Comparison of Unsupervised Image Anomaly Detection Models for Sheet Metal Glue Lines

Comparison of Unsupervised Image Anomaly Detection Models for Sheet Metal Glue Lines

Siyuan Chen[a], Sunith Bandaru[b], Silvan Marti[a], Ebru Turanoglu Bekar[a], Anders Skoogh[a]

[a]*Industrial and Materials Science at Chalmers University of Technology, Hörsalsvägen 7A, Gothenburg, SE-412 96, Sweden*
[b]*School of Engineering Science at University of Skövde, Skövde, SE-541 28, Sweden*

Abstract

Accurate anomaly detection and localization in sheet metal glue line applications is crucial for quality assurance in automotive manufacturing. Most current vision-based inspection systems that rely on geometric deviations from a predefined shape often suffer from high false-positive rates, leading to unnecessary interventions and operational inefficiencies. This research investigates the potential of unsupervised deep learning models to significantly reduce false positives in the analysis of sheet metal glue line images, even with limited datasets. We conducted a comparative evaluation of 17 unsupervised deep learning models covering different categories with 28 backbones on datasets of approximately 300 industrial glue line images per part from a Swedish vehicle manufacturer. A data synthesis method was applied to balance the glue line dataset, further enhancing the reliability of the models. To address the challenge of limited training data and improve model generalization, we incorporated data augmentation techniques and performed robustness experiments to ensure applicability to real-world industrial conditions. Our findings demonstrate that deep learning approaches can effectively detect and localize anomalies, significantly reducing false positives and gluing machine downtimes compared to the existing system. Moreover, we propose a multi-criteria decision-making based approach for model selection, enabling decision-makers to achieve optimal trade-offs between accuracy and inference time, thus improving operational efficiency. These advancements highlight that even with limited training data, unsupervised deep learning models can enhance anomaly detection reliability, streamline the automotive production process, and reduce unnecessary resource expenditures.

Keywords: Computer Vision, Anomaly Detection, Unsupervised Deep Learning, Glue Line

List of Abbreviations

AE Autoencoder

AI Artificial Intelligence

CFA Coupled-hypersphere-based Feature Adaptation

CFlow Conditional Normalizing Flow

CNN Convolutional Neural Networks

CSFlow Cross-Scale Flows

DFM Deep Feature Modeling

DRAEM Denoising Reconstruction-based Anomaly Embedding Model

DSR Deep Subspace Representation

GAN Generative Adversarial Network

NF Normalizing Flows

PaDiM Patch Distribution Modeling

SPADE Spatially-Adaptive Denormalization

STFPM Student-Teacher Feature Pyramid Matching

SVDD Support Vector Data Description

UFlow U-shaped Normalizing Flow

VAE Variational Autoencoder

YOLO You Only Look Once

1. Introduction

Glued joints are widely used in the automotive industry as they offer several advantages over traditional methods. For instance, adhesives can join different material types and do not influence the material properties as traditional methods such as riveting or welding do (Maláková et al., 2019). Adhesives need to be dispensed on the material before joining, for which robots are frequently employed (Prezas et al., 2022). However, dispensing the viscous fluid can sometimes lead to fluctuating quality. Quality assurance in automotive manufacturing is crucial. Most companies know that effective geometry assurance processes enable smooth and uninterrupted production, leading to lower costs (Söderberg et al., 2016).

Traditional quality assurance methods such as manual inspection can be laborious and susceptible to human errors. There are also researchers or industries that use image detection methods such as OpenCV (Bradski, 2000) for image detection. However, these require a large amount of various types of image data, and in practice, there is often a lack of images representing anomalies or defects. In such cases, traditional vision-based systems that rely on the detection of deviations from a predefined geometry do not perform well. In this regard, data-driven methods such as deep learning-based image anomaly detection are promising approaches to automatically supervise quality. Anomaly detection has already been widely applied in many industries ranging from automotive to lace production (Zipfel et al., 2023; Jiang et al., 2019; Tang et al., 2020; Lu et al., 2022). In this domain, the advent of deep learning has enabled a general increase in performance compared to traditional image processing approaches (Pang et al., 2021; Luo et al., 2022; Liu et al., 2024b).

Sheet metal glue line defects in the automotive context often manifest as wavelike irregularities along the adhesive line. These defects can appear as inconsistent thickness, gaps, or uneven dispersion, which can compromise the joint's structural integrity. Images capturing these glue line defects are typically gray-scale, with variations in intensity highlighting the anomalies. These characteristics pose challenges for detection and segmentation, as the subtle differences can be easily missed by traditional approaches.

Deep learning based anomaly detection algorithms can be trained in a supervised , semi-supervised or unsupervised fashion (Pang et al., 2021; Liu et al., 2024b). As with many Artificial Intelligence (AI) use cases, data collection and annotation is a problem as it can be costly and time consuming. This problem can be mitigated or circumvented using semi-supervised or unsupervised methods (Pang et al., 2021). In this regard, this study analyses a specific industrial case from an unsupervised perspective.

The primary objective of this research is to investigate the use of unsupervised deep learning models to reduce the incidence of false positives in the automated inspection system for detecting small and large anomalies in images captured by industrial cameras. Specifically, this study aims to compare various unsupervised deep learning models to identify the most effective methods for anomaly detection and segmentation in an industrial context.

This study aims to address the following research questions:

- Which unsupervised deep learning models provide the optimal accuracy and efficiency in detecting and segmenting anomalies in glue-line images from industrial cameras?

- How can the lack of defect class datasets in industry be addressed to improve model performance and reliability?

This research aims to address the current gap by offering a detailed comparison of unsupervised deep learning models, helping manufacturers choose the most suitable techniques to improve the accuracy, reliability and efficiency of their inspection processes. By reducing false positives with unsupervised deep learning techniques, this study seeks to pave the way for increased accuracy and operational efficiency in anomaly detection, then help the glue line system to avoid frequent downtime, inspection by maintenance workers and greatly improve overall equipment effectiveness of the whole process system.

The paper is organized into several sections: following this introduction, we review the relevant literature and categorize the unsupervised deep learning models, then describe the methodology and experimental setup, and present the results of the comparative analysis, discuss the implications of the findings, and conclude with recommendations for future research and implementation.

2. Related Work

In manufacturing, maintaining high-quality standards is essential to ensure product reliability and customer satisfaction, making anomaly detection techniques crucial.

2.1. Anomaly Detection in Manufacturing

Industrial anomaly detection is widely applied across various industrial data problems, including image anomaly detection and IoT time series anomalies. Anomaly detection in IoT focuses on identifying irregularities in time series data to prevent equipment failures and optimize maintenance such as for instance the work of (Weihan, 2020) or (Jeong et al., 2022). However, in this section we concentrates on image anomaly detection. The goal of image anomaly detection is to detect defects on the appearance of various types of industrial products (Luo et al., 2022). Some of these defects are small and difficult to detect, but they can be harmful for functionality of the product.

In the manufacturing industry, defects tend to appear in small regions of the image with low significance, and in turn, industrial defect detection focuses more on detecting anomalous pixels in the image (Luo et al., 2022). We review and compare selected works that evaluated models on industrial image datasets. Table 1 compares the datasets used, application domains, models employed, and training mechanisms, highlighting advancements in industrial surface inspection methodologies. It shows that image anomaly detection and quality check are widely applied in various domains.

The MVTec (Bergmann et al., 2021) dataset is a popular benchmark dataset in the industrial domain; however, people are also applying their own dataset to test their models' accuracy. For example, the study by Haselmann et al. (2018) employs decorated plastic parts on their Convolutional Neural Networks (CNN) approach, and Jiang et al. (2019) applied semi-supervised training techniques using GAN and You Only Look Once (YOLO) v3 models on their cigarette production dataset. Staar et al. (2019) and Tayeh et al. (2020) use unsupervised training with custom datasets and the MVTec AD dataset, respectively. They both employ Triplet Networks and CNNs, which demonstrates the versatility of these models in various industrial applications.

Table 1: Comparison of approaches for anomaly detection in industrial surface inspection.

Reference	Dataset	Domain	Models Used	Training
(Zipfel et al., 2023)	VIN labels	Automotive	GANomaly, PaDiM, Patchcore	Unsupervised
(Jiang et al., 2019)	Cigarette production dataset	Industrial production	GAN, YOLOv3	Semi-supervised
(Haselmann et al., 2018)	Decorated plastic parts	Manufacturing	CNN	Unsupervised
(Staar et al., 2019)	DAGM dataset	Automotive & Medical	Triplet Networks, CNN	Unsupervised
(Lu et al., 2022)	Lace video	Lace production	RNN	Unsupervised
(Posilović et al., 2022)	Ultrasonic nondestructive testing dataset	Mechanical	GANomaly, PaDiM, DifferNet	Unsupervised
(Tang et al., 2020)	Mobile phone screen glass/wood surface	Manufacturing	DAGAN	Unsupervised

This comparison underscores the importance of selecting appropriate datasets and training mechanisms tailored to specific industrial needs. It also illustrates the ongoing trend of leveraging advanced neural network architectures to improve the accuracy and reliability of anomaly detection systems in manufacturing.

2.2. Deep Learning Based Image Anomaly detection

Deep learning methods for image anomaly detection can be trained in supervised, semi-supervised, weakly-supervised and unsupervised. Supervised learning requires labeled datasets, semi-supervised learning combines a small amount of labeled data with a large amount of unlabeled data, and unsupervised learning identifies patterns in entirely unlabeled data, making these methods versatile and adaptable to various industrial scenarios.

2.2.1. Supervised Deep Learning Method

Supervised deep learning methods have a wide and mature application to industrial vision tasks, and are often used for industrial defect detection when the defect types are known and have sufficient labeled samples, or to solve the problem of classifying defect types. Li et al. (2018) improve YOLO network and made it all convolutional to provide an end-to-end solution for surface defects detection of steel strip. Chen and Tsai (2021) develop a defect detector on the basis of YOLOv3 and used densely connected convolutional networks (DenseNet) to inspect the chips of surface-mounted device light-emitting diodes (SMD LED). Božič et al. (2021) propose a deep learning architecture for surface-defect detection that reduces the need for detailed annotations by utilizing a range of supervision levels, from weak image-level labels to full pixel-level annotations, resulting in effective defect segmentation and classification. And to cope with

the problems of texture offset and partial visual confusion, Zeng et al. (2021) propose Reference-based Defect Detection Network, which introduces template references and contextual references to solve the problems respectively. Qiu et al. (2019) propose a three stages supervised deep learning method, which uses a lightweight fully convolutional network for pixel-wise defect prediction, detection to correct improper segmentation, and matting to refine defect contours using a guided filter. To balance efficiency and accuracy, the method replaces standard convolution, pooling, and deconvolution layers with depthwise & pointwise, strided depthwise, and upsample depthwise convolution layers, respectively. However, supervised deep learning methods often face the problem of not having sufficient and balanced labeling-containing datasets, and the cost of labeling is relatively high, and the problem cannot be completely solved even by using data augmentation (Luo et al., 2022).

2.2.2. Semi-supervised and Weakly-supervised Deep Learning Method

To address the challenge of limited defective samples and unbalanced data, semi-supervised deep learning methods leverage both labeled and unlabeled data to enhance anomaly detection performance. By combining the strengths of supervised learning with the abundance of unlabeled data, these models can achieve high accuracy with fewer labeled examples, making them particularly effective in scenarios where acquiring labeled data is costly or time-consuming. Chu and Kitani (2020) propose a novel semi-supervised learning algorithm for anomaly detection and segmentation that uses an anomaly classifier based on the loss profile of data processed through an autoencoder. Class activation map guided UNet used sufficient normal training images and limited annotated anomalous images to train a defect segmentation model with a feedback refinement mechanism (Lin et al., 2020).

Weakly-supervised image anomaly detection methods leverage small amounts of annotated abnormal data to enhance detection performance, providing valuable guidance even when abnormal samples are limited compared to normal ones. Methods like DevNet (Zhou et al., 2022) and approaches using Logit Inducing Loss (LIS) and Abnormality Capturing Module (ACM) demonstrate that even with coarse-grained annotations, models can achieve fine-grained detection results, comparable to fully supervised models (Wan et al., 2022). Methods for neural network interpretability are also applied in weakly supervised settings. These approaches typically train classification models using image-level annotations and then use techniques like Class Activation Mapping (CAM) and Gradient-weighted Class Activation Mapping (Grad-CAM) to identify the regions in the feature maps that contribute the most to the classification result (Zhou et al., 2016) (Selvaraju et al., 2017), thereby achieving defect localization. By focusing on these key regions, the models can effectively pinpoint anomalies even with limited annotated data, enhancing the overall performance of anomaly detection tasks in scenarios where fine-grained annotations are scarce.

2.2.3. Unsupervised Deep Learning Method

Unsupervised deep learning methods require only easily accessible normal samples for model training, eliminating the need for real defective samples. This approach not only addresses the limitation of supervised deep learning methods in identifying unknown defects but also offers a stronger representation of image features compared to traditional methods. The core idea behind these methods is to construct a "template" that closely resembles the sample being tested. By comparing this template to the sample, defects can be detected and localized based on pixel or feature differences. Depending on the comparison dimensions, unsupervised deep learning methods are generally categorized into feature-embedding based and reconstruction-based approaches

as shown at Figure 1 (Liu et al., 2024b). We will discuss the state of the art unsupervised models and compare some of them in our study and dataset.

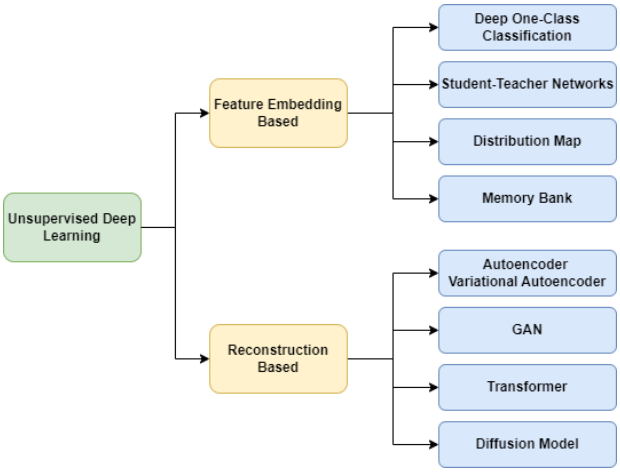


Figure 1: Unsupervised Deep Learning Models for Image Anomaly Detection Category

Feature Embedding Based Methods. Feature embedding based models aim to learn compact and informative representations of normal data. These models transform input images into a lower-dimensional feature space, capturing essential characteristics while discarding redundant information. Anomalies are identified by measuring deviations from these learned embeddings (Liu et al., 2024b). By focusing on feature embeddings, these models leverage powerful deep learning techniques to detect subtle discrepancies that signify defects.

Deep One-class classification methods use deep neural networks to extract high-quality features from normal images, ensuring these features are compactly distributed in the feature space. This compact distribution allows for the construction of precise boundaries to distinguish normal features from anomalies. Deep support vector data description (Deep SVDD) is an important method in one-class classification, and it trains a neural network to map normal sample features into the hypersphere to distinguish whether the test sample is abnormal or not (Ruff et al., 2018). Different researchers have continously based and optimized SVDD methods to enhance the effectiveness of image anomaly detection and localization, such as PatchSVDD (Yi and Yoon, 2020), Fully Convolutional Data Description (FCDD) (Liznerski et al., 2020), etc.

Student-teacher networks employ a dual-network system where a student network learns to replicate the feature representations of a pre-trained teacher network. The network structure is shown at Figure 2. The difference between the two networks helps in identifying anomalies. Bergmann et al. (2020) firstly apply this network into anomaly detection. The representational power of a large pre-trained network is transferred to a lightweight teacher network through knowledge distillation. Then, multiple randomly initialized student networks are trained on standard datasets to ensure they represent normal samples similarly to the teacher network. This method relies on regression errors in defect representation between the student and teacher networks, as well as high uncertainty in defect representation among multiple student networks, to achieve pixel-level defect segmentation. Wang et al. (2021) use a pre-trained image classification model as a teacher to distill knowledge into a single student network, which learns the distribution

of anomaly-free images while preserving key cues. By integrating a multi-scale feature matching strategy, the student network can detect anomalies of various scales, with the difference between the feature pyramids of the two networks serving as a scoring function for anomaly probability. While using similar architectures to build the student and teacher models hinders the diversity of anomalous representations, Deng and Li (2022) propose a novel teacher-student model and an effective reverse distillation paradigm where the student restores the teacher's multi-scale representations from its one-class embedding to tackle this problem.

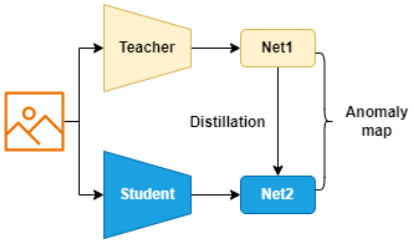


Figure 2: Student-teacher network

Distribution map models aim to model the probability distribution of the features of normal samples, thus eliminating the need to build a large library of normal samples. After capturing the underlying distribution of normal data, anomalies are detected by identifying data points that deviate from that distribution (Liu et al., 2024b). Rippel et al. (2021) first extract multi-scale features of normal samples using a pre-trained network and modeled each feature map as a multivariate Gaussian distribution separately, and they applied the Mahalanobis distance as the anomaly score. Normalizing Flows (NF)-based methods now are dominant (Liu et al., 2024b), where NF is a technique for constructing complex distributions by transforming probability densities through a series of invertible mappings (Rezende and Mohamed, 2015). DifferNet first applied NF-based models to increase the flexibility (Rudolph et al., 2021). CFlow-AD enhances the conditional NF framework by introducing positional encoding, thereby improving anomaly detection performance and thoroughly analyzing the rationale behind the multivariate Gaussain assumption in earlier models (Gudovskiy et al., 2022). CSFlow incorporates cross-convolutional blocks within the NF, leveraging contextual information from multi-scale feature mappings to increase the accuracy of anomaly detection (Rudolph et al., 2022). Meanwhile, FastFlow alternates between large and small convolutional kernels to effectively model both global and local distributions (Yu et al., 2021).

Memory bank approach maintains a repository of normal feature representations. The primary idea is to store features of normal data during training, which can then be used during inference to compare and detect anomalies. Spatially-Adaptive Denormalization (SPADE) uses a pre-trained CNN model to extract the feature vectors of the training set to construct a database of normal samples, and then uses the K-Nearest neighbors method to obtain anomaly segmentation results using a multi-resolution feature pyramid matching method (Cohen and Hoshen, 2020). PaDiM uses a pretrained CNN for patch embedding and employed multivariate Gaussian distributions to obtain a probabilistic representation of the normal class (Defard et al., 2021). It leverages correlations between different semantic levels of the CNN to improve anomaly localization. PatchCore uses a maximally representative memory bank of nominal patch-features, and it can detect minute defects that might be missed by other methods by storing and comparing patch-level features (Roth et al., 2022).

Reconstruction Based Methods. The core idea of reconstruction-based methods is to train a model using normal samples to learn the distribution characteristics of normal data. Anomalies are then detected based on the reconstruction error (Luo et al., 2022). It is assumed that normal data can be reconstructed accurately, while abnormal data will have a larger reconstruction error due to its deviation from the normal data distribution. Therefore, by analyzing the difference between the input image and the reconstructed image, anomalies can be effectively identified. This method includes autoencoder (AE), variational autoencoder (VAE), Generative Adversarial Network (GAN), transformer, diffusion, etc.

AE consists of two main parts: an encoder and a decoder. The encoder compresses the input image data into a lower-dimensional latent space, capturing the essential features of the input. The decoder then attempts to reconstruct the original data from this compact representation. Defect localization can then be achieved based on the reconstruction error between the input image and the reconstructed image. While in VAE, a variant of AE, it maps the input to a distribution, typically a Gaussian distribution, instead of mapping the input image data to a single point in the latent space (Kingma et al., 2019). The framework of AE and VAE is shown at Figure 3.

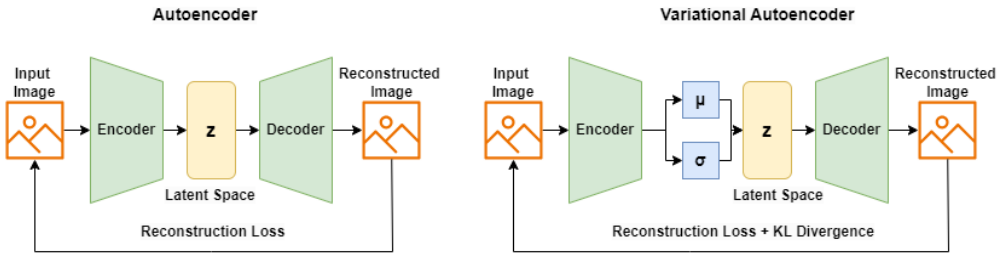


Figure 3: Framework of AE and VAE

In order to solve the blurring phenomenon of AE in reconstructed images, Discriminative Feature Refinement (DFR) improves the quality of reconstructed images by choosing to implement the multi-scale fusion of information in the hidden space (Yang et al., 2020), while another method simulates the blurring effect of AE by introducing a stylized distillation branch, which stylizes the input image and reduces the misdetection of the normal pixel points when calculating reconstruction errors (Chung et al., 2020). DRAEM combines both reconstructive and discriminative approaches by learning a joint representation of an anomalous image and its anomaly-free reconstruction, while simultaneously establishing a decision boundary between normal and anomalous examples without the need for additional post-processing (Zavrtanik et al., 2021). DSR, based on a quantized feature space representation with dual decoders, avoids the need for image-level anomaly synthesis by generating anomalies at the feature level through sampling the learned quantized feature space, allowing for controlled generation of near-in-distribution anomalies (Zavrtanik et al., 2022).

VAE can construct structured latent space manifolds that are more controllable than AE, and thus the main common features of normal samples can be learned from the perspective of latent space distribution (Luo et al., 2022). To cope with the difficulty of obtaining clear and consistent reconstructed images due to random sampling, (Dehaene et al., 2020) uses the idea of iterative approximation, while FAVAE models the feature distributions extracted by the pre-trained model to enhance the generalization of the model (Dehaene and Eline, 2020), and in another approach,

VQ-VAE is used as a reconstruction model to obtain the discrete latent space of the normal samples and to estimate the discrete latent space of the probabilistic model. In the detection phase, the autoregressive model will determine the portion of the input latent space that deviates from the normal distribution. The deviant code is then resampled and decoded from the normal distribution to obtain a restored image that is closest to the anomalous input (Wang et al., 2020).

GAN has a powerful ability to model distributions and generate high-quality images. A GAN consists of two components: a generator (G) that creates images and a discriminator (D) that evaluates their realism (Creswell et al., 2018). The adversarial training mechanism between the generator and discriminator is key to producing clear images, as the generator improves by trying to fool the discriminator, which in turn becomes better at distinguishing real images from generated ones. There are different models that have used GAN for image anomaly detection, such as AnoGAN which uses the idea of iterative optimisation, although the model inference time is long and the practicality is poor (Schlegl et al., 2017). While f-AnoGAN adds additional encoders to extract image features and uses a multi-stage training approach to guide the generator to produce the best matching image (Schlegl et al., 2019). OCR-GAN proposes a frequency decoupling module to separate the input image into different frequency components, modeling reconstruction as parallel omni-frequency restorations. Additionally, it introduces a channel selection module that enhances frequency interaction among different encoders by adaptively selecting channels (Liang et al., 2023).

Transformers leverage self-attention mechanisms to capture long-range dependencies in data, making them highly effective for anomaly detection (Han et al., 2022). By modeling complex relationships within the data, transformers can accurately reconstruct normal patterns and identify anomalies based on deviations from these patterns. (Mishra et al., 2021) present a transformer-based image anomaly detection and localization network that combines reconstruction and patch embedding, using a Gaussian mixture density network to localize anomalies. A masked Swin Transformer Unet (MSTUnet) is proposed for anomaly detection, using the Swin Transformer's global learning ability to inpaint masked areas created by an anomaly simulation and mask strategy, followed by a convolution-based Unet for end-to-end detection (Jiang et al., 2022).

Diffusion models are deep generative models based on two stages: a forward diffusion stage and a reverse diffusion stage. In the forward diffusion stage, the input data is gradually perturbed over several steps by adding Gaussian noise, while in the reverse stage, a model learns to recover the original input data by gradually reversing the diffusion process (Croitoru et al., 2023). For anomaly detection, these models leverage their ability to produce high-quality and diverse normal data patterns, identifying anomalies based on deviations from these patterns during the reverse diffusion process, despite their computational burdens due to the high number of steps involved. There are three generic diffusion modeling frameworks which are denoising diffusion probabilistic models, noise conditioned score networks, and stochastic differential equations (Croitoru et al., 2023). Denoising Diffusion Probability Models (DDPM) perform well on anomaly detection benchmarks, but are computationally expensive (Sasaki et al., 2021). By simplifying DDPM for anomaly detection, (Livernoche et al., 2023) propose Diffusion Time Estimation (DTE), which estimates the distribution over diffusion time for a given input and uses the mode or mean as the anomaly score. Also in medical imaging anomaly detection, diffusion model has a wide range of applications. Iqbal et al. (2023) use masked-DDPM which introduces masking-based regularization, specifically Masked Image Modeling (MIM) and Masked Frequency Modeling (MFM), to enhance the generation task of diffusion models for brain medical applications.

Many of the current state-of-the-art methods are also use a blend of methods, rather than

being limited to using only one approach. Masked Multi-scale Reconstruction (MMR) integrates both feature embedding and reconstruction-based methods (Zhang et al., 2023). Functioning as a student-teacher network, the frozen pre-trained encoder serves as the teacher while the student network learns from it. By employing a masked AE strategy, MMR enhances the model's ability to understand spatial dependencies and causality in normal samples, preventing information leakage from visible to masked parts of the image.

2.3. Data Augmentation and Synthesis

Since there is often a lack of defective samples with precision labeling in industry today, which is not enough to support the training of neural networks, data augmentation and synthesis are often required to improve model performance. Rippel et al. (2020) address lack of large amounts of annotated training data by leveraging the consistency of defect appearance across fabrics to transfer knowledge about anomalies from one fabric to another. While Defect-GAN approach can automated generate realistic and diverse defect samples for training inspection network (Zhang et al., 2021). It uses a compositional layer-based architecture to generate and restore defects on normal surface images, offering realistic defect generation with flexible control over their location, category, and appearance. Liu et al. (2024a) propose SyNet, a novel unsupervised learning method based on noisy anomaly synthesis for medical image anomaly detection and Tayeh et al. (2020) use random erasing techniques to synthesize defective training samples by introducing artificial defects into non-defective samples.

Cutpaste is commonly used for data augmentation by cutting an image patch and pasting it at a random location on a larger image (Li et al., 2021). Building on this, Natural Synthetic Anomalies (NSA) integrates Poisson image editing to create more naturally appearing sub-image irregularities (Schlüter et al., 2021). Similarly, AnoSeg enhances the diversity of synthetic defects by applying data augmentations like random rotation, positional disruption, and color dithering before cropping, and it incorporates coordinate channels representing pixel positional information into AnoSeg's inputs to account for the positional relationships within the image (Song et al., 2021).

3. Methodology

3.1. Use Case Background

This research was conducted in collaboration with a prominent Nordic automobile manufacturer known for its extensive global production footprint, with facilities spread across various international locations. The study specifically focused on one of the manufacturer's key production plants in Sweden. At this plant, the maintenance department faced a critical challenge: verifying glue lines against stringent industry standards. Ensuring the quality of glue lines is important, as any deviation from the established standards could significantly impact the vehicle assembly line and compromise the overall manufacturing process.

Historically, the quality control process for glue lines at this prominent Swedish automobile manufacturer relied on an automated system utilizing industrial-grade cameras. Specifically, the SICK Pim60 Vision robot (illustrated in Figure 4), mounted above the production line at a fixed height and angle to capture optimal views of the glue lines (SICK AG, 2024), captured images at set intervals (details on the cycle interval are omitted for confidentiality). Ten images were captured for each control point, and the system analyzed them using embedded optical techniques. This analysis produced a binary output (”OK” or ”NOT OK”) for each image, indicating whether

the glue line met quality standards. Despite this automation, the approach still required manual review to ensure the accuracy and reliability of defect detection.

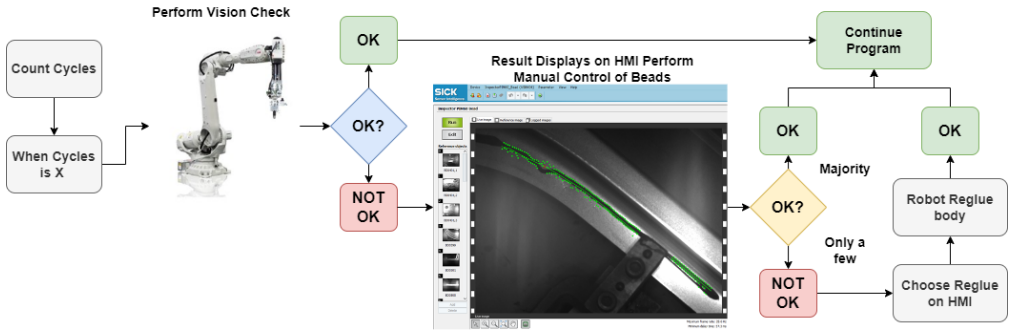


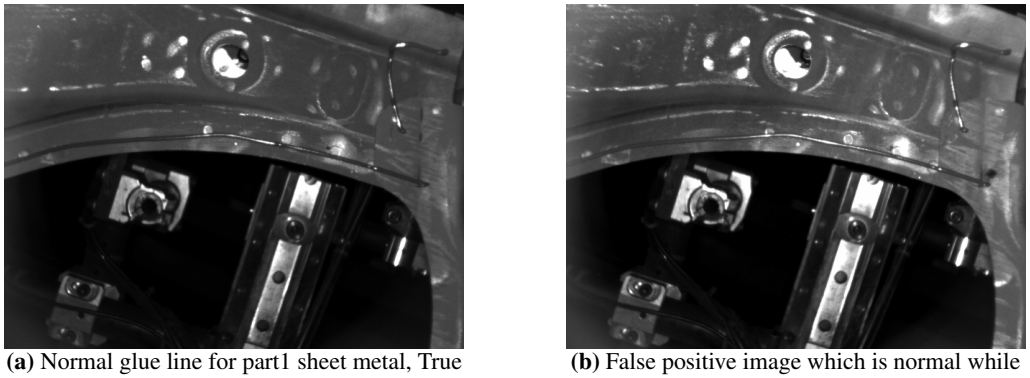
Figure 4: Industry Process Description: Cycle Based Vision Controls

During the company's operations, a high rate of false detections has been noted when inspecting glue lines. Data shows that about majority of glue lines meet the required standards, yet they are still flagged as defective by the imaging system. This misclassification is mostly due to external factors, especially inconsistent lighting conditions during image capture. These false detections have several consequences. When the automated system identifies a supposed defect, it can trigger a shutdown, requiring manual checks. Maintenance staff must then verify the glue lines and restart the equipment, which reduces productivity and increases the workload. Given that true instances of non-compliance are rare, this highlights the gap between what the system detects and the actual quality of the glue lines.

3.2. Dataset

To conduct our research and assist the company in addressing the aforementioned issues, we collaborated with the maintenance department to collect glue lines' image data and form a dataset. This dataset consists of multiple sets of grayscale images of different sections of glue lines, systematically captured by stationary industrial cameras. The dataset includes 20 distinct components, each containing approximately 300 images, with a resolution of 480 pixels high by 640 pixels wide. These images, consisting of a single channel, contain no color data and emphasize contrast variations important for the analysis, and show segments of glue lines across various machinery, providing a representative cross-section of the production line's conditions.

To provide visual context for the following analyses, Figure 5 shows some sample images from this collection. In the image, the glue line appears in the middle, adjacent to various sheet metal structures of the car. The glue line is not particularly prominent in the image. Displayed sequentially from left to right are representative images from one part: a standard glue line indicative of proper application and a glue line meeting the required standards but erroneously classified as defective. The glue lines with errors not detected by the system constitute a very small sample and are not shown here. Additionally, due to confidentiality reasons, the real fault images and the percentage of faulty images are not disclosed. The majority of this paper will focus on the most representative section (Part 1), which presents a moderate level of difficulty but reflects the majority of cases.



(a) Normal glue line for part1 sheet metal, True positive image

(b) False positive image which is normal while detected as anomaly by vision robot

Figure 5: Example glue line images from part1 sheet metal.

Due to the imbalanced dataset exhibiting a high false-positive rate and a low false-negative rate, we employed a data augmentation strategy to improve the training set. Along with consulting industry professionals, we used relevant image editing software, GIMP, to modify specific regions within acceptable glue line images. These edited regions were altered to closely resemble true industrial defects, effectively generating synthetic false images to supplement the training set.

We selected a representative industrial image and divided the depicted glue line into five distinct segments, labeled A through E (see Figure 6). The primary glue line is segmented into four distinct areas, each with unique characteristics affecting their inspection:

- **Area A** is situated in the middle of the left side of the image and features a darker background underneath, which can influence detection accuracy.
- **Area B**, located centrally within the image, is very close to the parts below it, creating potential challenges in distinguishing the glue line from adjacent components.
- **Area C** is near a small hole above it and influenced by a similarly long sheet metal below it, complicating accurate detection.
- **Area D**, near the right side of the image, is also affected by the sheet metal below it and the proximity to the edge of the image, complicating segmentation.
- **Area E**, in the upper right corner, has a curvature where the glue line detaches from the main glue line, making it the most error-prone during inspection.

By categorizing the glue line into specific areas, we tailored the inspection process to address the unique challenges presented by each region, thereby improving the overall accuracy and reliability of defect detection. To augment our dataset with a comprehensive range of false cases, we employed GIMP's Warp transform tool (GIMPDoc, 2020) to simulate realistic glue application errors within each image segment (A through E). These simulations replicate potential defects caused by instability or jitter during the robotic application process, ranging from subtle localized distortions to extensive application anomalies.

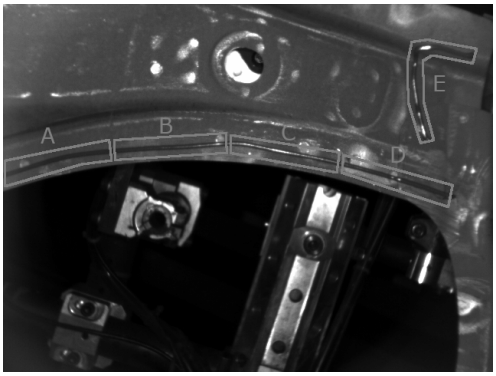
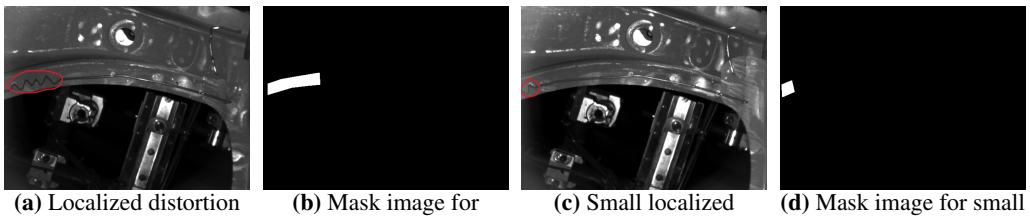


Figure 6: Part1 glue line segmentation from Area A to E.

In Figure 7, we present examples of both normal and anomalous glue line segments, with the anomalies explicitly marked by red circles for improved clarity. The first example illustrates a localized distortion within Segment A, where the anomaly is clearly annotated. The corresponding binary mask image further delineates the anomalous region, enabling precise pixel-wise comparison. Similarly, the second example demonstrates a small localized distortion within Segment A, marked with a red circle, and its corresponding binary mask image. These annotated and masked images enhance the interpretability of the data, ensuring that both large and small anomalies are effectively represented.



(a) Localized distortion at segment A

(b) Mask image for segment A distortion

(c) Small localized distortion at segment A

(d) Mask image for small localized distortion

Figure 7: Examples of normal and anomalous glue line segments, with corresponding mask images. Anomalies are marked by red circles and further segmented in binary mask images for precise evaluation of defect localization.

3.3. Selection of Anomaly Detection Models

Given the unknown and irregular nature of our industrial image defects, the high cost of manual annotation, the stringent requirements for detection accuracy and speed, and the imbalance in our sampled dataset, we adopt unsupervised deep learning approaches to determine whether samples contain defects and to localize them. While Zipfel et al. (2023) compared three unsupervised deep learning anomaly detection models (PatchCore, Skip-GANomaly, and PaDiM) for vehicle identification numbers (VIN labels), our work extends these efforts by including a broader range of models and backbones, as listed in Table 2. This approach enables us to encompass a wider spectrum of mainstream detection methods in unsupervised deep learning, including both feature embedding–based and reconstruction-based techniques. Because diffusion-based reconstruction

approaches are predominantly applied to medical images rather than manufacturing scenarios, and because transformer-based models generally require substantial training data which is often unavailable in industrial defect detection (Luo et al., 2022), we exclude these models. To keep our experiments and evaluations consistent, we primarily use models from the Anomalib library (Akcay et al., 2022), supplemented by additional representative models not included in Anomalib (e.g., SimpleNet, MMR). Table 2 presents each model's trainable parameters and the original datasets used in their respective publications. Their accuracy on the public image anomaly dataset MVTec AD is provided in Appendix A.7. The overall framework of our methods is shown in Figure 8, where we compare and evaluate these state-of-the-art unsupervised models on our sheet metal glue line image dataset.

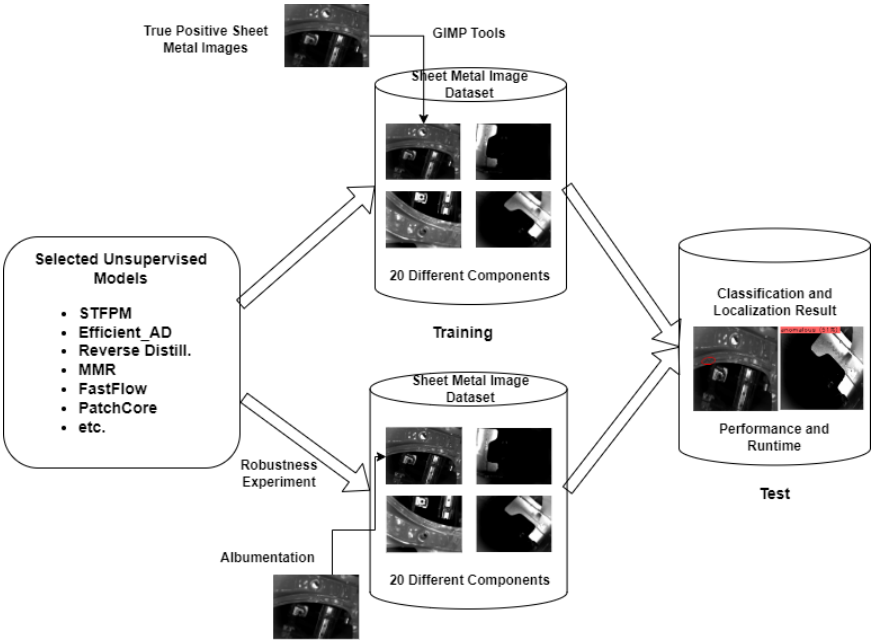


Figure 8: Framework of our methods. Including training and robustness experiment.

Table 2: Comparison of Anomaly Detection Models

	Sub-category	Model Name	Backbone	Model Parameters	Datasets
Feature Embedding based	Student-Teacher Networks	Efficient_AD. (Batzner et al., 2024)	EfficientAd	8.1M Trainable	MVTec AD, VisA (Kagawade and Angadi, 2021)
		stfpm. (Wang et al., 2021)	ResNet18	2.8M Trainable	MVTec AD
		Reverse_Distillation. (Deng and Li, 2022)	ResNet18	18.7M Trainable	MVTec AD, MNIST, Cifar10, F-MNIST (Xiao et al., 2017)
		MMR. (Zhang et al., 2023)	WideResNet50		MVTec AD, AeBAD (Zhang et al., 2023)
	Distribution Map	CFlow. (Gudovskiy et al., 2022)	WideResNet50	81.6M Trainable; 154M Non-trainable	MVTec AD, STC (Liu et al., 2018)
		CSFlow. (Rudolph et al., 2022)	EfficientNet-B5	275M Trainable; 17.5M Non-trainable	MVTec AD, MTC (Huang et al., 2020)
		FastFlow. (Yu et al., 2021)	ResNet18 WideResNet50 Cait Deit	5.6M Trainable; 4.2M Trainable 78.0M Trainable; 46.9M Non-Trainale 31.9M Trainable; 365M Non-Trainable 7.1M Trainable; 111M Non-Trainable	MVTec AD, Cifar10, BTAD (Ma et al., 2023)
		DFM. (Ahuja et al., 2019)	ResNet50	2.8M Trainable	MNIST (Xiao et al., 2017), Cifar10(Alex, 2009)
	Memory Bank	PatchCore. (Roth et al., 2022)	WideResNet50	24.9M Trainable	MVTec AD, STC, MTC
		PaDiM. (Defard et al., 2021)	ResNet18 WideResNet50	2.8M Trainable 24.9M Trainable	MVTec AD, STC

Continued on next page

Table 2 – Continued

	Sub-category	Model Name	Backbone	Model Parameters	Datasets
Reconstruction Based		CFA. (Lee et al., 2022)	ResNet18 WideResNet50	3.2 M Trainable 31.3 M Trainable	MVTec AD (Bergmann et al., 2021)
		SimpleNet. (Liu et al., 2023)	WideResNet50		MVTec AD, Cifar10
	OCC	UFlow. (Tailanian et al., 2022)	mCaiT ResNet18 WideResNet50	12.2M Trainable; 409M Non-Trainable 4.3M Trainable; 3.6M Non-Trainable 34.8M Trainable; 37.4M Non-Trainable	MVTecAD, STC, BT (Mishra et al., 2021), MRI (Buda et al., 2019)
	AutoEncoder	DRAEM. (Zavrtanik et al., 2021)		97.4M Trainable	DTD (Cimpoi et al., 2014)
		DSR. (Zavrtanik et al., 2022)		36.3M Trainable; 4.0M Non-Trainable	MVTec AD, KSDD2 (Božič et al., 2021)
	GAN	GANomaly. (Akcay et al., 2019)*	GAN	188M Trainable	MNIST, Cifar10, UBA (Rogers et al., 2017), FFOB (UK Home Office Centre for Applied Science and Technology (CAST), 2016)

** Indicates image level, otherwise are pixel level*

3.4. Experimental Setup

We executed a series of experiments applying selected models on our glue line dataset. For baseline comparisons, we employed deep feature kernel density estimation. The hyperparameters for selected models for comparison are shown in Appendix B.8. Initially, we adopted the default hyperparameters as provided by their respective implementations, as these are generally optimized for a broad range of scenarios. To ensure suitability for our specific dataset and objectives, we conducted preliminary evaluations to verify their effectiveness. This approach allowed us to maintain consistency and reliability across comparisons while focusing on the broader objectives of the study. To ensure experimental fairness and consistency, we integrated all implementations within a shared environment. Our computational environment consisted of Python (version 3.10.13) as the programming language, PyTorch (version 1.13.1) as the deep learning framework, a Linux system with CPU: i9-13900K, RAM: 128GB, GPU: RTX4090 (24GB VRAM), and CUDA (version 11.6) for GPU acceleration.

3.5. Robustness Experiment

We conducted an additional experiment utilizing a data augmentation strategy during model training to assess and compare the model's robustness in our plant environment. This approach aims to improve model generalization by exposing it to a wider range of variations, thereby promoting the learning of robust feature representations applicable to unseen data. By simulating diverse scenarios, data augmentation also enhances the model's ability to handle potential real-world image imperfections and helped prevent over-fitting by increasing the diversity of the training dataset. Additionally, our experiments sought to address common issues such as camera shake and other potential artifacts caused by the motion performance of the vision robot. This experiment provided valuable insights into the model's performance and robustness under augmented conditions.

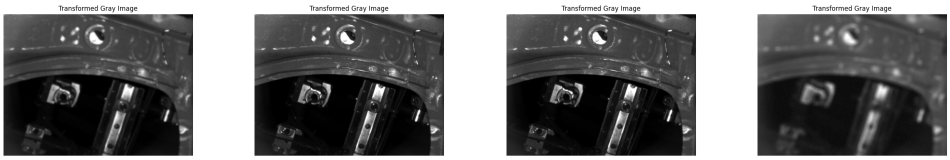
Informed by expert interviews and common photographic challenges in industrial robotics, we selected the following data augmentation modes: Defocus, simulating potential blurring due to focus errors; Random Brightness Contrast, addressing variations in lighting conditions; and ISO Noise, emulating image noise artifacts that may arise from camera sensor limitations (AlbumentationsAI, 2024). Table 3 details the specific parameter settings for each technique. Examples of applying this data augmentation are shown in Figure 9, which illustrate the application of each data augmentation individually and the use of all data augmentations at the same time.

Table 3: Data Augmentation Parameter Settings

Mode	**Parameters**
Defocus	p = 0.5, radius = [3, 10], alias_blur = [0.1, 0.5]
RandomBrightnessContrast	p = 0.5, brightness_limit = [-0.2, 0.2], contrast_limit = [-0.2, 0.2], brightness_by_max = True
ISO Noise	p = 0.5, color_shift = [0.01, 0.05], intensity = [0.1, 0.5]

3.6. Evaluation Metrics

In this research, we evaluated the performance of our deep learning models for glue line anomaly detection using a combination of image-level and pixel-level metrics.



(a) Defocus **(b)** Random Brightness **(c)** ISO Noise **(d)** All Applied

Figure 9: Data Augmentations Application Example.

3.6.1. Image-Level Metrics

Image-Level AUROC (Area Under the Receiver Operating Characteristic Curve): The AUROC metric provides an aggregate measure of a model's ability to discriminate between normal and anomalous glue line images. It is calculated across all possible classification thresholds, representing the trade-off between the true positive rate (TPR) and false positive rate (FPR). These are defined as:

TPR (also known as recall): The proportion of actual anomalies (positive samples) that are correctly identified by the model:

$$TPR = \frac{TP}{TP + FN}$$

where TP represents true positives and FN represents false negatives.

FPR: The proportion of normal images (negative samples) that are incorrectly identified as anomalous:

$$FPR = \frac{FP}{FP + TN}$$

where FP represents false positives and TN represents true negatives.

The AUROC summarizes the model's performance over various decision thresholds by plotting TPR against FPR and measuring the area under this curve:

$$AUROC = \int_0^1 TPR(FPR^{-1}(x))dx$$

A model with perfect discrimination will have an AUROC of 1, while a random classifier will have an AUROC of 0.5.

Image-Level F1 Score: The F1 score balances precision (the proportion of true positives out of predicted positives) and recall (the proportion of true positives correctly identified). It is particularly useful when the dataset exhibits class imbalance, as is often the case with anomaly detection. Precision is the proportion of images predicted as anomalous that are actually anomalous while recall is shown as TPR.

$$Precision = \frac{TP}{TP + FP}$$

Then F1 score is calculated as:

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

3.6.2. Pixel-Level Metrics

Pixel-Level AUROC: Similar to the image-level AUROC, this metric assesses a model's discrimination ability at the individual pixel level. It is particularly relevant for tasks involving localization, such as pinpointing the exact areas within a glue line that exhibit anomalies.

Pixel-Level F1 Score: The pixel-level F1 score provides a granular evaluation of how well the model correctly identifies anomalous pixels. This metric is valuable when precise localization of defects is crucial.

3.6.3. Image-Level vs. Pixel-Level Evaluation

Image-level metrics provide a global assessment of whether an image contains an anomaly or not (classification focus). Pixel-level metrics offer a more detailed analysis, pinpointing the specific regions within an image that are considered anomalous (segmentation focus). By using both image-level and pixel-level metrics, we gain a comprehensive understanding of our models' ability to both detect and localize glue line defects.

4. Results

This section presents the results of our experiments. To ensure the reliability of our findings, we conducted experiments using five different random seeds for each model configuration.

4.1. Performance

4.1.1. Run Time Performance

The average training time, inference time, and throughput per second for part1 are presented in Table 4. DSR has the longest training time at approximately 2.5 hours, whereas dfkde requires only 2.4 seconds, making it the fastest among the models. It is evident that feature-embedding based methods generally require less training time compared to reconstruction-based models. Efficient AD exhibits a higher training time due to its use of the AE method. Among feature-embedding based models, those from the memory bank category are relatively quick, with training times of about 1 minute. Models utilizing the ResNet18 backbone train faster than those with WideResNet50. This is primarily because ResNet18 has fewer layers and parameters, resulting in lower computational complexity and faster processing. Additionally, ResNet18's smaller memory footprint allows for more efficient batch processing. In contrast, WideResNet50, with its increased depth and width, demands more computation and memory, leading to longer training times. However, an exception is observed with Uflow, where the training times for ResNet18 and WideResNet50 backbones are quite similar. When the backbone is changed to mcait, the training time increases by about 14 times compared to ResNet18.

Inference time, a critical factor for productivity and efficiency, is analyzed in Figure 10. Lower inference times are preferred, while higher throughput—measuring the number of images processed per second—is advantageous. The models' inference times per image range from 91.29 ms to 302.41 ms, with less variation compared to training times. Throughput depends on batch size, which is typically set to 32 for most models. While inference speeds between feature-embedding and reconstruction-based methods are similar, variations occur due to batch size and backbone architecture. Notably, WideResNet50 and ResNet18 backbones exhibit comparable inference times within the same models.

The inference time and throughput of the models evaluated in this work are well-suited for real-time detection requirements on industrial production lines. For example, even the slowest

model achieves a throughput exceeding 3 images per second, which meets the general demands of industrial scenarios. These performance metrics ensure efficient and timely detection capabilities, making the models practical for real-world deployment in production lines.

Table 4: Runtime Comparison of Models

Category	Model	BackBone	Train(min)	Inference(ms)	Throughput(fps)
	dfkde	ResNet18	0.04	91.29	10.95
		WideResNet50	0.06	93.22	10.73
Student-Teacher	Efficient AD		50.99	173.07	5.78
	STFPM	ResNet18	0.97	193.22	5.18
		WideResNet50	3.13	195.41	5.12
	Reverse Distill.	WideResNet50	7.57	199.25	5.02
		ResNet18	2.53	194.05	5.15
	MMR	WideResNet50	6.83	253.57	3.94
Distribution Map	CFlow-AD	WideResNet50	13.85	281.38	3.55
	CSFlow	EfficientNet-B5	6.78	302.41	3.31
	FastFlow	ResNet18	0.51	194.80	5.13
		WideResNet50	1.37	194.37	5.14
		cait	4.36	281.95	3.55
		deit	0.86	210.99	4.74
	DFM	ResNet50	0.06	192.12	5.21
		ResNet18	0.05	186.21	5.37
Memory Bank	PatchCore	WideResNet50	1.57	198.18	5.05
	PaDiM	ResNet18	0.13	214.10	4.67
		WideResNet50	0.39	192.60	5.19
	CFA	ResNet18	0.30	256.61	3.90
		WideResNet50	0.90	261.64	3.82
	SimpleNet	WideResNet50	0.85	225.00	4.44
OCC	Uflow	mcait	37.77	292.11	3.42
		ResNet18	2.62	212.06	4.72
		WideResNet50	2.61	213.84	4.68
AE	DRAEM		16.93	216.40	4.62
	DSR		151.11	222.26	4.50
GAN	GANomaly		1.08	96.17	10.40

4.1.2. Predictive Performance

For model performance, we report the mean and standard deviation across these runs, providing a comprehensive assessment of model performance. Evaluation focuses on both image-level (Image AUROC, Image F1 Score) and pixel-level (Pixel AUROC, Pixel F1 Score) metrics to capture both the detection and localization capabilities of the models. The overall performance

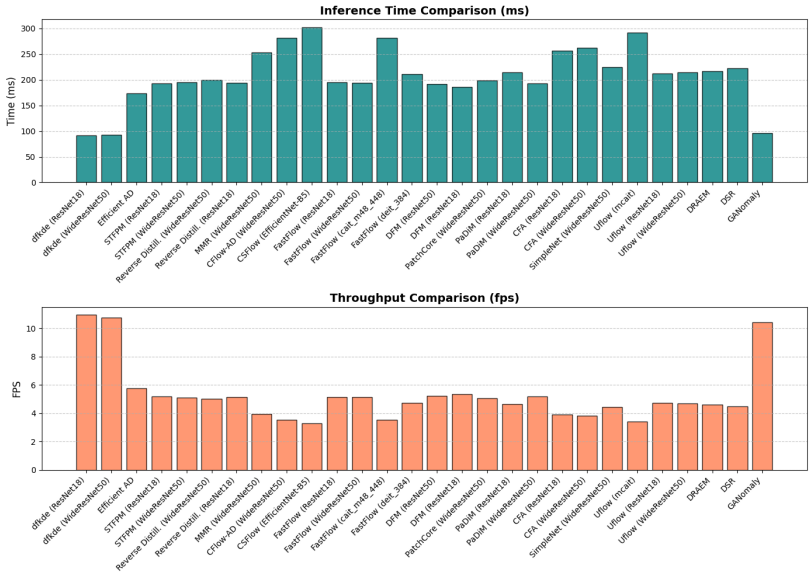


Figure 10: Inference time and throughput comparison

for each model is shown at Table 5. In the table, the best performance for different performance is shown in bold. The false positive reduction rate can be presented from the image level AUROC.

Regarding image level AUROC, the best performer is STFPM, which achieved an AUROC of 0.985. While the false positive rate of Part 1 from the company remains confidential, the accuracy of the optimal model significantly surpasses that of the company's current solution. Furthermore, STFPM successfully resolved all false positive images in Part 1. This indicates that STFPM excels in distinguishing between normal and anomalous images, showcasing exceptional reliability and accuracy. Among the top four models, namely STFPM, SimpleNet, FastFlow and Efficient_AD, SimpleNet emerges as the most stable model due to its low variance, which indicates consistent performance across different runs. The models with the worst performance are the DFM and baseline model DFKDE. An AUROC lower than 0.5 indicates poor performance, essentially worse than random guessing. Therefore, DFKDE and DFM are considered highly ineffective for anomaly detection tasks due to their extremely low AUROC values, which highlight their inability to distinguish between normal and anomalous images effectively. The segmentation image that shown in Figure 11 for DFM illustrates DFM has a rougher split area compared to stfpm.

The performance comparison of different backbones with the same model reveals insightful patterns. For instance, the CFA model shows a significant improvement when using the WideResNet50 backbone compared to ResNet18. Similarly, PaDiM with ResNet18 achieves an AUROC of 0.901, while with WideResNet50, it improves to 0.962, indicating a clear enhancement with the more complex backbone. FastFlow exhibits variability depending on the backbone used, with WideResNet50 showing the best performance, followed by deit and cait. These comparisons suggest that models generally perform better with the WideResNet50 backbone compared to ResNet18, while the training time is longer, indicating that a more complex backbone architecture tends to enhance the model's anomaly detection capabilities. However,

Table 5: Prediction performance metrics of Part1 for various unsupervised DL models and corresponding backbones.

Model	Backbone	Image AUROC	Image F1 Score	Pixel AUROC	Pixel F1 Score
CFA	ResNet18	0.810 ± 0.015	0.950 ± 0.007	0.984 ± 0.001	0.453 ± 0.020
	WideResNet50	0.889 ± 0.026	0.960 ± 0.008	0.993 ± 0.000	0.544 ± 0.002
CFlow-AD	WideResNet50	0.871 ± 0.042	0.950 ± 0.007	0.993 ± 0.000	0.539 ± 0.008
CSFlow	EfficientNet-B5	0.822 ± 0.015	0.943 ± 0.000	0.955 ± 0.002	0.224 ± 0.006
DFKDE	ResNet18	0.130 ± 0.000	0.943 ± 0.000		
	WideResNet50	0.120 ± 0.000	0.943 ± 0.000		
DFM	ResNet50	0.320 ± 0.000	0.943 ± 0.000	0.986 ± 0.000	0.342 ± 0.000
	ResNet18	0.280 ± 0.000	0.943 ± 0.000	0.983 ± 0.000	0.316 ± 0.000
DRAEM		0.934 ± 0.037	0.970 ± 0.016	0.982 ± 0.006	0.635 ± 0.079
DSR		0.968 ± 0.026	0.966 ± 0.016	0.971 ± 0.016	0.683 ± 0.051
Efficient_AD		0.981 ± 0.007	0.978 ± 0.004	0.954 ± 0.000	0.674 ± 0.004
FastFlow	ResNet18	0.935 ± 0.022	0.967 ± 0.008	0.994 ± 0.001	0.560 ± 0.028
	WideResNet50	0.982 ± 0.010	0.982 ± 0.004	0.997 ± 0.000	0.638 ± 0.009
	cait	0.837 ± 0.040	0.969 ± 0.008	0.996 ± 0.000	0.662 ± 0.014
	deit	0.948 ± 0.024	0.977 ± 0.011	0.992 ± 0.002	0.597 ± 0.033
GANomaly		0.737 ± 0.251	0.948 ± 0.007		
MMR	WideResNet50	0.980 ± 0.004	0.997 ± 0.002	0.997 ± 0.000	0.597 ± 0.012
PaDiM	ResNet18	0.901 ± 0.040	0.969 ± 0.004	0.997 ± 0.000	0.660 ± 0.013
	WideResNet50	0.962 ± 0.012	0.973 ± 0.008	0.997 ± 0.000	0.614 ± 0.012
PatchCore	WideResNet50	0.944 ± 0.006	0.975 ± 0.005	0.996 ± 0.000	0.571 ± 0.002
Reverse_Distill.	WideResNet50	0.943 ± 0.036	0.969 ± 0.008	**0.998 ± 0.000**	0.662 ± 0.016
	ResNet18	0.871 ± 0.026	0.965 ± 0.005	0.997 ± 0.000	0.656 ± 0.006
SimpleNet	WideResNet50	0.983 ± 0.000	0.969 ± 0.011	0.967 ± 0.000	0.538 ± 0.006
stfpm	ResNet18	**0.985 ± 0.009**	0.982 ± 0.011	0.997 ± 0.000	0.673 ± 0.008
	WideResNet50	0.981 ± 0.004	0.976 ± 0.005	**0.998 ± 0.000**	**0.696 ± 0.003**
UFlow	mcait	0.951 ± 0.010	**0.986 ± 0.006**	0.996 ± 0.000	0.578 ± 0.015
	ResNet18	0.935 ± 0.019	0.967 ± 0.005	0.993 ± 0.000	0.538 ± 0.030
	WideResNet50	0.941 ± 0.033	0.967 ± 0.011	0.993 ± 0.001	0.475 ± 0.039

*No pixel level result for DFKDE and GANomaly because they only do classification.

the choice of backbone can significantly impact performance, and should therefore be selected based on the specific requirements of the application.

However, for image level F1score, all the models perform well and all the values are higher than 0.9. Even for those models that don't have good performance in image AUROC, DFKDE and DFM, the image F1 Scores are 0.943. This exceptional performance suggests that the models achieve a strong balance between precision and recall, meaning they are adept at correctly identifying anomalies while minimizing false positives and false negatives.

Only models specifically designed for segmentation exhibit pixel-level values. It is noteworthy that all the models achieve high pixel-level AUROC scores, with the lowest values observed in CSFlow and Efficient-AD, which are 0.955 and 0.954, respectively. These high pixel-level AUROC indicate that the models are proficient at localizing anomalies on a granular level, which is essential for applications requiring precise detection of defects within an image. The consistently high scores across different models suggest that the algorithms are well-tuned for detailed anomaly detection tasks. Notably, models such as MMR, stfpm, Reverse_Distillation, and PatchCore exhibit near-perfect pixel-level AUROC values, highlighting their reliability in accurately identifying anomalies at a fine-grained level. This performance trend underscores the effectiveness of these models in practical scenarios where pinpoint accuracy is crucial.

For the pixel-level F1 score, we observe varying performance across the different models. The best-performing models are stfpm and DSR, with pixel-level F1 scores of 0.696 and 0.683. While these scores are higher than 0.5, indicating some capability in identifying and segmenting anomalies at a fine-grained level, they are not exceptionally high. This suggests that even the best models have room for improvement in achieving precise pixel-level anomaly detection. Overall, the pixel-level F1 scores are generally lower than the image-level F1 scores across all models. This disparity highlights the increased challenge of precise anomaly localization at the pixel level compared to broader image-level anomaly detection. While models exhibit high performance in identifying anomalies at the image level, achieving the same precision and recall at the pixel level is more demanding, as evidenced by the lower F1 scores. This trend imply the complexity and higher granularity required for effective pixel-level anomaly segmentation.

4.1.3. Segmentation Performance

The representative example segmentation results for part1 anomalies are shown at Figure 11. In this figure, we present eight images that illustrate the results of image anomaly segmentation. The selection and sequence of these images are based on the Image AUROC performance metrics. Specifically, the first four images represent the models with the top four performances, while the last four images represent the models with the lowest performances. To ensure consistency and ease of comparison, all models shown are from Anomalib, as they follow the same format. The regions depicted in the images are A, C, D, and E. Region B, although analyzed, is not included because its characteristics are quite similar to Region A. The first four images correspond to the top-performing models in regions A, C, D, and E, respectively, while the last four images correspond to the worst-performing models, also in regions A, C, D, and E, respectively, and include their segmentation results.

As shwon in Figure 11, there is a clear relationship between model performance and segmentation precision. A more accurate model yields finer segmentation, which more precisely isolates the regions of glue line anomalies. For instance, in Region A, the area segmented by STFPM is significantly smaller and more accurately represents the actual deformation compared to DFM. While the pixel level performance of these two models is quite similar.

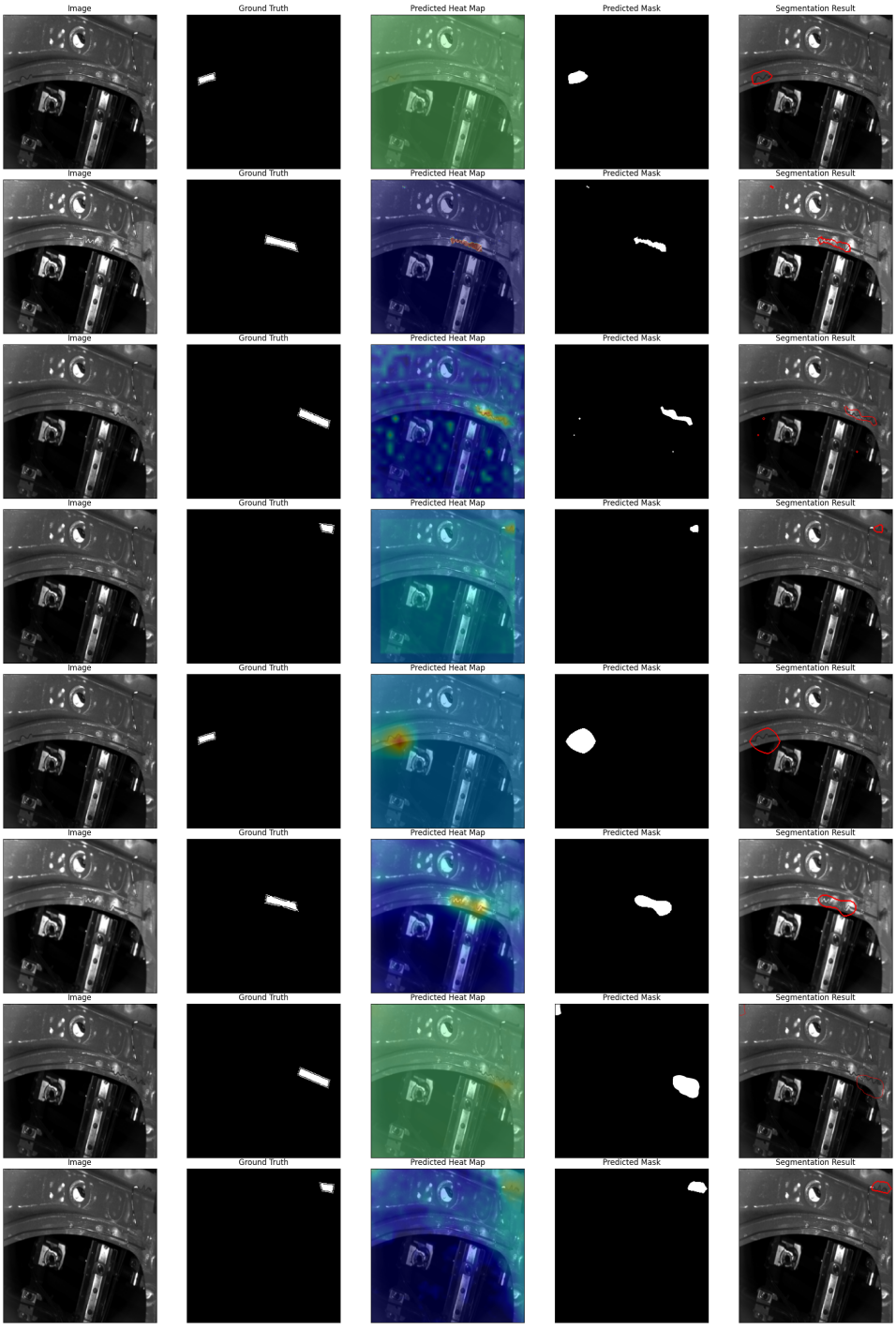


Figure 11: Segmentation Result, from the top to bottom: stfpm, DSR, fastflow, Efficient_Ad, DFM, CFA, CSFlow, CFlow-AD

4.2. *Robustness*

Robustness is essential for anomaly detection models, particularly when deployed in environments with varying data quality. In this study, robustness was evaluated by analyzing the impact of data augmentation on key metrics such as Image AUROC. The complete set of results are presented in Appendix C.9 and these are summarized in Figure 12.

Models like DFKDE show strong robustness, with both ResNet18 and WideResNet50 architectures improving in performance after augmentation. This indicates that DFKDE is well-suited to handle noisy and distorted data. On the other hand, models like PaDiM experienced noticeable decreases in Image AUROC, especially with the WideResNet50 architecture, suggesting a higher sensitivity to data variability.

Other models, such as DFM, showed slight improvements, while PatchCore, which had previously demonstrated strong robustness, exhibited a more noticeable decline in performance under the new augmentations. These findings underscore the varying levels of robustness across different models and architectures, highlighting the need to carefully select models that can maintain performance under diverse data conditions.

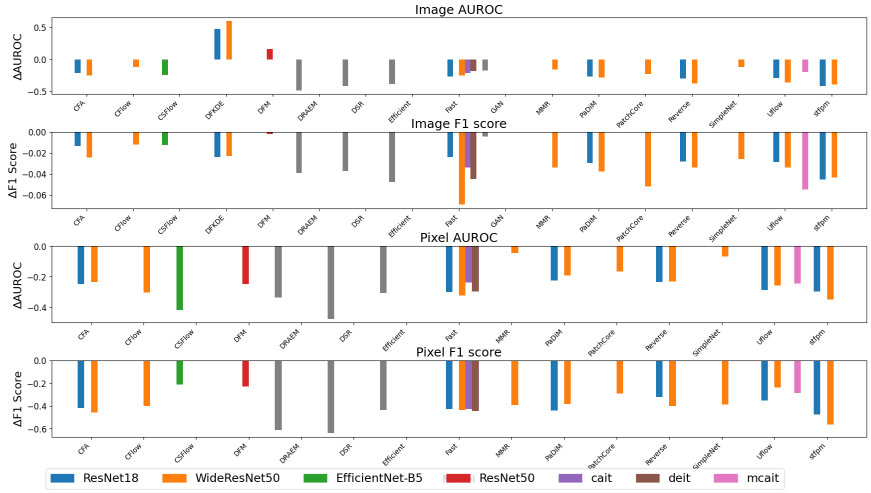


Figure 12: Robustness Performance Results

The image level F1 score differences highlight that most models retain a decent balance between precision and recall even after data augmentation. While some models like FastFlow show a more noticeable drop, the overall impact remains relatively modest, suggesting that these models are fairly robust in maintaining detection accuracy under varying data conditions.

The Pixel Level AUROC plot demonstrates how various anomaly detection models respond to pixel-level classification challenges after data augmentation. Generally, most models showed a decline in Pixel AUROC, indicating that data augmentation complicates the task of distinguishing between normal and anomalous pixels. Models like DRAEM and DSR experienced significant drops in performance, reflecting their struggle with pixel-wise accuracy under these altered conditions. In contrast, models such as PatchCore showed a more moderate decline, suggesting relatively better resilience at the pixel level. These results highlight that while data robustness ex-

periment tends to reduce pixel-level AUROC across models, some architectures are better suited to maintain their ability to accurately detect anomalies at finer, pixel-level granularity.

The pixel level F1 score plot indicates that most anomaly detection models experienced a decline in precision and recall balance at the pixel level after data robustness experiment, with models like DRAEM and DSR showing the most significant decreases. Despite the general downward trend, some models like PatchCore managed to limit the impact, suggesting a degree of resilience in maintaining pixel-level detection accuracy.

5. Discussion

We compared 17 different models with 28 backbones on our glue line image datasets, focusing on moderately challenging areas (part 1). Although the selected models have been extensively evaluated on public benchmarking datasets like MVTec AD, it remains crucial to assess their performance on our industrial datasets. Some models may perform exceptionally well on public datasets but have yet to be deployed in real industrial manufacturing environments, where conditions can be significantly different.

5.1. Multi Criteria Decision Making

Selecting the optimal model for our use case is challenging due to the wide range of performance metrics available, making it a multi-criteria decision-making (MCDM) problem. Among these metrics, inference time, classification accuracy, and segmentation accuracy are prioritized, as they directly align with the manufacturer's requirements. To address the inherent trade-offs in these criteria, we use *knee solutions*, where the models offer balanced performance across metrics without a strong preference for any specific criterion. Such knee solutions are considered "no preference" options in MCDM literature, providing a compromise that meets all major requirements without overly emphasizing one metric over others. Figure 13 illustrates the relationship between model inference time and overall performance, with models in the lower-right corner – closer to knee points –representing our top preferences. Note that the x-axis scale differs between plots, and not all x-axes start at zero.

Based on the model performance depicted in the graphs, our top choice is Efficient_AD, which consistently appears in the lower right corner of both the image-level and inference-time plots. This highlights its strong categorization capabilities combined with high operational efficiency. While Efficient_AD appears in the bottom left corner of the Pixel AUROC plot, this placement is due to the scale of the X-axis; its actual segmentation performance is well-demonstrated by the Pixel F1 Score plot. Our second choice is STFPM, which performs effectively with both backbones and is similarly located in the lower right corner of each plot, indicating its fast execution while maintaining accuracy.

To focus on model's performance metrics, we also compare the image level performance versus pixel level performance at Figure 14. The upper-right corner represents the optimal performance model. The presence of STFPM and MMR in the upper right corner of both graphs indicates that the MMR model is a viable option if operational efficiency is not a primary concern.

From Table 2, we can see that models based on student-teacher networks from feature-embedding based approach, such as STFPM, Efficient_AD, and MMR, generally perform better. Notably, both Efficient_AD and MMR also incorporate the autoencoder method which is reconstruction based, enhancing their performance in defect detection and localization.

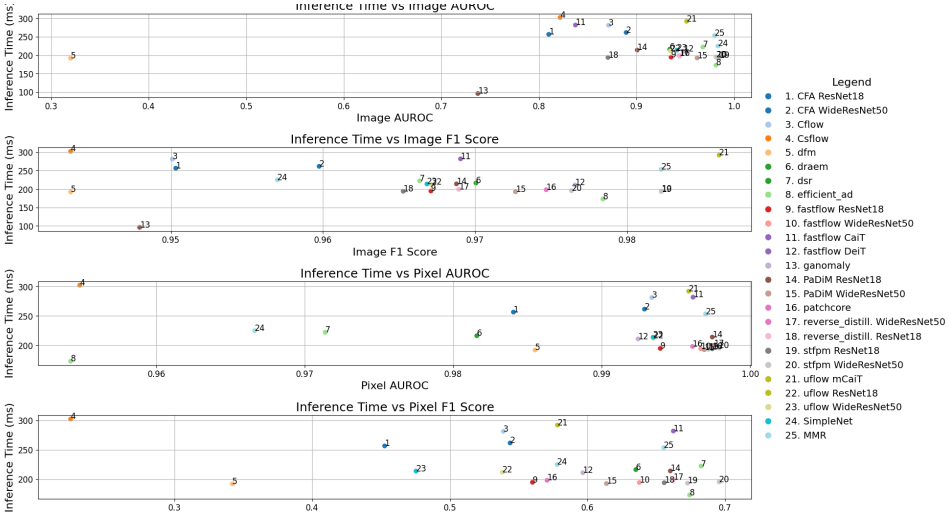


Figure 13: Model Inference Time vs Performance

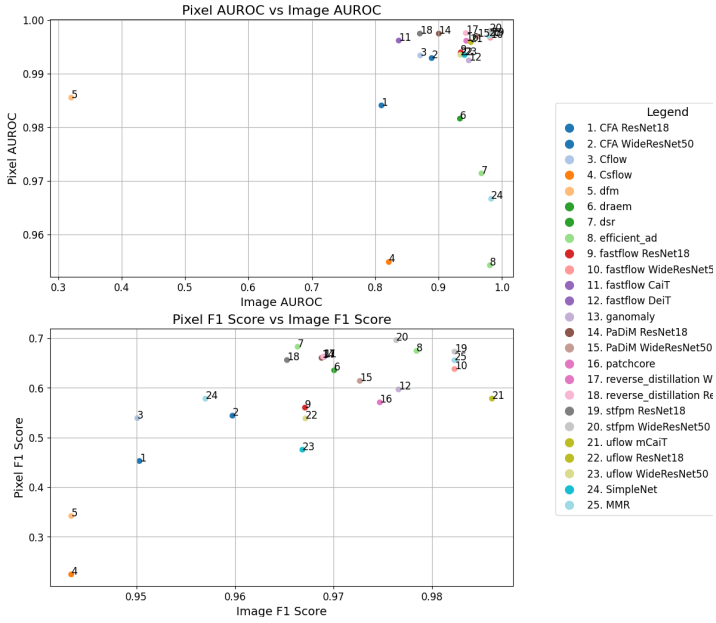


Figure 14: Image vs. Pixel Performance

5.2. Application to Other Components

For glue line detection in other components, certain representative sets of components require special attention. For instance, in part16, the significant variations in lightness and darkness make it challenging to accurately localize defects during image detection. The false positive rate for this part is really high, which is 5 times comparing to part1. This variability also complicates the use of unsupervised deep learning methods. The representative images are shown at Figure 15. From left to right images are normal image, image that is overexposed but pass the machine's inspection, and image that is misdetected. In this part, we will only apply the models that perform well for our previous part.

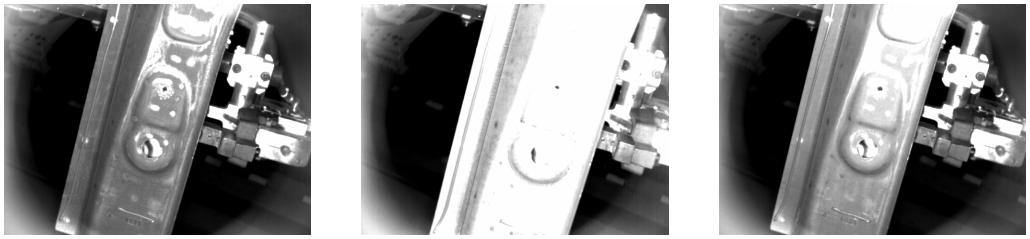


Figure 15: Part16 Representative Images

The results for part16 are shown at Table 6 and the results for other representative parts are shown in Appendix D. As shown in the table, the performance of individual models varies significantly for parts with pronounced variations in brightness. When considering the models' ability to differentiate between defects, only a few perform well. Notably, the Fastflow model with the cait backbone achieves the highest image-level AUROC of 0.927. The Uflow model with the WideResNet50 backbone also distinguishes defects effectively, with an AUROC of 0.884. However, many of the remaining models have image-level AUROCs below 0.5, and some are even lower than 0.15, such as the DRAEM and DSR model. This suggests that the DRAEM and DSR models are more sensitive to variations in brightness than the other models. For pixel level AUROC, the performance is better than image level. The significant difference between pixel and image-level performance could indicate that while the models are sensitive to small, localized anomalies, these anomalies may not be pronounced enough to influence the classification of the entire image. This could result in missed detections at the image level, potentially leading to lower overall performance in applications where image-level classification is critical. The F1 score at the image level remains relatively stable, hovering around 0.83. In contrast, the F1 scores at the pixel level for individual models are less impressive, with none exceeding 0.5.

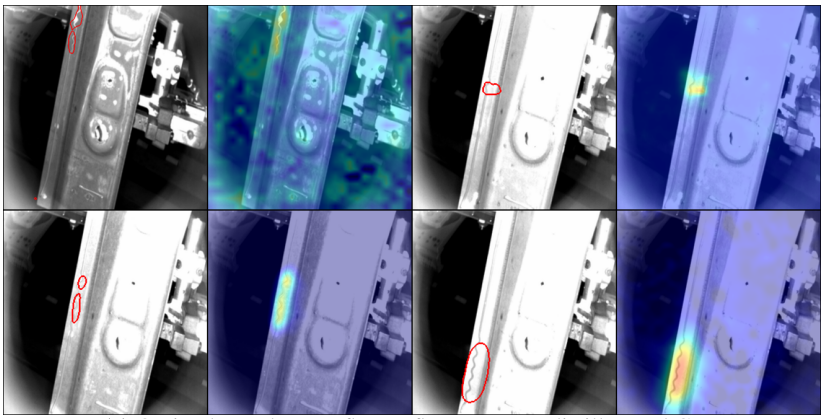
The performance of models across different regions, including both well-performing and underperforming ones, is shown in Figure 16. The top-performing models are Fastflow, Uflow, Reverse_Distill, and PatchCore, which accurately localize areas of glue line distortion. Their segmentation results are precise, effectively pinpointing the deformed regions.

In contrast, the underperforming models are DRAEM, CFA, CSFlow, and Efficient_AD. These models appear to be sensitive to brightness variations, which significantly impacts their classification performance, often falling below 0.5. The segmentation results are also suboptimal. For instance, DRAEM mistakenly segments many of the light and shadow changes on the parts as defective areas. The CFA model misidentifies variations at the ends of the glue lines as defects, while the CSFlow model's segmentation tends to be more random and inconsistent. Although Efficient_AD performs reasonably well in segmentation, it only captures a small portion

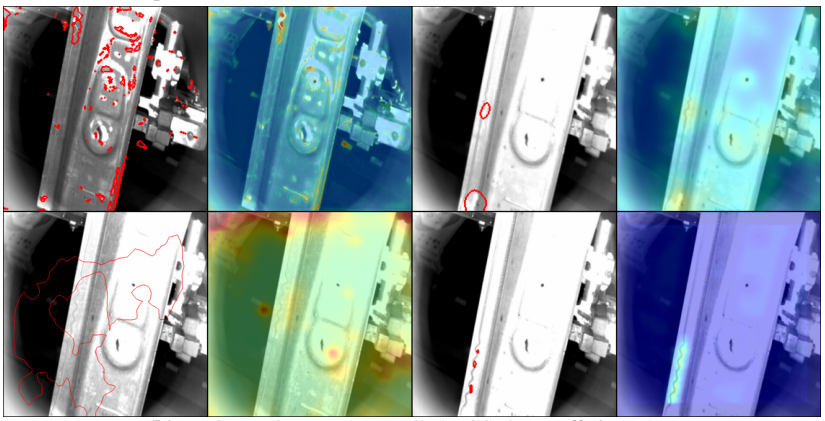
Table 6: Prediction performance metrics of Part16 for various unsupervised DL models and corresponding backbones.

Model	Backbone	Image AUROC	Image F1 Score	Pixel AUROC	Pixel F1 Score
CFA	ResNet18	0.165 ± 0.003	0.828 ± 0.000	0.715 ± 0.009	0.015 ± 0.002
	WideResNet50	0.128 ± 0.021	0.828 ± 0.000	0.790 ± 0.004	0.040 ± 0.008
CFlow	WideResNet50	0.576 ± 0.054	0.828 ± 0.000	0.966 ± 0.006	0.350 ± 0.050
CSFlow	EfficientNet-B5	0.148 ± 0.001	0.842 ± 0.000	0.491 ± 0.000	0.008 ± 0.000
DFM	ResNet50	0.313 ± 0.000	0.828 ± 0.000	0.929 ± 0.000	0.050 ± 0.000
DRAEM		0.093 ± 0.079	0.830 ± 0.006	0.877 ± 0.019	0.040 ± 0.010
DSR		0.143 ± 0.064	0.821 ± 0.015	0.661 ± 0.072	0.017 ± 0.005
Efficient_AD		0.296 ± 0.037	0.842 ± 0.000	0.848 ± 0.003	0.105 ± 0.006
FastFlow	ResNet18	0.352 ± 0.072	0.828 ± 0.000	0.810 ± 0.048	0.065 ± 0.049
	WideResNet50	0.822 ± 0.085	0.878 ± 0.031	0.983 ± 0.006	0.443 ± 0.027
	cait	**0.927 ± 0.061**	**0.941 ± 0.019**	0.977 ± 0.004	**0.492 ± 0.031**
	deit	0.832 ± 0.047	0.890 ± 0.035	0.965 ± 0.003	0.381 ± 0.026
GANomaly		0.791 ± 0.100	0.899 ± 0.074		
PaDiM	ResNet18	0.258 ± 0.039	0.828 ± 0.000	0.928 ± 0.008	0.148 ± 0.022
	WideResNet50	0.485 ± 0.082	0.830 ± 0.006	0.961 ± 0.003	0.177 ± 0.033
PatchCore	WideResNet50	0.601 ± 0.009	0.842 ± 0.000	0.975 ± 0.000	0.227 ± 0.003
Reverse_Distill.	ResNet18	0.218 ± 0.023	0.828 ± 0.000	0.946 ± 0.003	0.101 ± 0.009
	WideResNet50	0.630 ± 0.029	0.833 ± 0.013	0.984 ± 0.001	0.320 ± 0.020
stfpm	ResNet18	0.508 ± 0.156	0.833 ± 0.008	0.660 ± 0.133	0.017 ± 0.012
	WideResNet50	0.288 ± 0.288	0.833 ± 0.013	0.577 ± 0.038	0.010 ± 0.001
Uflow	mcait	0.632 ± 0.113	0.848 ± 0.031	0.968 ± 0.018	0.223 ± 0.069
	ResNet18	0.381 ± 0.061	0.828 ± 0.000	0.913 ± 0.025	0.086 ± 0.013
	WideResNet50	0.884 ± 0.028	0.896 ± 0.009	**0.986 ± 0.001**	0.302 ± 0.010

of the glue line deformation, failing to cover the full extent of the defect, which shows that this model is sensitive to brightness.



(a) Optimal Results, Fastflow, Uflow, Reverse_distill., PatchCore



(b) Bad Results, DRAEM, CFA, CSFlow, Efficient_AD

Figure 16: Segmentation and Heatmap for Part16

5.3. Contribution and Limitation

The main contribution of this study is the comprehensive comparison of a wide variety of existing anomaly detection models applied to a real industrial use case involving multiple scenarios. Unlike most empirical studies in the literature that rely on benchmark anomaly detection image datasets, which often fail to capture the variability of real-world industrial conditions (Wilmet et al., 2021; Cui et al., 2023), this study uniquely evaluates models on a dataset derived from a production environment. A thorough comparison of existing models on images from actual manufacturing settings has not been comprehensively addressed in previous research. Our evaluation spans multiple dimensions, including image-level accuracy, pixel-level accuracy, training and inference times, and the robustness of models to variations in image quality.

Additionally, this study introduces a MCDM-based approach for model selection, enabling decision-makers to achieve optimal trade-offs between accuracy and inference time. This dual

contribution—comprehensive evaluation and actionable model selection framework—addresses key challenges in deploying image anomaly detection systems in industrial contexts.

In this work, we assessed models from two major categories: reconstruction-based and feature-embedding-based approaches. Specifically, our analysis focused on temporal performance, segmentation precision, and robustness in industrial environments. These evaluations directly address RQ1, identifying models that deliver optimal accuracy and efficiency for detecting and segmenting anomalies in glue-line images.

Among the models evaluated, Efficient_AD and STFPM demonstrated superior segmentation precision, accuracy and efficiency, particularly in localizing small and subtle anomalies, which is critical in minimizing false positives. While Fastflow is robust and performs very well on most difficult component that has extreme brightness contrast, part16. By reducing the rate of false positives, we were able to significantly enhance the overall reliability of the automated inspection system. This finding aligns with our primary objective of improving anomaly detection accuracy and operational efficiency. The reduction of false positives not only decreases unnecessary maintenance checks and system downtimes, but also boosts the overall equipment effectiveness (OEE) of the glue system, a key performance indicator in industrial settings.

To address RQ2, we addressed the challenge of limited defect class datasets in industrial applications by incorporating data augmentation and synthetic data generation techniques. These methods enriched the diversity of training data, improving model generalization to unseen anomalies. Such strategies are especially valuable in unsupervised settings, where labeled data is scarce or expensive to acquire. Additionally, our data augmentation experiments ensured that the models' robustness extended beyond mere accuracy metrics, proving their practical applicability. These findings underscore the effectiveness of data augmentation in mitigating dataset imbalances, providing a viable solution for enhancing model performance despite limited defect-specific datasets.

This comparative study provides a roadmap for selecting the most suitable unsupervised deep learning model tailored to specific industrial needs. It also demonstrates a clear reduction in false positives, minimizing glue machine downtimes and further enhancing the OEE of the entire glue-line system.

Despite these contributions, the study has certain limitations. For instance, the dataset used is single-channel, potentially limiting the approach's applicability to other industrial datasets, where models may behave differently with three-channel images. Future research could explore the integration of thermal imaging to introduce an additional channel, further optimizing model performance.

Moreover, uncertainties introduced by our data synthesis methods may limit generalizability, and not all types of unsupervised deep learning models were evaluated. Future studies could investigate diffusion models and transformer-based architectures. Comparing supervised, semi-supervised, and weakly supervised models on the glue-line dataset could also provide deeper insights and potentially enhance performance further.

6. Conclusions

In this study, we conducted a comprehensive comparison of various unsupervised deep learning models for anomaly detection in industrial glue lines. We assessed the models' accuracy, segmentation precision, and robustness to identify the most stable and effective model for our specific industrial environment. The models successfully detected and localized defects in glue

lines, significantly reducing the high false-positive rate. Additionally, through the application of data sythesis techniques, we addressed the challenge of limited datasets, enhancing the models' performance and reliability. These findings demonstrate the potential for real-world application in industrial image anomaly detection, improving both the accuracy of defect detection and the overall efficiency of the glue line system.

Acknowledgements

The study was supported by the Swedish innovation agency VINNOVA under grant number 2021-02537 (Integrated Manufacturing Analytics Platform, IMAP project). The computation was enabled by resources provided by Chalmers e-Commons at Chalmers. The work was carried out within Chalmers' Area of Advanced Production whose support is greatly acknowledged. During the preparation of this work the authors used ChatGPT 4o in order to proofread and enhance readability. After using this tool, the authors reviewed and edited the content and take full responsibility.

References

Ahuja, N.A., Ndiour, I., Kalyanpur, T., Tickoo, O., 2019. Probabilistic modeling of deep features for out-of-distribution and adversarial detection. arXiv preprint arXiv:1909.11786 .

Akcay, S., Ameln, D., Vaidya, A., Lakshmanan, B., Ahuja, N., Genc, U., 2022. Anomalib: A deep learning library for anomaly detection, in: 2022 IEEE International Conference on Image Processing (ICIP), IEEE. pp. 1706–1710.

Akcay, S., Atapour-Abarghouei, A., Breckon, T.P., 2019. Ganomaly: Semi-supervised anomaly detection via adversarial training, in: Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III 14, Springer. pp. 622–637.

AlbumentationsAI, 2024. Albumentations documentation: Defocus. `https://albumentations.ai/docs/api_reference/augmentations/blur/transforms/`. Accessed: 2024-12-26.

Alex, K., 2009. Learning multiple layers of features from tiny images. https://www. cs. toronto. edu/kriz/learning-features-2009-TR. pdf .

Batzner, K., Heckler, L., König, R., 2024. Efficientad: Accurate visual anomaly detection at millisecond-level latencies, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 128–138.

Bergmann, P., Batzner, K., Fauser, M., Sattlegger, D., Steger, C., 2021. The mvtec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection. International Journal of Computer Vision 129, 1038–1059.

Bergmann, P., Fauser, M., Sattlegger, D., Steger, C., 2020. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 4183–4192.

Božič, J., Tabernik, D., Skočaj, D., 2021. Mixed supervision for surface-defect detection: From weakly to fully supervised learning. Computers in Industry 129, 103459.

Bradski, G., 2000. The OpenCV Library. Dr. Dobb's Journal of Software Tools .

Buda, M., Saha, A., Mazurowski, M.A., 2019. Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm. Computers in biology and medicine 109, 218–225.

Chen, S.H., Tsai, C.C., 2021. Smd led chips defect detection using a yolov3-dense model. Advanced engineering informatics 47, 101255.

Chu, W.H., Kitani, K.M., 2020. Neural batch sampling with reinforcement learning for semi-supervised anomaly detection, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16, Springer. pp. 751–766.

Chung, H., Park, J., Keum, J., Ki, H., Kang, S., 2020. Unsupervised anomaly detection using style distillation. IEEE Access 8, 221494–221502.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., , Vedaldi, A., 2014. Describing textures in the wild, in: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).

Cohen, N., Hoshen, Y., 2020. Sub-image anomaly detection with deep pyramid correspondences. arXiv preprint arXiv:2005.02357 .

Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., Bharath, A.A., 2018. Generative adversarial networks: An overview. IEEE signal processing magazine 35, 53–65.

Croitoru, F.A., Hondru, V., Ionescu, R.T., Shah, M., 2023. Diffusion models in vision: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 45, 10850–10869.

Cui, Y., Liu, Z., Lian, S., 2023. A survey on unsupervised anomaly detection algorithms for industrial images. IEEE Access 11, 55297–55315.

Defard, T., Setkov, A., Loesch, A., Audigier, R., 2021. Padim: a patch distribution modeling framework for anomaly detection and localization, in: International Conference on Pattern Recognition, Springer. pp. 475–489.

Dehaene, D., Eline, P., 2020. Anomaly localization by modeling perceptual features. arXiv preprint arXiv:2008.05369 .

Dehaene, D., Frigo, O., Combrexelle, S., Eline, P., 2020. Iterative energy-based projection on a normal data manifold for anomaly localization. arXiv preprint arXiv:2002.03734 .

Deng, H., Li, X., 2022. Anomaly detection via reverse distillation from one-class embedding, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9737–9746.

GIMPDoc, 2020. Warp tool. URL: https://docs.gimp.org/2.10/en/gimp-tool-warp.html. retrieved January 2, 2025.

Gudovskiy, D., Ishizaka, S., Kozuka, K., 2022. Cflow-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, pp. 98–107.

Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., Tang, Y., Xiao, A., Xu, C., Xu, Y., et al., 2022. A survey on vision transformer. IEEE transactions on pattern analysis and machine intelligence 45, 87–110.

Haselmann, M., Gruber, D.P., Tabatabai, P., 2018. Anomaly detection using deep learning based image completion, in: 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 1237–1242. doi:10.1109/ICMLA.2018.00201.

Huang, Y., Qiu, C., Yuan, K., 2020. Surface defect saliency of magnetic tile. The Visual Computer 36, 85–96.

Iqbal, H., Khalid, U., Chen, C., Hua, J., 2023. Unsupervised anomaly detection in medical images using masked diffusion model, in: International Workshop on Machine Learning in Medical Imaging, Springer. pp. 372–381.

Jeong, K.J., Park, J.D., Hwang, K., Kim, S.L., Shin, W.Y., 2022. Two-stage deep anomaly detection with heterogeneous time series data. IEEE Access 10, 13704–13714. URL: https://ieeexplore.ieee.org/document/9695481/, doi:10.1109/ACCESS.2022.3147188.

Jiang, J., Zhu, J., Bilal, M., Cui, Y., Kumar, N., Dou, R., Su, F., Xu, X., 2022. Masked swin transformer unet for industrial anomaly detection. IEEE Transactions on Industrial Informatics 19, 2200–2209.

Jiang, Y., Wang, W., Zhao, C., 2019. A machine vision-based realtime anomaly detection method for industrial products using deep learning, IEEE. pp. 4842–4847. URL: https://ieeexplore.ieee.org/document/8997079/, doi:10.1109/CAC48633.2019.8997079.

Kagawade, V.C., Angadi, S.A., 2021. Visa: a multimodal database of face and iris traits. Multimedia Tools and Applications 80, 21615–21650.

Kingma, D.P., Welling, M., et al., 2019. An introduction to variational autoencoders. Foundations and Trends® in Machine Learning 12, 307–392.

Lee, S., Lee, S., Song, B.C., 2022. Cfa: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. IEEE Access 10, 78446–78454.

Li, C.L., Sohn, K., Yoon, J., Pfister, T., 2021. Cutpaste: Self-supervised learning for anomaly detection and localization, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9664–9674.

Li, J., Su, Z., Geng, J., Yin, Y., 2018. Real-time detection of steel strip surface defects based on improved yolo detection network. IFAC-PapersOnLine 51, 76–81.

Liang, Y., Zhang, J., Zhao, S., Wu, R., Liu, Y., Pan, S., 2023. Omni-frequency channel-selection representations for unsupervised anomaly detection. IEEE Transactions on Image Processing .

Lin, D., Li, Y., Prasad, S., Nwe, T.L., Dong, S., Oo, Z.M., 2020. Cam-unet: class activation map guided unet with feedback refinement for defect segmentation, in: 2020 IEEE International Conference on Image Processing (ICIP), IEEE. pp. 2131–2135.

Liu, J., Liang, W., Chen, Y., Jin, K., 2024a. Synet:medical image anomaly detection with noise synthesis network, in: 2024 IEEE 10th Conference on Big Data Security on Cloud (BigDataSecurity), pp. 96–100. doi:10.1109/BigDataSecurity62737.2024.00024.

Liu, J., Xie, G., Wang, J., Li, S., Wang, C., Zheng, F., Jin, Y., 2024b. Deep industrial image anomaly detection: A survey. Machine Intelligence Research 21, 104–135.

Liu, W., W. Luo, D.L., Gao, S., 2018. Future frame prediction for anomaly detection – a new baseline, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Liu, Z., Zhou, Y., Xu, Y., Wang, Z., 2023. Simplenet: A simple network for image anomaly detection and localization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20402–20411.

Livernoche, V., Jain, V., Hezaveh, Y., Ravanbakhsh, S., 2023. On diffusion modeling for anomaly detection. arXiv

preprint arXiv:2305.18593 .

Liznerski, P., Ruff, L., Vandermeulen, R.A., Franks, B.J., Kloft, M., Müller, K.R., 2020. Explainable deep one-class classification. arXiv preprint arXiv:2007.01760 .

Lu, B., Xu, D., Huang, B., 2022. Deep-learning-based anomaly detection for lace defect inspection employing videos in production line. Advanced Engineering Informatics 51, 101471. URL: https://www.sciencedirect.com/science/article/pii/S1474034621002214, doi:https://doi.org/10.1016/j.aei.2021.101471.

Luo, D., Cai, Y., Yang, Z., Zhang, Z., Zhou, Y., Bai, X., 2022. Suevey on industrial defect detection with deep learning (in chinese). Sci Sin Inform 52, 1002–1039. doi:10.1360/SSI-2021-0336.

Ma, M., Han, L., Zhou, C., 2023. Btad: A binary transformer deep neural network model for anomaly detection in multivariate time series data. Advanced Engineering Informatics 56, 101949.

Maláková, S., Guzanová, A., Frankovskỳ, P., Neumann, V., Janoško, E., 2019. Glued joints in the automotive industry. Acta Mechatronica 4, 23–28.

Mishra, P., Verk, R., Fornasier, D., Piciarelli, C., 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), IEEE. pp. 01–06.

Pang, G., Shen, C., Cao, L., Hengel, A.V.D., 2021. Deep learning for anomaly detection: A review. ACM Comput. Surv. 54. URL: https://doi.org/10.1145/3439950, doi:10.1145/3439950.

Posilović, L., Medak, D., Milković, F., Subašić, M., Budimir, M., Lončarić, S., 2022. Deep learning-based anomaly detection from ultrasonic images. Ultrasonics 124, 106737. URL: https://www.sciencedirect.com/science/article/pii/S0041624X2200049X, doi:https://doi.org/10.1016/j.ultras.2022.106737.

Prezas, L., Michalos, G., Arkouli, Z., Katsikarelis, A., Makris, S., 2022. Ai-enhanced vision system for dispensing process monitoring and quality control in manufacturing of large parts. Procedia CIRP 107, 1275–1280. URL: https://www.sciencedirect.com/science/article/pii/S2212827122004280, doi:https://doi.org/10.1016/j.procir.2022.05.144. leading manufacturing systems transformation – Proceedings of the 55th CIRP Conference on Manufacturing Systems 2022.

Qiu, L., Wu, X., Yu, Z., 2019. A high-efficiency fully convolutional networks for pixel-wise surface defect detection. IEEE Access 7, 15884–15893.

Rezende, D., Mohamed, S., 2015. Variational inference with normalizing flows, in: International conference on machine learning, PMLR. pp. 1530–1538.

Rippel, O., Mertens, P., 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), pp. 6726–6733. doi:10.1109/ICPR48806.2021.9412109.

Rippel, O., Müller, M., Merhof, D., 2020. Gan-based defect synthesis for anomaly detection in fabrics, in: 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), pp. 534–540. doi:10.1109/ETFA46521.2020.9212099.

Rogers, T.W., Jaccard, N., Morton, E.J., Griffin, L.D., 2017. Automated x-ray image analysis for cargo security: Critical review and future promise. Journal of X-ray science and technology 25, 33–56.

Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., Gehler, P., 2022. Towards total recall in industrial anomaly detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 14318–14328.

Rudolph, M., Wandt, B., Rosenhahn, B., 2021. Same same but differnet: Semi-supervised defect detection with normalizing flows, in: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp. 1907–1916.

Rudolph, M., Wehrbein, T., Rosenhahn, B., 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, pp. 1088–1097.

Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S.A., Binder, A., Müller, E., Kloft, M., 2018. Deep one-class classification, in: International conference on machine learning, PMLR. pp. 4393–4402.

Sasaki, H., Willcocks, C.G., Breckon, T.P., 2021. Unit-ddpm: Unpaired image translation with denoising diffusion probabilistic models. arXiv preprint arXiv:2104.05358 .

Schlegl, T., Seeböck, P., Waldstein, S.M., Langs, G., Schmidt-Erfurth, U., 2019. f-anogan: Fast unsupervised anomaly detection with generative adversarial networks. Medical image analysis 54, 30–44.

Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G., 2017. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery, in: International conference on information processing in medical imaging, Springer. pp. 146–157.

Schlüter, H.M., Tan, J., Hou, B., Kainz, B., 2021. Self-supervised out-of-distribution detection and localization with natural synthetic anomalies (nsa). arXiv preprint arXiv:2109.15222 2.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D., 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE international conference on computer vision, pp. 618–626.

SICK AG, 2024. Robot Guidance with SICK Inspector. URL: https://cobots.se/shop/universal-robots/tillbehor/visionsystem1/robot-guidance-with-sick-inspector/. operating Instructions Handbook.

Söderberg, R., Lindkvist, L., Wärmefjord, K., Carlson, J.S., 2016. Virtual Geometry Assurance Process and Toolbox, in: Procedia CIRP, Elsevier B.V.. pp. 3–12. doi:10.1016/j.procir.2016.02.043.

Song, J., Kong, K., Park, Y.I., Kim, S.G., Kang, S.J., 2021. Anoseg: Anomaly segmentation network using self-supervised learning. arXiv preprint arXiv:2110.03396 .

Staar, B., Lütjen, M., Freitag, M., 2019. Anomaly detection with convolutional neural networks for industrial surface inspection. Procedia CIRP 79, 484–489. URL: https://www.sciencedirect.com/science/article/pii/S2212827119302409, doi:https://doi.org/10.1016/j.procir.2019.02.123. 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 18-20 July 2018, Gulf of Naples, Italy.

Tailanian, M., Pardo, Á., Musé, P., 2022. U-flow: A u-shaped normalizing flow for anomaly detection with unsupervised threshold. arXiv preprint arXiv:2211.12353 .

Tang, T.W., Kuo, W.H., Lan, J.H., Ding, C.F., Hsu, H., Young, H.T., 2020. Anomaly detection neural network with dual auto-encoders gan and its industrial inspection applications. Sensors 20. URL: https://www.mdpi.com/1424-8220/20/12/3336, doi:10.3390/s20123336.

Tayeh, T., Aburakhia, S., Myers, R., Shami, A., 2020. Distance-based anomaly detection for industrial surfaces using triplet networks, in: 2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), pp. 0372–0377. doi:10.1109/IEMCON51383.2020.9284921.

UK Home Office Centre for Applied Science and Technology (CAST), 2016. OSCT Borders X-ray Image Library. Technical Report 146/16. UK Home Office Centre for Applied Science and Technology (CAST). Publication Number: 146/16.

Wan, Q., Gao, L., Li, X., 2022. Logit inducing with abnormality capturing for semi-supervised image anomaly detection. IEEE Transactions on Instrumentation and Measurement 71, 1–12.

Wang, G., Han, S., Ding, E., Huang, D., 2021. Student-teacher feature pyramid matching for unsupervised anomaly detection. arxiv 2021. arXiv preprint arXiv:2103.04257 1.

Wang, L., Zhang, D., Guo, J., Han, Y., 2020. Image anomaly detection using normal data only by latent space resampling. Applied Sciences 10, 8660.

Weihan, W., 2020. Magan: A masked autoencoder generative adversarial network for processing missing iot sequence data. Pattern Recognition Letters 138, 211–216. URL: https://linkinghub.elsevier.com/retrieve/pii/S0167865520302713, doi:10.1016/j.patrec.2020.07.025.

Wilmet, V., Verma, S., Redl, T., Sandaker, H., Li, Z., 2021. A comparison of supervised and unsupervised deep learning methods for anomaly detection in images. arXiv preprint arXiv:2107.09204 .

Xiao, H., Rasul, K., Vollgraf, R., 2017. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 .

Yang, J., Shi, Y., Qi, Z., 2020. Dfr: Deep feature reconstruction for unsupervised anomaly segmentation. arXiv preprint arXiv:2012.07122 .

Yi, J., Yoon, S., 2020. Patch svdd: Patch-level svdd for anomaly detection and segmentation, in: Proceedings of the Asian conference on computer vision.

Yu, J., Zheng, Y., Wang, X., Li, W., Wu, Y., Zhao, R., Wu, L., 2021. Fastflow: Unsupervised anomaly detection and localization via 2d normalizing flows. arXiv preprint arXiv:2111.07677 .

Zavrtanik, V., Kristan, M., Skočaj, D., 2021. Draem-a discriminatively trained reconstruction embedding for surface anomaly detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8330–8339.

Zavrtanik, V., Kristan, M., Skočaj, D., 2022. Dsr–a dual subspace re-projection network for surface anomaly detection, in: European conference on computer vision, Springer. pp. 539–554.

Zeng, Z., Liu, B., Fu, J., Chao, H., 2021. Reference-based defect detection network. IEEE Transactions on Image Processing 30, 6637–6647.

Zhang, G., Cui, K., Hung, T.Y., Lu, S., 2021. Defect-gan: High-fidelity defect synthesis for automated defect inspection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 2524–2534.

Zhang, Z., Zhao, Z., Zhang, X., Sun, C., Chen, X., 2023. Industrial anomaly detection with domain shift: A real-world dataset and masked multi-scale reconstruction. arXiv preprint arXiv:2304.02216 .

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A., 2016. Learning deep features for discriminative localization, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2921–2929.

Zhou, K., Hong, L., Chen, C., Xu, H., Ye, C., Hu, Q., Li, Z., 2022. Devnet: Self-supervised monocular depth learning via density volume construction, in: European Conference on Computer Vision, Springer. pp. 125–142.

Zipfel, J., Verworner, F., Fischer, M., Wieland, U., Kraus, M., Zschech, P., 2023. Anomaly detection for industrial quality assurance: A comparative evaluation of unsupervised deep learning models. Computers & Industrial Engineering 177, 109045. URL: https://www.sciencedirect.com/science/article/pii/S0360835223000694, doi:https://doi.org/10.1016/j.cie.2023.109045.

Appendix A. Selected unsupervised deep learning models' accuracy performance

Table A.7: Selected unsupervised deep learning models' accuracy performance on MVTec AD benchmarking dataset

Model	Backbone	Image AUROC
CFA	ResNet18	0.930
	WideResNet50	0.956
CFlow	WideResNet50	0.962
CSFlow	EfficientNet-B5	0.987
DFKDE	ResNet18	0.762
	WideResNet50	0.774
DFM	ResNet50	0.936
DRAEM		0.980
DSR		0.982
Efficient_AD		0.982
FastFlow	ResNet18	0.907
	WideResNet50	0.963
	CaiT	0.925
	DeiT	0.944
GANomaly		0.421
PaDiM	ResNet18	0.891
	WideResNet50	0.950
PatchCore	WideResNet50	0.980
Reverse_Distillation	WideResNet50	0.985
	ResNet18	0.978
stfpm	ResNet18	0.893
	WideResNet50	0.876
Uflow	mcait	0.987
	ResNet18	0.942
	WideResNet50	0.968
MMR	WideResNet50	0.984
SimpleNet	WideResNet50	0.996

Appendix B. Choice of the Hyperparameters

Table B.8: Hyperparameters Used for Deep Learning Models

Model	Tuning Parameters	Implemented Values in this Study
PaDiM	Backbone	ResNet18, WideResNet50_2
	Layers	(layer1 + layer2 + layer3)
	Image Resolution	256 * 256
	Train Batch Size	32

	Eval Batch Size	32
	Normalization	Imagenet
	Normalization Method	min_max
	Max Epochs	1
CFA	Backbone	ResNet18, WideResNet50_2
	Image Resolution	224 * 224
	Train Batch Size	4
	Eval Batch Size	4
	Inference Batch Size	4
	Normalization	imagenet
	Normalization Method	min_max
	Max Epochs	30
	Learning Rate	1.00E-03
	Weight Decay	5.00E-04
	gamma_c	1.00E+00
	gamma_d	1.00E+00
	num_nearest_neighbors	3.00E+00
Cflow	Backbone	WideResNet50_2
	Layers	(layer2 + layer3 + layer4)
	Decoder	freia-cflow
	Image Resolution	256 * 256
	condition_vector	128
	coupling_blocks	8
	clamp_alpha	1.9
	fiber_batch_size	64
	Train Batch Size	16
	Eval Batch Size	16
	Inference Batch Size	16
	Normalization	imagenet
	Normalization Method	min_max
	Max Epochs	50
	Learning Rate	0.0001
Csflow	Backbone	EfficientNet-B5
	Layers	6.8
	Image Resolution	768 * 768
	Train Batch Size	16
	Eval Batch Size	16
	Normalization	imagenet
	Normalization Method	min_max
	Max Epochs	240
	Learning Rate	2.00E-04
	Weight Decay	1.00E-05
	eps	1.00E-04
dfkde	Backbone	ResNet18, WideResNet50_2
	Layers	layer4
	Image Resolution	256 * 256
	Train Batch Size	32
	Eval Batch Size	32
	n_pca_components	16

	max_training_points	40000
	feature_scaling_method (scale, norm)	scale
	Normalization	imagenet
	Normalization Method	min_max
	Max Epochs	1
dfm	Backbone	Resnet18, Resnet50
	Layers	layer3
	pooling Kernel Size	2
	PCA Level	0.97
	Score Type	pca feature reconstruction error
	Image Resolution	256 * 256
	Train Batch Size	32
	Eval Batch Size	32
	Normalization	imagenet
	Normalization Method	min_max
	Max Epochs	1
draem	Image Resolution	256 * 256
	Train Batch Size	8
	Eval Batch Size	32
	Normalization	none
	Normalization Method	min_max
	Max Epochs	700
	Learning Rate	0.0001
	beta	[0.1, 1.0]
	sspcab lambda	0.1
dsr	Image Resolution	256 * 256
	Train Batch Size	8
	Eval Batch Size	16
	Normalization	none
	Normalization Method	none
	Max Epochs	700
	Learning Rate	0.0002
	latent_anomaly_strength	0.2
	upsampling_train_ratio	0.7
efficient_ad	Image Resolution	256 * 256
	Train Batch Size	1
	Eval Batch Size	16
	Normalization	none
	teacher_out_channels	384
	Normalization Method	min_max
	Max Epochs	200
	Learning Rate	0.0001
	weight_decay	0.00001
GANomaly	Image Resolution	256 * 256
	Train Batch Size	32
	Eval Batch Size	32
	Inference Batch Size	32

	Normalization	imagenet
	Normalization Method	none
	Max Epochs	100
	Learning Rate	0.0002
	beta1	0.5
	beta2	0.999
	wadv	1
	wcon	50
	wenc	1
patchcore	Backbone	wide50_resnet50_2
	Layers	(layer2 + layer3)
	Image Resolution	256 * 256
	Train Batch Size	32
	Eval Batch Size	32
	Normalization	imagenet
	Normalization Method	min_max
	Max Epochs	1
	coreset_sampling_ratio	0.1
	num_neighbors	9
reverse_distill.	Backbone	ResNet18, WideResNet50_2
	Layers	(layer1 + layer2 + layer3)
	Image Resolution	256 * 256
	Train Batch Size	16
	Eval Batch Size	32
	Inference Batch Size	32
	Normalization	imagenet
	Normalization Method	min_max
	Max Epochs	200
	Learning Rate	0.005
	beta1	0.5
	beta2	0.999
stfpm	Backbone	ResNet18, WideResNet50_2
	Layers	(layer1 + layer2 + layer3)
	Image Resolution	256 * 256
	Train Batch Size	32
	Eval Batch Size	32
	Inference Batch Size	32
	Normalization	imagenet
	Normalization Method	min_max
	Max Epochs	100
	Learning Rate	0.4
	Weight Decay	0.0001
	momentum	0.9
fastflow	Backbone	ResNet18[1], WideResNet50_2[2], cait_m48_448[3], deit_base_distilled_patch16_384[4]
	Image Resolution	256 * 256[1, 2], 448 * 448[3], 384 * 384[4]
	Train Batch Size	32
	Eval Batch Size	32
	Normalization	Imagenet

	Normalization Method	min_max
	max epochs	500
	Learning rate	0.001
	weight decay	0.00001
	flow steps	8[1, 2], 20[3, 4]
	hidden ratio	1[1, 2], 0.16[3, 4]
	conv3x3 only	TRUE[1, 2], FALSE[3, 4]
Uflow	Backbone	mcait[1], ResNet18[2], WideResNet50_2[3]
	Image Resolution	448 * 448[1], 256 * 256[2, 3]
	Train Batch Size	14
	Eval Batch Size	16
	Inference Batch Size	16
	Normalization	Imagenet
	Normalization Method	min_max
	max epochs	200
	Learning Rate	0.001
	Weight Decay	0.00001
	Flow Steps	4
	Affine Clamp	2
	affine subnet channels ratio	1
SimpleNet	Backbone	WideResNet50
	layers	layer2 + layer3
	Batch Size	8
	Image Resolution	288 * 288
MMR	Backbone	WideResNet50
	layers	layer1 + layer2 + layer3
	Image Resolution	256 * 256
	epochs	200
	warmup epochs	50
	Learning Rate	0.001
	Weight Decay	0.05

Appendix C. Experimental Results for Robustness Study

Table C.9: Robustness Anomaly Detection Results

Model	Backbone	Image AUROC	Image F1 Score	Pixel AUROC	Pixel F1 Score
CFA	ResNet18	0.599 ± 0.118	0.937 ± 0.011	0.735 ± 0.012	0.034 ± 0.003
	WideResNet50	0.639 ± 0.137	0.935 ± 0.005	0.758 ± 0.025	0.087 ± 0.026
CFlow	WideResNet50	0.755 ± 0.146	0.938 ± 0.023	0.689 ± 0.025	0.140 ± 0.048
CSFlow	EfficientNet-B5	0.584 ± 0.134	0.931 ± 0.017	0.536 ± 0.025	0.014 ± 0.002
CutPaste		0.603 ± 0.173	0.931 ± 0.020	0.647 ± 0.026	0.237 ± 0.083
DFKDE	ResNet18	0.610 ± 0.060	0.919 ± 0.025	-	-
	WideResNet50	0.719 ± 0.078	0.920 ± 0.026	-	-
DFM	ResNet50	0.482 ± 0.016	0.941 ± 0.005	0.739 ± 0.041	0.113 ± 0.008
DRAEM		0.448 ± 0.155	0.931 ± 0.017	0.644 ± 0.018	0.025 ± 0.007
DSR		0.551 ± 0.111	0.929 ± 0.009	0.494 ± 0.056	0.043 ± 0.020
FastFlow	ResNet18	0.669 ± 0.091	0.943 ± 0.000	0.691 ± 0.047	0.133 ± 0.030
	WideResNet50	0.735 ± 0.086	0.913 ± 0.039	0.674 ± 0.055	0.203 ± 0.104
	cait_m48_448	0.627 ± 0.118	0.935 ± 0.005	0.758 ± 0.051	0.236 ± 0.096
	deit384	0.770 ± 0.063	0.932 ± 0.023	0.695 ± 0.019	0.152 ± 0.078
GANomaly		0.567 ± 0.080	0.943 ± 0.000	-	-
MMR	WideResNet50	0.824 ± 0.023	0.942 ± 0.005	0.953 ± 0.032	0.207 ± 0.004
PaDiM	ResNet18	0.634 ± 0.134	0.939 ± 0.009	0.773 ± 0.021	0.218 ± 0.073
	WideResNet50	0.682 ± 0.122	0.935 ± 0.014	0.806 ± 0.049	0.233 ± 0.075
PatchCore	WideResNet50	0.717 ± 0.082	0.923 ± 0.020	0.830 ± 0.023	0.279 ± 0.048
Reverse_Distill.	WideResNet50	0.572 ± 0.062	0.935 ± 0.008	0.766 ± 0.066	0.261 ± 0.101
	ResNet-18	0.575 ± 0.153	0.937 ± 0.009	0.762 ± 0.068	0.334 ± 0.027
stfpm	ResNet18	0.573 ± 0.121	0.937 ± 0.009	0.700 ± 0.058	0.197 ± 0.046
	WideResNet50	0.595 ± 0.141	0.933 ± 0.013	0.649 ± 0.028	0.132 ± 0.048
SimpleNet	WideResNet50	0.869 ± 0.026	0.943 ± 0.012	0.899 ± 0.014	0.152 ± 0.008
UFlow	mcait	0.753 ± 0.088	0.931 ± 0.021	0.752 ± 0.017	0.292 ± 0.042
	ResNet18	0.645 ± 0.070	0.939 ± 0.006	0.705 ± 0.030	0.185 ± 0.041
	WideResNet50	0.585 ± 0.121	0.933 ± 0.010	0.734 ± 0.030	0.237 ± 0.017

Appendix D. Experimental Results for Additional Parts

Table D.10: Anomaly Detection Results on Part6 Dataset (Default Random Seed: 42)

Model Name	Backbone	Image AUROC	Image F1Score	Pixel AUROC	Pixel F1Score
CFA	ResNet18	0.595	0.667	0.935	0.295
	WideResNet50	0.648	0.640	0.966	0.358
CFlow	WideResNet50	0.991	0.968	0.997	0.599
CSFlow	EfficientNet-B5	0.287	0.542	0.430	0.013
DFKDE	ResNet18	0.081	0.542	-	-
	WideResNet50	0.107	0.542	-	-
DFM	ResNet50	0.375	0.542	0.992	0.463
DRAEM	-	0.639	0.629	0.955	0.378
DSR	-	0.646	0.600	0.933	0.308
Efficient_AD	-	0.900	0.875	0.957	0.561
FastFlow	ResNet18	0.940	0.968	0.990	0.579
	WideResNet50	0.998	0.970	0.997	0.679
	cait	0.940	0.968	0.997	0.654
	deit	0.940	0.968	0.995	0.700
GANomaly	-	0.169	0.542	-	-
MMR	WideResNet50	0.958	0.968	0.995	0.681
PaDiM	ResNet18	0.968	0.938	0.997	0.668
	WideResNet50	0.970	0.968	0.997	0.622
PatchCore	WideResNet50	0.951	0.968	0.994	0.603
Reverse_Distillation	ResNet18	0.900	0.815	0.995	0.616
	WideResNet50	0.991	0.968	0.997	0.681
stfpm	ResNet18	0.875	0.897	0.992	0.596
	WideResNet50	0.938	0.968	0.996	0.693
SimpleNet	WideResNet50	1.000	0.970	0.996	0.700
UFlow	mcait	0.995	0.968	0.998	0.655
	ResNet18	0.988	0.933	0.993	0.558
	WideResNet50	0.979	0.933	0.994	0.554

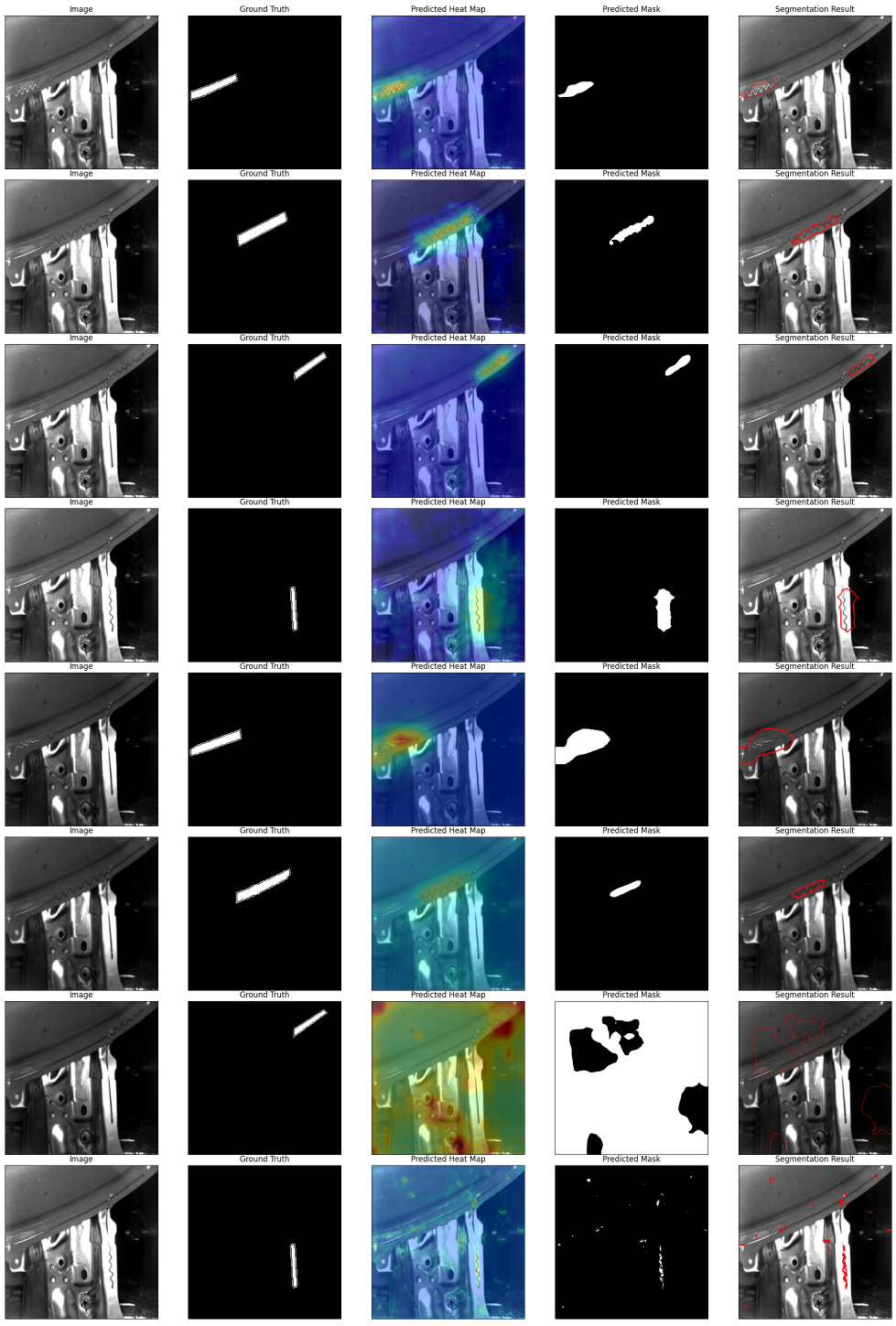


Figure D.17: Part6 Segmentation Result, from the top to bottom: Uflow, Fastflow, Reverse_Dis.,Cflow, DFM, CFA, CSFlow, DRAEM.

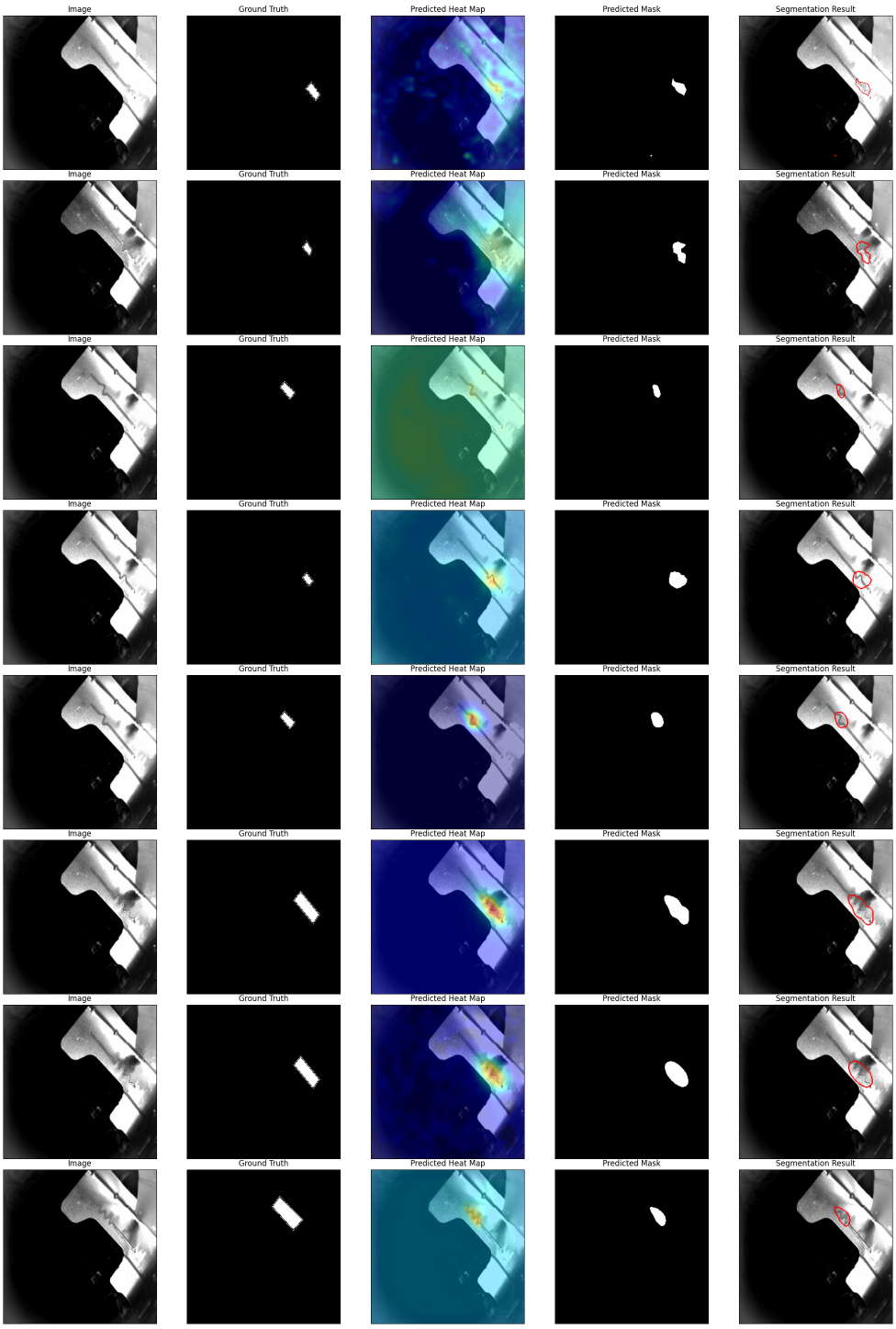


Figure D.18: Optimal Part 4 Segmentation Result, from the top to bottom: Fastflow, CFlow, stfpm, UFlow, Reverse_Distill, PaDiM, Patchcore, Efficient_AD