surrounding vectors. Each feature vector in the map corresponds to an image pixel. A transformer-based reconstruction network that can effectively handle feature embeddings as discrete "word" tokens. This ensures that regional and local information related to the original pixel is retained within the feature vector. Mapping these feature embeddings into text space enables efficient and accurate reconstruction of images with the help of the M2O Transformer, which we will discuss in the following section 3.2.

The output feature map of the first component, denoted as $F \in R^{C\times H\times W}$, is a multi-scale representation of the image that contains information about both regional and local features. To further analyze this feature map, we divide it into individual input feature embeddings of size $H \times W$, represented as $I \in R^{C\times 1\times 1}$, which will then be used as input for the M2O-Transformer in order to reconstruct the features at a higher level. The M2O-Transformer outputs the reconstructed feature map, denoted as $\hat{F}$. In M2O-Transformer, we employ linear projections (specifically 1×1 conv) to reduce the channel dimension at both the beginning and end stages.

$$\hat{F} = M_{M2O}(F) \tag{1}$$

Here, $M_{M2O}(\cdot)$ in equation 1 represents the M2O-Transformer network model. The M2O-Transformer network is trained using the reconstruction loss L_{M2O}, which utilizes the MSE loss. This loss function quantifies the difference between the original and reconstructed outputs. E Equation 2 provides a mathematical representation of this process.

$$L_{M2O} = \text{MSE}(\hat{F} - F) \tag{2}$$

To generate the anomaly score map, the reconstruction error $||\hat{F} - F||_2^2$ is used,

where $|| \cdot ||_2$ represents the l_2 norm across channels. However, the reconstruction error is still inconsistent for normal features. To address this issue, we utilize an MLP network to estimate the variance. The output of MLP $\Lambda \in \mathbb{R}^{H \times W}$ acts as a scale factor to refine the reconstruction error map to the score map. Finally, we define the score map $S \in \mathbb{R}^{H \times W}$ by performing element-wise division of the MSE loss and Λ as per the equation 3.

$$S = ||\hat{F} - F||_2 \oslash \Lambda \tag{3}$$

where $\oslash$ denotes element-wise division. The MSE loss L_{scale} is used

$$L_{\text{scale}} = \text{MSE}(F_{\text{scale}}(\hat{Z}), ||\hat{Z} - Z||_2). \tag{4}$$

$$L = L_{M2O} + L_{scale} \tag{5}$$

In the equation 4 & 5, the MLP network is trained using MSE loss denoted as L_{scale} and the overall loss function L is the sum of L_{M2O} and L_{scale} respectively. Further, to improve optimization, gradient flow is truncated between the M2O-Transformer and MLP.

3.2 M2O Transformer

In this section, we provide a comprehensive description of the M2O Transformer, including its encoder, skip connection aggregation function, decoder, and finetuning. The transformer has been demonstrated to be a powerful tool for various vision tasks due to its global attention mechanism and robust representational capacity. However, the original transformer has a high computational complexity, which limits its application as a feature reconstruction model for image anomaly detection [15]. Moreover, existing methods based on the transformer, such as [15] have a large model size and inference time. To address these issues, we propose a novel architecture M2O Trans-

former as an enhancement to [15], which employs a multi-level transformer encoder with a skip connection aggregation function MLFF and a single-layer decoder to enhance computational efficiency. The proposed model can serve as a feature reconstruction model with lower computational cost and model size compared to existing transformer-based autoencoders with skip-connection methods [15] while maintaining high performance for image anomaly detection.

3.2.1 Multi-Level Transformer Encoder

We use a Multi-Level Transformer Encoder to process an $H \times W$ feature map. The map is split into $N_H \times N_W$ patches of images, and each patch is a word token of size $T_H \times T_W$, where $T_H = H/N_H$ and $T_W = W/N_W$. Once we tokenize each image patch into $T_H \times T_W$ feature embeddings, an additional zero-padding token is added. These $T_H \times T_W + 1$ tokens serve as input tokens for the level-1 Transformer Encoder. The output of this encoder consists of $T_H \times T_W + 1$ tokens, including $T_H \times T_W$ feature latent vectors (corresponding to the $T_H \times T_W$ input feature embeddings) and a header latent vector (corresponding to the zero padding token). The $T_H \times T_W$ features latent vectors serve as inputs to the Multi-Level Feature Fuse Layer(MLFF) which in turn will be input to the Transformer Decoder. The header latent vector serves as an overall feature embedding of the entire patch and is used as input for the level-2 Transformer Encoder. In total, there are $N_H \times N_W$ header latent vectors. We reshape these vectors into a feature map of shape $N_H \times N_W$. We then repeat the tokenization and padding process to obtain input tokens for the level-2 Transformer Encoder. We can similarly construct a level-3 Transformer Encoder from a level-2 Transformer Encoder. In practice, the number of Transformer Encoder levels is set to 3. We use H_l and W_l to denote the input size of the l^{th} M2O-Transformer Encoder, and T_l to denote its patch size. The mathematical expressions of the M2O-Transformer Encoder at level

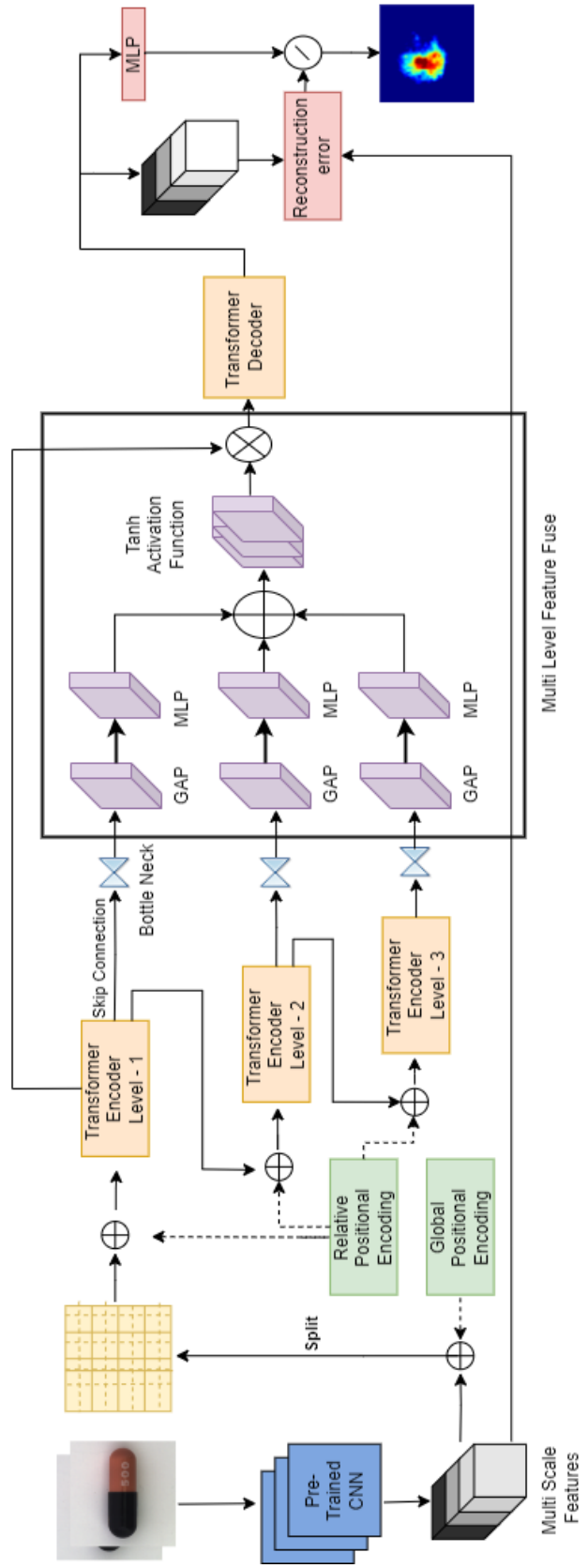

Figure 3.2.1: An Illustration of our M2O Framework for anomaly detection and localization.

l are defined as:

$$V_{l,0}^{(n)}, (V_{l,1}^{(n)}, V_{l,2}^{(n)}, \ldots, V_{l,T_l \times T_l}^{(n)}) = \\ F\text{Level} - l\text{-Encoder}(I_0, I_1^{(n)}, I_2^{(n)}, \ldots, I_{T_l \times T_l}^{(n)}), \\ n \in \{1, \ldots, \frac{H_l W_l}{(\prod_{k=1}^{l} T_l)^2}\} \tag{6}$$

where $I_0 = 0^{C \times 1 \times 1}$, $I_i^{(n)}, i \in \{1, \ldots, T_l \times T_l\}$ are the input embeddings of patch n at level l, $V_{l,i}^{(n)}, i \in \{0, \ldots, T_l \times T_l\}$ are the corresponding outputs, and $V_{l,0}^{(n)}$ is the general semantics embedding of patch n at level l for equation 6. The bottleneck can be defined as:

$$\hat{V}_{l,1}, \hat{V}_{l,2}, \ldots, \hat{V}_{l,H_l \times W_l} = F_{\text{Level}-l\text{-Bottleneck}}(V_{l,1}, V_{l,2}, \ldots, V_{l,H_l \times W_l}) \tag{7}$$

$$B_l = \hat{V}_{l,1}, \hat{V}_{l,2}, \ldots, \hat{V}_{l,H_l \times W_l} \tag{8}$$

Where B_l in equation 8 maintains the output of each encoder level after passing through the bottleneck.

To address the computational efficiency and reduce the overall model size and training time of transformer-based autoencoder for image anomaly detection and localization, we propose a novel approach in which the bottleneck outputs of the skip connections are preserved and fused using the Multi-Level Feature Fuse layer (MLFF). This technique allows us to efficiently aggregate information from multiple levels of the encoder and obtain a more comprehensive representation for feature reconstruction. Additionally, the proposed model achieves better performance compared to existing transformer-based autoencoders with skip connections while maintaining lower computational cost and model size.

3.2.2 Multi-Level Feature Fuse

A skip connection component, inspired by U-Net [51], to improve anomaly localization feature and preserve preserves low-level details embeddings are combined at different levels. However, our experiments revealed that the use of bottlenecks has a significant effect. We considered bottlenecks to each skip connection to promote better information flow. These bottlenecks are small autoencoders with 2 channelwise convolutional layers, with the first layer reducing the embedding dimension to $1/4^{th}$ and the second layer restoring the dimension.

At every level, the output from the skip connection is preserved as B_l and is used to fuse the features from every level along the Channel axis to propose a Multi-Level Feature Fuse layer (MLFF). This module guides the channel and information selection of the transformer features, resulting potentially more informative and redundant free representation of the input image which is crucial for reconstruction and to use it as one of the inputs to the decoder layer.

Mathematically, we take the $l-th$ level Bottleneck output $B_l \in \mathbb{R}^{C\times H\times W}$ for all the three levels. First, we perform the global average pooling (GAP) layer used to perform Spatial squeeze and produce vector $G(X) \in \mathbb{R}^{C\times 1\times 1}$, where its k^{th} channel is given by equation 9. To generate the attention mask A, global spatial information is embedded and is defined in equation 10.

$$g(X) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} B_l^k(i,j) \tag{9}$$

$$A = L_1 \cdot G(B_l) + L_2 \cdot G(B_{l-1}) + L_2 \cdot G(B_{l-2}) \tag{10}$$

In equation 10, $L_1 \in \mathbb{R}^{C\times C}$, $L_2 \in \mathbb{R}^{C\times C}$,$L_3 \in \mathbb{R}^{C\times C}$ are weight matrices of three Linear layers, and $\delta(\cdot)$ is the ReLU operator. This operation encodes channel-wise dependencies in the model. To improve the channel attention learning process, we followed the approach proposed in ECA-Net [73], which demonstrated the significance of avoiding dimensionality reduction. Therefore, a Linear layer with a sigmoid

function is utilized to build the channel attention map and is represented as:

$$\hat{O} = \sigma(A) \cdot B_l \tag{11}$$

As shown in equation 11, the resulting vector re-calibrates or excites the features at level 1 i.e. B_l to obtain the masked output, with the activation function $\sigma(A)$ that denotes the channel importance. Finally, the resultant output $\hat{O}$ will be one of the inputs to the decoder to reconstruct the image.

In order to better comprehend the MLFF module's aggregation functionality during

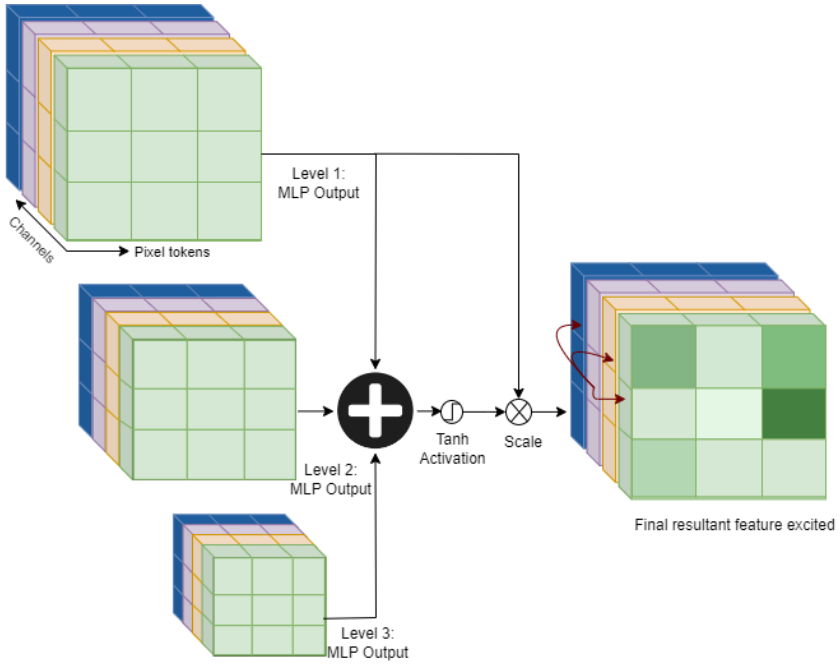

Figure 3.2.2: Illustration of Feature Fuse in MLFF of M2O Architecture

the Feature Fusion phase, Figure 3.2.2 provides a visual representation. The output obtained from each level of the encoder undergoes an upscaling process to match the size of the bottleneck at that particular level. These scaled outputs are then combined along the channel axis to produce an aggregated output. This aggregated output is subsequently utilized to adjust the size of the encoder output at level 1 and is inputted into the decoder to facilitate image reconstruction.

3.2.3 Transformer Decoder:

To reconstruct the image, the transformer decoder needs three inputs. First, the output from the Multi-Level Feature Fuse layer(MLFF) as residual embeddings represented by $T_H \times T_W$ as feature latent vectors. Second, the level-1 transformer encoder output used is a head latent vector. The residual embeddings and the head latent vector combine to create $T_H \times T_W + 1$ tokens, which serve as the memory inputs of the decoder. Furthermore, the query embeddings and transformer decoder outputs in the model have the same shape as the corresponding residual embeddings, and the query embeddings are also learnable embeddings. Consequently, the output feature embeddings of the transformer decoder, reconstructed as $T_H \times T_W$, are obtained.

$$
\begin{aligned}
&I_1^{(n)}, I_2^{(n)}, \ldots, I_{T_1 \times T_1}^{(n)} = \\
&F\text{Level} - l\text{-Decoder}(I_{1,i,j}^{(n)}, (O_1^{(n)}, O_2^{(n)}, \ldots, O_{l,T_l \times T_l}^{(n)}), \\
&n \in \{1, \ldots, \frac{H_l W_l}{(\prod_{k=1}^{l} T_1)^2}\}
\end{aligned}
\tag{12}
$$

The resultant output embeddings from the MLFF layer is denoted as $O_i^{(n)}, i = 1, \ldots, \frac{H_l W_l}{(\prod_{k=1}^{l} T_1)^2}\}$. and $I_{1,i,j}^{(n)}$ hold level 1 encoder output, and $I_1^{(n)}, i = 0, ..., T_1 \times T_1$ are the reconstructed output embeddings of the decoder. Algorithm 3.2.1 represents the inference pipeline of the complete framework.

3.2.4 Score Map Multi-Level Smoothing

Before calculating the final score map S, we split the score map into three, smooth them with a kernel size equal to the level scale, then add them to obtain the final score map. It is computed using Eq. (3), but reconstruction is done level-by-level, leading to a discrete score map with discontinuities between patches.

Algorithm 3.2.1 : Inference Pipeline

Input: Input image X
Output: Anomaly score map S

Initialization
$F \leftarrow Backbone(X)$
$F' \leftarrow GlobalPostionalEncoding(F)$
for $level = 1$ to 3 **do**
    Split feature map F' to $T_{level} \times T_{level}$ patches $F'_{i,j}$, $(i = 1, ..., T_{level}, j = 1, ..., T_{level})$
    for each $(i, j), i = 1, ..., T_{level}, j = 1, ..., T_{level}$ **do**
        Patch $F'_{i,j} \leftarrow LocalPostionalEncoding(F'_{i,j})$
        Patch $V_{level,i,j}, vector\, I_{level,i,j} \leftarrow Encoder_{level}(F'_{i,j}, 0)$
    end for
    Patch $\hat{V}_{level,i,j} \leftarrow Bottleneck_{level}(V_{level,i,j})$
    $B_{level} \leftarrow \hat{V}_{level,i,j}$
    Reform vectors $I_{i,j}$ to new-level feature map F'
end for
$\hat{O} \leftarrow MLFF(B_{level})$
$\hat{F} \leftarrow Decoder(\hat{O}, I_{1,i,j})$
$\sigma^2 \leftarrow MLP(\hat{F})$
$S \leftarrow ||\hat{F} - F||_2/\sigma^2$
return S

3.2.5 Positional Encoding

Our approach utilizes learned positional encodings, which have been demonstrated to outperform fixed positional encodings in previous research [23]. We drew inspiration from UTRAD and proposed a novel approach that employs two types of positional encodings, namely global positional encodings (GPE) and Local positional encodings (LPE) [15]. GPE encodes the position of features across the entire feature map, while LPE encodes the position of features within each patch. To preserve the spatial structure information, positional encodings must be added. LPE is added to the feature tokens prior to inputting them into the transformer encoder at each level, whereas GPE is added before the features are divided into tokens.

Chapter 4

Experimental Setup

This chapter describes our experimental setup and environment, including tools and libraries used to implement our model (M2O), System Configuration, Hyper-parameters for training, Dataset details, and detail of evaluation measures used to evaluate our model.

4.1 Tools and Libraries

To develop our model M2O, we utilized the UTRAD code, which was previously developed and provided to us as a resource [1]. Our code for the M2O model, which is outlined in this paper, is accessible on GitHub [2]. The implementation of our code was done in Python 3.8.10 [63]. We relied on various libraries and tools, the specifics of which are detailed below:

- Sklearn is used for Result analysis.
- PIL, OpenCV were used for Data set Manipulations.
- Pytorch, torchvision were used for Model Construction.
- Timeit is used for Time analysis.
- ptflops is used for Parameter Count.

[1] `https://github.com/gordon-chenmo/UTRAD/tree/main`
[2] `https://github.com/JyothirmayeeDodda/M2O`

- torchinfo is used for calculating Model size in memory.
- VSCode is used as IDE for development.
- ImageJ is the tool used for ground truth creation.

4.2 System Configuration

We conducted experiments on the Compute Canada - SHARCNET Cedar cluster, utilizing the powerful NVIDIA Tesla V100-SXM2 Graphics Processing Unit with a CUDA version of 11.8. The Tesla V100-SXM2 GPU is equipped with 32 GB of memory and offers exceptional performance. Furthermore, in terms of the Central Processing Unit core, the Tesla V100-SXM2 boasts an impressive 32 CPU cores, enhancing the overall computational capabilities of the system.

4.3 Datasets

The presented model is a versatile one that demonstrates excellent performance across various industrial datasets. We thoroughly assessed our M2O model using two well-established benchmark datasets - MVtec AD and BTAD - alongside several other cutting-edge models. Both the MVtec AD and BTAD datasets contain different subtypes of data, specifically pertaining to textures or non-textures. Moreover, we are introducing a novel dataset focused on electrical connectors Anomaly detection (ECAD), which holds significant importance given the ongoing revolution in electric vehicles. This dataset will be instrumental in accurately identifying anomalous regions within such connectors.

4.3.0.1 MVTec AD Dataset

The MVTec AD dataset is a well-known benchmark in the field of industrial inspection for detecting and localizing anomalies. Comprising over 5823 high-resolution images, it covers 15 distinct categories, including objects and textures commonly found in

industrial settings. Each category consists of both defect-free training images and test images that showcase various types of defects as well as defect-free samples.

Specifically, the training set contains approximately 3600 normal images, making up around 70% of the data. The remaining portion represents the test set with approximately 2200 images (30%). Within this subset, there are 468 good or non-defective images, along with 1726 defective ones. The subsequent table 4.3.1 provides a concise statistical overview of the data per category observed within the dataset:

Table 4.3.1: Data distribution of the MVTec AD dataset [9]

	Category	Train	Test (good)	Test (defective)	Defect groups	Defect regions	Image side length
Textures	Carpet	280	28	89	5	97	1024
	Grid	264	21	57	5	170	1024
	Leather	245	32	92	5	99	1024
	Tile	230	33	84	5	86	840
	Wood	247	19	60	5	168	1024
Objects	Bottle	209	20	63	3	68	900
	Cable	224	58	92	8	151	1024
	Capsule	219	23	109	5	114	1000
	Hazelnut	391	40	70	4	136	1024
	Metal Nut	220	22	93	4	132	700
	Pill	267	26	141	7	245	800
	Screw	320	41	119	5	135	1024
	Toothbrush	60	12	30	1	66	1024
	Transistor	213	60	40	4	44	1024
	Zipper	240	32	119	7	177	1024
Total		3629	467	1258	73	1888	-

4.3.0.2 BTAD Dataset

The dataset comprises RGB images depicting three different industrial products. The first product's image has dimensions of 1600 × 1600 pixels, the second product's image is sized at 600 × 600 pixels, and the third product's image measures 800 × 600 pixels. Each product category consists of a varying number of training images: Product 1 has been trained using a set of 400 images, Product2 with 1000 images, and Product3 with 399 images.

During the training process, all input images were initially scaled to a standardized size of 256 before being fed into the model for further processing. Additionally, in order to facilitate evaluation during testing or validation stages, a precise pixel-based ground truth mask was provided for each anomalous test image in this dataset.

4.3.0.3 ECAD Dataset

Our contribution, the newly proposed Electric Connector Anomaly detection (ECAD) dataset, comprises 1002 images, with 611 of them being anomaly-free and the remaining 391 for both test and ground truth images being anomalous. The dataset comprises RGB images depicting four distinct industrial products, namely connectors C1, C2, C3, and C4. These images possess a dimension of 1024x1024 pixels each. The ultimate goal of proposing this dataset is to detect the missing components. The training set encompasses 140, 138, 166 and 167 images for the respective categories - C1, C2, C3, and C4. On the other hand, the test set is composed of a total number of anomalies that includes both anomalous as well as anomaly-free instances from all four product categories with image counts totalling to-87(C1),101(C2),103(C3)and 100(C4) finally. We have used IPEVO, V4K Ultra High Definition USB Document Camera to create this dataset. We have hosted the dataset on Kaggle under the name Electric Connectors Anomaly Detection [3].

All the images were resized to 256x256, using only anomaly-free samples for training. Each category's training set has normal samples, while the test set contains both

[3] `https://www.kaggle.com/datasets/nagajyothirmayee/electricconnectorsanomalydetection`

anomalous and normal samples, making it challenging for anomaly detection. The subsequent table 4.3.2 provides a concise statistical overview of the data per category observed within the dataset:

Table 4.3.2: Data distribution of the proposed Electric Connectors Anomaly Detection (ECAD) dataset

Category	Train (good)	Test (good)	Test (defective)	Defect groups	Image side length
Connector 1 (C1)	140	15	72	4	1024
Connector 2 (C2)	138	15	86	4	1024
Connector 3 (C3)	166	15	88	4	1024
Connector 4 (C4)	167	31	69	4	1024
Mean	611	76	315	-	-

Figure 4.3.1 represents the sample images from the newly proposed ECAD dataset. This image portrays the various categories of images present in the ECAD dataset. Column (a) illustrates images that are devoid of any anomalies or issues. As we shift our attention to Column (b), an image is presented where a rubber gourmet is absent, indicating anomalous characteristics. Moving forward to examine Column (c), another instance of an anomalous image arises, revealing multiple missing components within it. Continuing further towards Column (d), an abnormality emerges as there is no presence of the Orange clamp component shown within the image. Lastly, taking into account column(e), yet another example of an anomalous image can be observed which encompasses both missing components and additional spots.

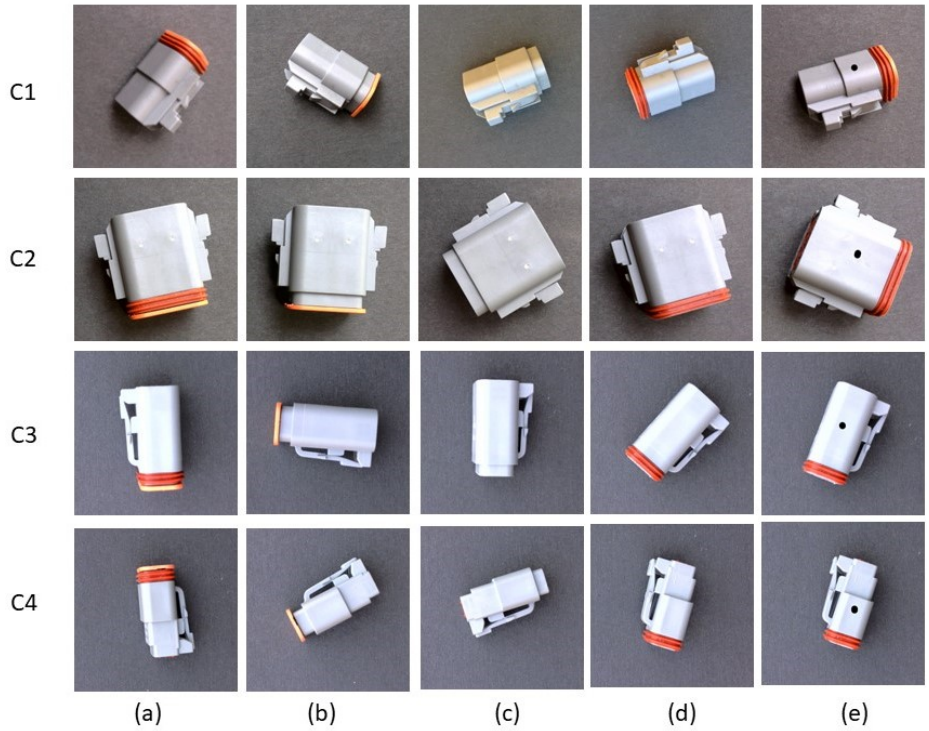

Figure 4.3.1: The image displays the four different types of images found within the ECAD dataset. In Column (a), we can observe anomaly-free images with no issues. Moving on to Column (b), an anomalous image is shown where a rubber gourmet is missing. In Column (c), another anomalous image depicts multiple components that are missing. Continuing to Column (d), an abnormality is seen where an Orange clamp component is absent from the image. Finally, in column (e), we have yet another anomalous image which not only has missing components but also contains additional spots.

4.4 Evaluation Metrics

4.4.1 Quantitative Evaluation

AUROC : The measure AUROC, which is an acronym for "Area under the ROC Curve," quantifies the complete two-dimensional region that lies beneath the entire ROC curve. In other words, it computes the integral of this curve from coordinates$(0, 0)$ to akin $(1, 1)$ to how one would calculate integrals in calculus. This metric serves as a way to assess and summarize the overall performance of a classification model based on its true positive rate and false positive rate.

ROC Curve and its Parameters: The ROC curve plots the True Positive Rate (TPR), also known as recall, against the False Positive Rate (FPR) at different classification thresholds. It helps us visualize how the model's performance changes as

we adjust the threshold for classification.

True Positive Rate (TPR): TPR is calculated as the ratio of True Positives (TP) to the total number of actual positive examples (P):
True Positive Rate (TPR) is defined in the equation 1 :

$$TPR = \frac{\text{True Positives (TP)}}{\text{Total Actual Positives (P)}} \tag{1}$$

False Positive Rate (FPR): FPR is calculated as the ratio of False Positives (FP) to the total number of actual negative examples (N):
False Positive Rate (FPR) is defined in the equation 2 :

$$FPR = \frac{\text{False Positives (FP)}}{\text{Total Actual Negatives (N)}} \tag{2}$$

Creating the ROC Curve: To construct the ROC curve, we evaluate the model at different classification thresholds. At each threshold, the model classifies some examples as positive and others as negative, leading to varying TPR and FPR values.

Computing AUROC: AUROC is obtained by calculating the area under the ROC curve, ranging from (0,0) to (1,1). It is computed using the trapezoidal rule or other numerical integration methods.

Interpretation of AUROC: AUROC ranges from 0 to 1, where 0 indicates that the model's predictions are entirely wrong, and one means the model's predictions are perfect. A higher AUROC value implies better model performance in distinguishing between positive and negative examples.
We assess our proposed method's quantitative performance using the AUROC metric. We further discuss the performance of each dataset in Section 5.1.

4.4.2 Computational Efficiency

The computational efficiency of our proposed architecture can be evaluated by considering several metrics. Firstly, the model size is measured in megabytes, which indicates the amount of memory required to store the model parameters. Secondly, we assess the training time and inference time in seconds to measure how quickly our system can process data during the training and deployment phases, respectively.

In addition, we evaluate the inference speed by calculating the number of frames processed per second using Equation 3. This metric provides insight into how efficiently our architecture performs real-time predictions. Another important factor contributing to computational efficiency is determining the number of parameters present in our framework. By comparing these metrics with a baseline model like UTRAD, which utilizes a U-shaped reconstruction structure along with skip connections, we ensure a fair evaluation and demonstrate that our approach outperforms state-of-the-art methods. Further analysis on performance will be elaborated upon in Section 5.2.

$$\text{FPS} = \frac{\text{Number of Images Processed}}{\text{Total Time Taken}} \tag{3}$$

4.4.3 Statistical Tests

Wilcoxon signed-rank test: According to [21], Wilcoxon signed-rank test is the appropriate test to measure the statistical difference between the two models. The Wilcoxon signed-rank test is a non-parametric statistical test used to compare two paired or matched samples and determine if their medians differ significantly. It is particularly useful when the underlying data does not follow a normal distribution or when the assumption of normality is not met.

In many cases, researchers are interested in investigating whether there is a significant difference between two related groups or conditions. The Wilcoxon signed-rank test addresses this question by focusing on the differences between paired observations from the two samples. The test procedure involves the following steps:

1. Formulate Hypotheses: The null hypothesis (H_0) states that there is no significant difference between the paired samples, meaning the median difference is zero. The alternative hypothesis (H_1) states that there is a significant difference between the paired samples, indicating that the median difference is not zero.

2. Calculate Differences: For each pair of data points in the two samples, calculate the differences ($d_i = x_i - y_i$), where x_i and y_i are the data points from the two related samples.

3. Rank Absolute Differences: Sort the absolute differences obtained in Step 2 in ascending order without considering the sign. Assign ranks to the sorted absolute differences, starting from 1 for the smallest absolute difference. For tied absolute differences, calculate the average rank they would receive if there were no ties.

4. Calculate the Test Statistic (W): The test statistic (W) is the sum of the ranks of positive (or negative) differences, whichever is smaller. If there are no tied absolute differences, calculate W as the sum of the ranks of positive differences. If there are tied absolute differences, consider the sum of ranks for both positive and negative differences and choose the smaller sum. The formula for calculating the test statistic is as follows:

$$W = \min\left(\sum \text{Rank}(d_i^+), \sum \text{Rank}(d_i^-)\right)$$

where: - d_i^+ are the positive differences ($d_i > 0$). - d_i^- are the negative differences ($d_i < 0$). - $\text{Rank}(d_i^+)$ is the rank of the i-th positive difference. - $\text{Rank}(d_i^-)$ is the rank of the i-th negative difference.

5. Find the Critical Value or P-Value: Determine the critical value or p-value associated with the test statistic (W) based on the size of your sample and chosen significance level (α). The p-value represents the probability of obtaining a test statistic as extreme as the observed one, assuming the null hypothesis is true. Lower p-values indicate stronger evidence against the null hypothesis.

6. Compare the p-value with the Significance Level: Compare the calculated p-value with the chosen significance level (α). If the p-value is less than or equal to α,

reject the null hypothesis (H_0) in favour of the alternative hypothesis (H_1), indicating a significant difference between the paired samples. If the p-value is greater than α, fail to reject the null hypothesis, suggesting no significant difference between the paired samples.

In summary, the Wilcoxon signed-rank test provides a robust and reliable method for comparing two paired samples without making assumptions about the data's underlying distribution. It is widely used in various fields, such as medical research, psychology, and social sciences, where normality assumptions may not be valid or critical.

Chapter 5

Discussions, Comparisons and Analysis

5.1 Quantitative Analysis

On the MVTec AD dataset, our proposed method, the M2O Transformer, was compared to several state-of-the-art models such as GANomaly, UTRAD, PaDiM, DifferNet, KDAD, f-AnoGAN, Patch SVDD, and TrustMAE, in terms of image-level anomaly classification performance in Table 5.1.1. The image-level anomaly score was determined by selecting the maximum value from the score map. Our M2O Transformer demonstrated the best overall performance, achieving an average AUROC of 97.25% with a standard deviation of ±0.3, based on five runs. It outperformed other methods in most categories effectively.

Table 5.1.1: Comparison of AUROC (%) results with other methods on MVtec AD at Image-level

Methods	Texture					Object										Mean		
	Carpet	Grid	Leather	Tile	Wood	Bottle	Cable	Capsule	Hazelnut	MetalNut	Pill	Screw	Toothbrush	Transistor	Zipper	Texture mean	Object mean	Mean
f-AnoGAN [58]	56.57	59.63	62.50	61.34	75.00	91.36	76.37	72.79	63.15	59.7	64.07	50.00	67.31	77.92	50.00	63.01	67.27	65.85
GANomaly [5]	84.20	74.30	79.20	79.50	65.30	89.20	73.20	70.80	79.40	74.50	75.70	69.90	70.00	74.60	83.40	76.50	76.07	76.21
KDAD [56]	80.46	78.01	95.05	91.57	94.29	99.39	89.19	80.46	73.58	73.58	82.70	83.31	92.17	85.55	93.24	87.88	85.32	86.20
TrustMAE [60]	**97.43**	99.08	95.07	97.29	99.82	96.98	85.06	78.82	98.50	76.10	83.31	82.37	96.94	87.50	87.54	97.74	87.31	90.79
Patch SVDD [79]	92.90	94.60	90.90	97.80	96.50	98.60	90.30	76.70	92.00	94.00	86.10	81.30	**100**	91.50	97.90	94.54	90.84	92.10
DifferNet [53]	84.00	97.10	99.40	92.90	**99.80**	99.00	86.90	88.80	91.10	95.10	95.90	**99.30**	96.10	96.30	98.60	94.60	94.70	94.70
PaDiM [20]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	98.80	93.60	95.30
UTRAD [15]	96.30	98.70	100	**99.90**	99.70	100	98.60	94.30	**99.50**	96.20	94.20	88.30	78.90	96.40	98.60	**98.90**	94.50	96.00
M2O (ours)	95.20	**99.60**	**100**	98.10	98.90	**100**	97.60	**96.60**	97.10	**99.60**	**96.20**	89.70	91.30	93.80	**99.50**	98.36	**96.20**	**97.25**

Table 5.1.2: Comparison of AUROC (%) results with other methods on MVtec AD at Pixel-level

Methods	Texture					Object										Mean
	Carpet	Grid	Leather	Tile	Wood	Bottle	Cable	Capsule	Hazelnut	Metal_nut	Pill	Screw	Toothbrush	Transistor	Zipper	Mean
AE(SYM) [10]	87.00	94.00	78.00	59.00	73.00	93.00	82.00	94.00	97.00	89.00	91.00	96.00	92.00	90.00	88.00	87.00
AE(l2) [10]	59.00	90.00	75.00	51.00	73.00	86.00	86.00	88.00	95.00	86.00	85.00	96.00	93.00	86.00	77.00	82.00
AnoGAN [57]	54.00	58.00	64.00	50.00	62.00	86.00	78.00	84.00	87.00	76.00	87.00	80.00	90.00	80.00	78.00	74.00
GANomaly [5]	55.00	80.00	77.00	69.00	91.00	82.00	83.00	72.00	86.00	69.00	76.00	72.00	82.00	79.00	84.00	77.00
VEVAE [37]	78.00	73.00	95.00	80.00	77.00	87.00	90.00	74.00	98.00	94.00	83.00	97.00	94.00	93.00	78.00	86.00
P-Net[81]	57.00	98.00	89.00	**97.00**	**98.00**	99.00	70.00	84.00	97.00	79.00	91.00	**100**	**99.00**	82.00	90.00	89.00
TrustMAE [60]	98.53	97.45	98.05	82.48	92.62	93.39	92.85	87.42	98.51	91.76	89.90	97.63	98.10	92.72	97.76	93.94
patch SVDD[79]	98.10	96.80	95.80	92.60	96.20	97.50	97.40	98.00	95.10	95.70	91.40	98.10	97.00	90.80	95.10	95.70
PaDiM [20]	**98.90**	94.90	99.10	91.20	93.60	98.10	95.80	98.30	97.70	96.70	94.70	97.40	98.70	97.20	98.20	96.70
SPADE [18]	97.50	93.70	97.60	87.40	88.50	**98.40**	97.20	**99.00**	**99.10**	**98.10**	96.50	98.90	97.90	94.10	96.50	96.00
UTRAD [15]	97.30	97.60	98.60	95.00	93.10	95.90	**97.30**	97.80	98.40	95.00	97.50	97.80	96.20	94.90	97.90	96.70
M2O (ours)	97.60	**98.10**	**99.20**	94.70	94.20	96.20	96.50	98.30	98.40	95.00	**97.80**	97.40	97.00	93.40	**98.40**	**96.81**

Table 5.1.2 and Table 5.1.3 present the anomaly localization performance on the original MVTec AD and BTAD datasets, respectively. Our proposed M2O model achieves state-of-the-art performance compared to other methods on both datasets. Our approach excels in various categories, showcasing its efficacy in both texture and object classifications by skillfully integrating local features and global semantics using a comprehensive hierarchical structure. Nonetheless, it does encounter unexpected false positives in the screw & toothbrush category because of insignificant imperfections within the background, rendering it susceptible to minor anomalies. These results indicate our model outperformed existing models with better AUROC.

5.2 Computational Analysis

The efficiency of the M2O model in terms of computation is compared with the UTRAD model. One of our research goals is to optimize the U-shaped network,

Table 5.1.3: Comparison of AUROC (%) results with other methods on BTAD at Pixel-level.

	Model Configuration				
	VT-ADL[40]	SPADE[18]	Patch SVDD [79]	UTRAD [15]	M2O (ours)
Class1	**99.0**	97.3	91.6	94.7	96.6
Class2	94.0	94.4	93.6	95.2	**95.8**
Class3	77.0	99.1	91.0	99.2	**99.5**
Mean	90.0	96.9	92.1	96.4	**97.3**

Table 5.1.4: Effect of each module in the proposed Architecture.

	Comparison with the Baseline model			
	UTRAD [15]	M2O(ours)	Improvements	
Model Size	207.00	145.95	**29.52%**	↓
No of Parameters	51754560.0	35738688.0	**30.93%**	↓
Inference Time	0.13	0.12	**8.79%**	↓
Inference Speed	7.52	8.28	**10.1%**	↑

Table 5.1.5: Comparison of AUROC (%) results with UTRAD on ECAD at Image and Pixel-level.

	Model Configuration			
	UTRAD$_I$ [15]	M2O$_I$	UTRAD$_P$ [15]	M2O$_P$
C1	97.0	100.0	92.6	94.3
C2	99.8	100.0	85.1	85.6
C3	100.0	100.0	94.7	95.2
C4	99.8	95.7	98.2	96.5

which we have addressed by introducing the proposed MLFF layer. A comparison of the training time computational efficiency of M2O and UTRAD on screw datatype of MVTec AD is shown in Figure 5.2.1, which depicts M2O takes 30% less time than [15]. Furthermore, Table 5.1.4 demonstrates that the model size has decreased by 29.52%, the inference speed of M2O is 10.1%, the inference time has reduced by 8.79%, and No of parameters in the network is reduced by 30.93 %.

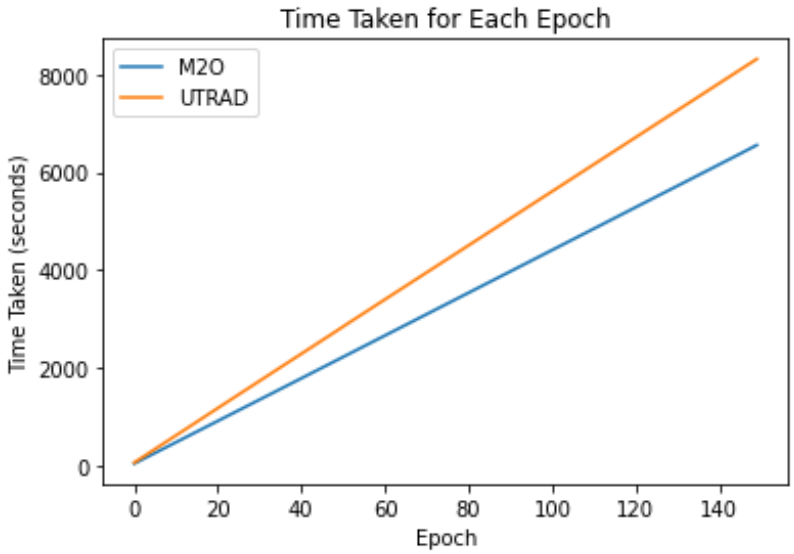

Figure 5.2.1: M2O vs UTRAD - Computation Time on Screw data in MVTec AD

In addition, we performed anomaly detection and localization on the proposed ECAD dataset, yielded exceptional results. Table 5.1.5 presents a detailed comparison of these two models. In the table, we denote "_I" and "_P" to represent the results at both image and pixel levels for their respective models. We observed that model M2O achieved comparable performance with the U Shaped network UTRAD, which is significant as it validates our initial contribution.

Therefore, the strength and reliability of our approach in effectively managing unforeseen variations in test data indicate its applicability to real-world scenarios. In conclusion, we did not need to adjust the hyperparameters for training and validating various datasets. As a result, we assert that our model demonstrates excellent generalization capabilities without the need for parameter tuning.

5.3 Statistical Analysis

Wilcoxon signed-rank test [49]:

To assess the performance of both the UTRAD model and our M2O Transformer Model, we employed the Wilcoxon signed-rank test. In line with our previous findings, wherein network optimization had no effect on quantitative performance compared to a U-shaped network architecture, it is anticipated that the results of this test will not reject the null hypothesis. The null hypothesis states that there is no significant disparity in performance between these two models. Here are our defined null and alternate hypotheses for reference:

Null Hypothesis (H_0)*:* There is no significant difference in performance between the UTRAD [15] and proposed M2O models.

Alternative Hypothesis (H_1)*:* There is a significant difference in performance between the UTRAD [15] and proposed M2O models. Let us consider the results of the MVTec AD dataset at Image-Level: UTRAD [15] model = 96.30, 98.70, 100, 99.90, 99.70, 100, 98.60, 94.30, 99.50, 96.20, 94.20, 88.30, 78.90, 96.40, 98.60

M2O model = 95.20, 99.60, 100, 98.10, 98.90, 100, 97.60, 96.60, 97.10, 99.60, 96.20, 89.70, 91.30, 93.80, 99.50

For the given data, Since the test statistic (T = 10) is equal to the critical value (10) at the 0.01 significance level, we fail to reject the null hypothesis (H0). There is not enough evidence to conclude that there is a statistically significant difference in performance between the old and new models at the 0.01 significance level.

In summary, based on the Wilcoxon signed-rank test, we do not find a significant difference in performance between the UTRAD [15] and proposed M2O models at the 0.01 significance level. Hence, Our results depicts that the M2O Transformer Model performs similarly to or slightly better than the UTRAD Model[15].

Table 5.3.1: Effect of each module in the proposed Architecture.

Combination	Model Configuration					Image AUROC	Pixel AUROC
	GPE	RPE	Loss Scale	Bottle Neck	Smoothing		
1	**X**	✓	✓	✓	✓	96.12	95.87
2	✓	**X**	✓	✓	✓	94.66	95.94
3	✓	✓	**X**	✓	✓	87.99	92.41
4	✓	✓	✓	**X**	✓	95.04	95.98
5	✓	✓	✓	✓	**X**	93.29	94.70
6	✓	✓	✓	✓	✓	**96.81**	**97.25**

5.4 Ablation Studies

An ablation study was conducted to assess the individual contributions of different modules in the M2O model. The results showed that the complete M2O configuration outperformed other variations and demonstrated the effectiveness of all its modules. Furthermore, a three-level M2O model achieved better performance compared to a two-level one, achieving mean AUROC scores of 97.25% (pixel-level) and 96.81% (image-level). It was evident that all modules had a positive impact on performance. Notably, when comparing full M2O with partial configurations, there were substantial improvements observed in terms of AUROC increase from 0.72% to 1.76% at pixel level and from 1.32% to 5.17% at image level, respectively. The outcomes for each of the seven configurations are illustrated in Table 5.3.1. Each title within the table indicates which component was removed for that specific configuration.

Chapter 6

Conclusion, Limitations and Future Works

6.1 Conclusion

Our research addresses the computational limitations of existing state-of-the-art models for anomaly detection and localization through the proposed MLFF module. Furthermore, our evaluation metrics show that the proposed M2O model performs better in solving the problem of anomaly detection and localization, which is demonstrated by comparing it with multiple state-of-the-art methods. Our proposed architecture demonstrates its generalization capabilities and improved robustness by being able to handle diverse data types without the need for explicit hyperparameter tuning. To the best of our knowledge, we are the first one to present a dataset specifically for Electric Connectors, ECAD, which serves as a benchmark and is applicable to real-world applications.

The proposed architecture demonstrated significant improvements in training time compared to UTRAD on various datasets such as MVTec AD, ECAD, and BTAD. Specifically, the proposed architecture achieved a 30% reduction in training time and a decrease of 29.52% in model size. Moreover, it also resulted in a reduction of approximately 8.8% inference time and an improved inference speed of 10.1%, all while maintaining the promised AUROC performance outlined in the thesis contribution. Furthermore, there was an improvement observed at the image level which amounted to an increase of 1.3021%. These outcomes clearly highlight how effective the new

architecture is when compared to its predecessor. Overall, these findings demonstrate that by leveraging this novel architectural design approach effectively decreases training time without compromising performance metrics like AUROC.

6.2 Limitations and Future Work

One of the main constraints we encountered when working with grayscale images was the performance issues, which in turn led to lower prediction scores for both anomaly detection and localization. This limitation can be attributed to the utilization of the tanh function, which tends to suppress a significant number of pixels that could potentially contain shades of gray. Consequently, this suppression results in approximation errors and subsequently leads to poor estimation accuracy. To address these performance challenges associated with grayscale imagery in anomaly detection and localization tasks, it is advisable to explore additional techniques related to feature aggregation. These approaches may involve reconsidering the activation functions utilized or even employing alternative methods for aggregating features.

Moreover, this approach is susceptible to interference from various sources of noise, such as dust particles or random blemishes in the input image. It would be beneficial to address this issue by implementing a sensitivity control mechanism. In order to enhance the efficacy of the MLFF layer within our model, it may be advantageous to explore Convolutional Neural Network settings. Experiment with a U-shaped network based on Convolutional Neural Network architecture by MLFF utilization and conduct further experimentation. Additionally, since the current model operates under unimodal distributions, there is potential for expansion into a classifier capable of handling multi-modalities. Building upon these advancements, an interesting research direction involves extending this approach into either a continual learning model or a lifelong learning model.

BIBLIOGRAPHY

[1] (2023). National highway traffic safety administration. RMISC-23V131-8287.pdf, Accessed 7 August 2023.

[2] Abe, N., Zadrozny, B., and Langford, J. (2006). Outlier detection by active learning. In *Proceedings of the 12th ACM SIGKDD*.

[3] Abraham, B. and Box, G. E. P. (1979). Bayesian analysis of some outlier problems in time series. *Biometrika*, 66(2):229–236.

[4] Ajithaprasad, S., Velpula, R., and Gannavarpu, R. (2019). Defect detection using windowed fourier spectrum analysis in diffraction phase microscopy. *Journal of Physics Communications*, 3(2):025006.

[5] Akcay, S., Atapour-Abarghouei, A., and Breckon, T. P. (2019). Ganomaly: Semi-supervised anomaly detection via adversarial training. In Jawahar, C. V., Li, H., Mori, G., and Schindler, K., editors, *Computer Vision – ACCV 2018*, pages 622–637, Cham. Springer International Publishing.

[6] Akçay, S., Atapour-Abarghouei, A., and Breckon, T. P. (2019). Skip-ganomaly: Skip connected and adversarially trained encoder-decoder anomaly detection. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

[7] Anscombe, F. J. and Guttman, I. (1960). Rejection of outliers. *Technometrics*, 2(2):123–147.

[8] Bai, X., Fang, Y., Lin, W., Wang, L., and Ju, B.-F. (2014). Saliency-based defect

detection in industrial images by using phase spectrum. *IEEE Transactions on Industrial Informatics*, 10(4):2135–2145.

[9] Bergmann, P., Fauser, M., Sattlegger, D., and Steger, C. (2019a). Mvtec ad – a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[10] Bergmann, P., Löwe, S., Fauser, M., Sattlegger, D., and Steger, C. (2019b). Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP,*, pages 372–380. INSTICC, SciTePress.

[11] Chalapathy, R. and Chawla, S. (2019). Deep learning for anomaly detection: A survey. *ACM Computing Surveys*, 51(3):1–37.

[12] Chandola, V., Banerjee, A., and Kumar, V. (2009a). Anomaly detection. *ACM Computing Surveys*, 41(3):1–58.

[13] Chandola, V., Banerjee, A., and Kumar, V. (2009b). Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3).

[14] Chawla, N. V., Japkowicz, N., and Kotcz, A. (2004). Editorial: special issue on learning from imbalanced data sets. *SIGKDD Explorations*, 6(1):1–6.

[15] Chen, L., You, Z., Zhang, N., Xi, J., and Le, X. (2022). Utrad: Anomaly detection and localization with u-transformer. *Neural Netw.*, 147(C):53–62.

[16] Chen, Q., Wang, Y., Yang, T., Zhang, X., Cheng, J., and Sun, J. (2021). You only look one-level feature. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13034–13043.

[17] CNBC (2021). Tesla recalling up to 7,696 vehicles over seatbelt issues in us. `https://www.cnbc.com/2021/06/03/tesla-recalling-up-to-7696-vehicles-over-seatbelt-issues-in-us.html`.

[18] Cohen, N. and Hoshen, Y. (2021). Sub-image anomaly detection with deep pyramid correspondences.

[19] Cui, Y., Liu, Z., and Lian, S. (2023). A survey on unsupervised anomaly detection algorithms for industrial images. *IEEE Access*, 11:55297–55315.

[20] Defard, T., Setkov, A., Loesch, A., and Audigier, R. (2021). Padim: A patch distribution modeling framework for anomaly detection and localization. In Del Bimbo, A., Cucchiara, R., Sclaroff, S., Farinella, G. M., Mei, T., Bertini, M., Escalante, H., and Vezzani, R., editors, *Pattern Recognition. ICPR International Workshops and Challenges*, pages 475–489, Cham. Springer International Publishing.

[21] Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7(1):1–30.

[22] Desforges, M., Jacob, P., and Cooper, J. (1998). Applications of probability density estimation to the detection of abnormal conditions in engineering. In *Proceedings of the Institute of Mechanical Engineers*, volume 212, pages 687–703.

[23] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

[24] Eskin, E. (2000). Anomaly detection over noisy data using learned probability distributions. In *Proceedings of the 17th International Conference on Machine Learning*, pages 255–262. Morgan Kaufmann Publishers Inc.

[25] Gudovskiy, D., Ishizaka, S., and Kozuka, K. (2022). Cow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows.

In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 98–107.

[26] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[27] Hu, G.-H. (2015). Automated defect detection in textured surfaces using optimal elliptical gabor filters. *Optik*, 126(14):1331–1340.

[28] Huang, H., Lin, L., Tong, R., Hu, H., Zhang, Q., Iwamoto, Y., Han, X., Chen, Y.-W., and Wu, J. (2020). Unet 3+: A full-scale connected unet for medical image segmentation. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1055–1059.

[29] Jing, J., Yang, P., Li, P., and Kang, X. (2014). Supervised defect detection on textile fabrics via optimal gabor filter. *Journal of Industrial Textiles*, 44(1):40–57.

[30] Joshi, M. V., Agarwal, R. C., and Kumar, V. (2001). Mining needle in a haystack: classifying rare classes via two-phase rule induction. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 91–102. ACM Press.

[31] Kingma, D. and Ba, J. (2015). Adam: A method for stochastic optimization.

[32] Kumaresan, P. (2017). Defect detection in texture by fourier analysis approach. *International Journal of Engineering, Science and Mathematics*, 6(3):124–130.

[33] Kwon, G., Prabhushankar, M., Temel, D., and AlRegib, G. (2020). Backpropagated gradient representations for anomaly detection. In Vedaldi, A., Bischof, H., Brox, T., and Frahm, J.-M., editors, *Computer Vision – ECCV 2020*, pages 206–226, Cham. Springer International Publishing.

[34] Li, P., Zhang, H., Jing, J., Li, R., and Zhao, J. (2015). Fabric defect detection based on multi-scale wavelet transform and gaussian mixture model method. *The Journal of The Textile Institute*, 106(6):587–592.

[35] Li, X., Chen, H., Qi, X., Dou, Q., Fu, C.-W., and Heng, P.-A. (2018). H-denseunet: Hybrid densely connected unet for liver and tumor segmentation from ct volumes. *IEEE Transactions on Medical Imaging*, 37(12):2663–2674.

[36] Liu, J., Xu, G., Ren, L., Qian, Z., and Ren, L. (2017). Defect intelligent identification in resistance spot welding ultrasonic detection based on wavelet packet and neural network. *The International Journal of Advanced Manufacturing Technology*, 90(9):2581–2588.

[37] Liu, W., Li, R., Zheng, M.and Karanam, S., Wu, Z.and Bhanu, B., Radke, R. J., and Camps, O. (2020). Towards visually explaining variational autoencoders. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[38] Liu, Y., Zhuang, C., and Lu, F. (2021). Unsupervised two-stage anomaly detection. *arXiv preprint arXiv:2103.11671*.

[39] Massoli, F. V., Falchi, F., Kantarci, A., Akti, S., Ekenel, H. K., and Amato, G. (2020). Mocca: Multi-layer one-class classification for anomaly detection. *arXiv preprint arXiv:2012.12111*.

[40] Mishra, P., Verk, R., Fornasier, D., Piciarelli, C., and Foresti, G. L. (2021). Vt-adl: A vision transformer network for image anomaly detection and localization. In *2021 IEEE 30th International Symposium on Industrial Electronics (ISIE)*, pages 01–06.

[41] Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N. Y., Kainz, B., Glocker, B., and Rueckert, D. (2018). Attention u-net: Learning where to look for the pancreas. In *Medical Imaging with Deep Learning*.

[42] Pang, G., Shen, C., Cao, L., and Hengel, A. V. (2021a). Deep learning for anomaly detection: A review. *ACM Computing Surveys*, 54(2):1–38.

[43] Pang, G., Shen, C., Cao, L., and Hengel, A. V. D. (2021b). Deep learning for anomaly detection: A review. *ACM Comput. Surv.*, 54(2).

[44] Parzen, E. (1962). On the estimation of a probability density function and mode. *Annals of Mathematical Statistics*, 33:1065–1076.

[45] Pastor-Lopez, I., Sanz, B., Puerta, J., and Bringas, P. (2019). Surface defect modelling using co-occurrence matrix and fast fourier transformation. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 745–757. Springer.

[46] Phua, C., Alahakoon, D., and Lee, V. (2004). Minority report in fraud detection: Classification of skewed data. *SIGKDD Explorer Newsletter*, 6(1):50–59.

[47] Pirnay, J. and Chai, K. (2021). Inpainting transformer for anomaly detection. *arXiv preprint arXiv:2104.13897*.

[48] Ravande, S. (2022). Council post: Unplanned downtime costs more than you think. *Forbes*.

[49] Rey, D. and Neuhäuser, M. (2011). *Wilcoxon-Signed-Rank Test*, pages 1658–1659. Springer Berlin Heidelberg, Berlin, Heidelberg.

[50] Rippel, O., Mertens, P., and Merhof, D. (2021). Modeling the distribution of normal data in pre-trained deep features for anomaly detection. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 6726–6733, Los Alamitos, CA, USA. IEEE Computer Society.

[51] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F., editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham. Springer International Publishing.

[52] Rudolph, M., Wandt, B., and Rosenhahn, B. (2021a). Same same but different: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1907–1916.

[53] Rudolph, M., Wandt, B., and Rosenhahn, B. (2021b). Same same but differnet: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1907–1916.

[54] Rudolph, M., Wehrbein, T., Rosenhahn, B., and Wandt, B. (2022). Fully convolutional cross-scale flows for image-based defect detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1088–1097.

[55] Salehi, M., Eftekhar, A., Sadjadi, N., Rohban, M. H., and Rabiee, H. R. (2022). Puzzle-ae: Novelty detection in images through solving puzzles.

[56] Salehi, M., Sadjadi, N.and Baselizadeh, S., Rohban, M. H., and Rabiee, H. R. (2021). Multiresolution knowledge distillation for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14902–14912.

[57] Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U., and Langs, G. (2017). Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In Niethammer, M., Styner, M., Aylward, S., Zhu, H., Oguz, I., Yap, P.-T., and Shen, D., editors, *Information Processing in Medical Imaging*, pages 146–157, Cham. Springer International Publishing.

[58] Schlegl, T., Seeböck, P., M. Waldstein, S., Langs, G., and Schmidt-Erfurth, U. (2019). f-anogan: Fast unsupervised anomaly detection with generative adversarial networks. *Medical Image Analysis*, 54:30–44.

[59] Steinwart, I., Hush, D., and Scovel, C. (2005). A classification framework for anomaly detection. *Journal of Machine Learning Research*, 6:211–232.

[60] Tan, D. S., Chen, Y.-C., Chen, T. P.-C., and Chen, W.-C. (2021). Trustmae: A noise-resilient defect classification framework using memory-augmented auto-

encoders with trust regions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 276–285.

[61] Tao, X., Gong, X., Zhang, X., Yan, S., and Adak, C. (2022). Deep learning for unsupervised anomaly localization in industrial images: A survey. *IEEE Transactions on Instrumentation and Measurement*, 71:1–21.

[62] Theiler, J. and Cai, D. M. (2003). Resampling approach for anomaly detection in multispectral images. In *Proceedings of the SPIE*, volume 5093, pages 230–240.

[63] TM, P. (2023). Python 3.8.10. `https://www.python.org/downloads/release/python-3810`.

[64] Tong, L., Wong, W., and Kwong, C. (2016). Differential evolution-based optimal gabor filter model for fabric inspection. *Neurocomputing*, 173:1386–1401.

[65] U.S. Consumer Product Safety Commission (2023). Advanced ev recalls advent 4 and 6 passenger golf carts due to fall and injury hazards - recall alert. `https://www.cpsc.gov/Recalls/2023/Advanced-EV-Recalls-Advent-4-and-6-Passenger-Golf-Carts-Due-to-Fall-and-Injury-Hazards-Recall-Alert`.

[66] Venkataramanan, S., Peng, K.-C., Singh, R., and Mahalanobis, A. (2020). Attention guided anomaly localization in images. In *European Conference on Computer Vision*, pages 485–503. Springer.

[67] Vijaykumar, V. and Sangamithirai, S. (2015). Rail defect detection using gabor filters with texture analysis. In *2015 3rd International Conference on Signal Processing*, page ...

[68] Vilalta, R. and Ma, S. (2002). Predicting rare events in temporal domains. In *Proceedings of the IEEE International Conference on Data Mining*, page 474. IEEE Computer Society.

[69] Volansys (2023). Edge ml for production line quality. `https://www.volansys.com/blog/edge-ml-for-production-line-quality/`.

[70] Wang, G., Han, S., Ding, E., and Huang, D. (2021). Student-teacher feature pyramid matching for unsupervised anomaly detection. *CoRR*, abs/2103.04257.

[71] Wang, H., Cao, P., Wang, J., and Zaiane, O. R. (2022). Uctransnet: Rethinking the skip connections in u-net from a channel-wise perspective with transformer. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(3):2441–2449.

[72] Wang, J.-W., Chen, W.-Y., and Lee, J.-S. (2012). Singular value decomposition combined with wavelet transform for lcd defect detection. *Electronics Letters*, 48(5):266–267.

[73] Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., and Hu, Q. (2020). Eca-net: Efficient channel attention for deep convolutional neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[74] Weiss, G. M. and Hirsh, H. (1998). Learning to predict rare events in event sequences. In Agrawal, R., Stolorz, P., and Piatetsky-Shapiro, G., editors, *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining*, pages 359–363. AAAI Press.

[75] Yamada, S. and Hotta, K. (2021). Reconstruction student with attention for student-teacher pyramid matching. *arXiv preprint arXiv:2111.15376*.

[76] Yang, C., Liu, P., Yin, G., Jiang, H., and Li, X. (2016). Defect detection in magnetic tile images based on stationary wavelet transform. *NDT & E International*, 83:78–87.

[77] Yang, J., Shi, Y., and Qi, Z. (2020). Dfr: Deep feature reconstruction for unsupervised anomaly segmentation. *arXiv preprint arXiv:2012.07122*.

[78] Yang, X., Qi, D., and Li, X. (2010). Multi-scale edge detection of wood defect images based on the dyadic wavelet transform. In *2010 International Conference on Machine Vision and Human-machine Interface*, pages 120–123. IEEE.

[79] Yi, J. and Yoon, S. (2020). Patch svdd: Patch-level svdd for anomaly detection and segmentation. In *Computer Vision – ACCV 2020: 15th Asian Conference on Computer Vision, Kyoto, Japan, November 30 – December 4, 2020, Revised Selected Papers, Part VI*, page 375–390. Springer.

[80] Zavrtanik, V., Kristan, M., and Skofcaj, D. (2021). Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, 112:107706.

[81] Zhou, K., Xiao, Y., Yang, J., Cheng, J., Liu, W., Luo, W., Gu, Z., Liu, J., and Gao, S. (2020). Encoding structure-texture relation with p-net for anomaly detection in retinal images. In Vedaldi, A., Bischof, H., Brox, T., and Frahm, J.-M., editors, *Computer Vision – ECCV 2020*, pages 360–377, Cham. Springer International Publishing.

[82] Zhou, Z., Rahman Siddiquee, M. M., Tajbakhsh, N., and Liang, J. (2018). Unet++: A nested u-net architecture for medical image segmentation. In Stoyanov, D., Taylor, Z., Carneiro, G., Syeda-Mahmood, T., Martel, A., Maier-Hein, L., Tavares, J. R., Bradley, A., Papa, J., Belagiannis, V., Nascimento, J. C., Lu, Z., Conjeti, S., Moradi, M., Greenspan, H., and Madabhushi, A., editors, *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 3–11, Cham. Springer International Publishing.

VITA AUCTORIS

NAME:	Naga Jyothirmayee Dodda
PLACE OF BIRTH:	Visakhapatnam,India
YEAR OF BIRTH:	1997
EDUCATION:	Masters in Computer Science - AI Specialized
	University of Windsor, M.Sc in Computer Science, Windsor, Ontario, 2023