Proceedings of the 35th European Safety and Reliability & the 33rd Society for Risk Analysis Europe Conference
Edited by Eirik Bjorheim Abrahamsen, Terje Aven, Frederic Bouder, Roger Flage, Marja Ylönen
 Published by Research Publishing, Singapore.
doi: 10.3850/978-981-94-3281-3_ESREL-SRA-E2025-P5342-cd

Traditional vs. AI-based Methods for Detection of Anomalies on Metal Surfaces

Jonas Strohhofer
Munich University of Applied Sciences, Munich, Germany. E-mail: jonas.strohhofer@hm.edu

Marcin Hinz
Munich University of Applied Sciences, Munich, Germany. E-mail: marcin.hinz@hm.edu

Anomalies are data that deviate significantly from expected patterns, often indicating defects or irregularities. Detecting such anomalies is especially important in manufacturing, where visual anomaly detection (VAD) based on image data plays an essential role in quality control.
By automating defect detection, VAD greatly reduces the time and costs associated with manual inspections. This has led to extensive research and the development of various approaches. While modern methods rely on deep learning (DL) techniques, earlier approaches are based on simpler statistical analyses.
This paper examines whether deep learning is necessary for detecting simple defects, such as scratches on metal surfaces. The results show that while AI-based methods achieve near-perfect detection accuracy, traditional methods using simple statistical features can still reach up to 89% accuracy. Additionally, these traditional approaches are far more efficient, requiring only a fraction of the inference time. This highlights their potential as a lightweight and effective solution, particularly in real-time or resource-constrained scenarios.

Keywords: Anomaly Detection, Surface Defects, Quality Control, Image Processing, Benchmarking.

1. Introduction

Anomaly detection (AD) is an important topic in various fields, which enables the automation of processes that traditionally relies on manual human inspection. Automated systems for AD do not only improve efficiency but also reduce errors in diverse applications ranging from industrial quality control to medical diagnostics. Anomalies can be categorized into logical anomalies, such as incorrect packaging counts, and structural anomalies, which involve deviations in the physical or visual structure of objects. This paper focuses exclusively on structural anomalies.

Despite its potential, AD poses significant challenges due to the highly variable nature of anomalies. Modern state-of-the-art methods have demonstrated remarkable performance, achieving high detection rates on benchmark datasets. These advanced approaches use automated feature extraction based on deep learning, which requires substantial computational resources.

In contrast, traditional methods rely on handcrafted features. While less sophisticated, we expect them to be both efficient and effective in situations where the analyzed object displays a uniform appearance, such as metallic or plastic surfaces.

This paper presents a comparative study of traditional and modern methods for VAD which is a notable gap in literature. The comparison highlights the trade-offs between the simplicity of traditional approaches and the accuracy offered by advanced techniques. The analysis is based on images of metallic surfaces with and without scratches.

2. Visual Anomaly Detection

VAD can be divided into three distinct subtasks, as illustrated in Figure 1. Anomaly *detection*, anomaly *classification*, and anomaly *segmentation*. Anomaly detection identifies whether an image contains any irregularities. Anomaly classification categorizes the detected irregularities into predefined classes. Finally, anomaly segmentation precisely localizes the anomalies within the image. But those terms are not uniformly used in literature. Some authors group the task referred to here as detection within classification, treating detection as a special case of segmenta-

tion (Prunella et al. (2023)). In general, not all approaches address these three sub-tasks independently. Some methods combine detection and classification into one step (Klarák et al. (2024); Niu et al. (2020)). Other approaches first divide the image into patches and then use the classification results of these patches to perform segmentation (Roth et al. (2021)). The segmentation of scratches is often described as a very hard task as they are often very subtle and have low contrast to the background of the image (Cao et al. (2021); Ho et al. (2022)).

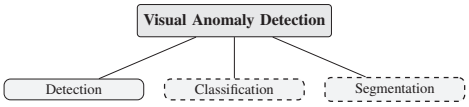

Fig. 1. The three different subtasks of VAD. This paper analyses just the detection ability of the different methods.

In this paper, only the detection ability of the different VAD approaches is analyzed. All algorithms are trained on the same input image data to learn the characteristics of scratch-free metal surfaces from the training set. Each approach then assigns an anomaly score to every image in the test set, which is used to classify the image as either anomalous or scratch-free.

2.1. *Learning paradigms for VAD*

VAD methods can be categorized into supervised and unsupervised approaches. There are also intermediate approaches which aim to balance the strengths and limitations of both techniques, but they are not as commonly used.

Supervised approaches require labeled datasets containing both normal and anomalous images. These methods rely on the explicit identification of anomalies during training, often resulting in high detection accuracy for known defect types. However, they face two major challenges: first, the model needs to have a strong ability to generalize to be able to detect unseen anomalies. Second, getting enough representative and also labeled data is difficult and costly, which limits their applicability in real-world scenarios (Yang et al. (2022)).

In contrast, unsupervised methods need only non-anomalous images, as they learn the normal appearance of the object. Anomalies are then identified as deviations from this learned representation. This approach is advantageous in scenarios where obtaining labeled anomalies is difficult or impractical. Unsupervised VAD also enables to detect new beforehand unseen anomalies (Bergmann et al. (2021)).

Intermediate methods, such as semi-supervised techniques, address these challenges by for example making us of partially labeled datasets. While promising, these approaches are less commonly applied in VAD due to their reliance on specific problem settings and additional implementation complexity.

Acquiring sufficient and representative data remains a fundamental challenge across all VAD approaches, as anomalies are inherently rare and diverse (Prunella et al. (2023)). Consequently unsupervised VAD methods are widely used in both industrial applications and research settings; accordingly, this paper focuses on unsupervised approaches.

2.2. *Traditional Methods*

In this study, ”traditional” methods refer to VAD approaches that rely on statistical measures extracted from an image. These methods are computationally efficient, as they use straightforward mathematical formulations. To provide a diverse perspective, the selected methods were divided into two categories: descriptive features, which directly represent raw image properties, and computational features, which involve simple mathematical calculations. Overall, those measures where not inherently designed for VAD but are explored here for this purpose

The descriptive features include *Pixel Intensity* and *Histogram Comparison*. The computational features, represented by the *Structural Similarity Index (SSIM)* and *Gray-Level Co-occurrence Matrix Entropy (GCME)*.

Pixel intensities refer to the numerical values that represent the brightness or color intensity of a pixel in a digital image (Gonzalez (2007)). It is used to calculate the mean intensity value for an

image patch. Deviations from the reference mean, which are based on the provided training images, indicate anomalies.

Histogram Comparison involves comparing the intensity distribution of an image patch to a reference histogram obtained from training images. The histogram represents the frequency of pixel intensity values within the image patch (Burger and Burge (2013)). Deviations between the patch histogram and the reference histogram are measured using the chi-square distance (Pele and Werman (2010)). Patches with large distances are classified as anomalies. The chi-square distance between two histograms H_1 and H_2 is calculated as:

$$\chi^2(H_1, H_2) = \sum_i \frac{(H_1[i] - H_2[i])^2}{H_1[i] + H_2[i]} \quad (1)$$

The index i in Eq. (1) refers to a specific bin in the histograms H_1 and H_2. Each bin corresponds to a range of pixel intensity values. The summation iterates over all bins in the histograms to compute the overall chi-square distance.

The SSIM was introduced by Wang et al. (2004). It quantifies image similarities by analyzing luminance, contrast, and structural properties. For each image patch, the similarity to a reference patch is measured. Anomalies are indicated by patches with low SSIM scores. The general formula for SSIM is provided in Eq. (2).

$$\text{SSIM}(x, y) = l(x, y) \cdot c(x, y) \cdot s(x, y) \quad (2)$$

The components $l(x, y)$, $c(x, y)$, and $s(x, y)$ correspond to luminance (Eq. (3)), contrast (Eq. (4)), and structural similarity (Eq. (5)), respectively.

$$l(x, y) = \frac{2\mu_x\mu_y + C}{\mu_x^2 + \mu_y^2 + C} \quad (3)$$

$$c(x, y) = \frac{2\sigma_x\sigma_y + C}{\sigma_x^2 + \sigma_y^2 + C} \quad (4)$$

$$s(x, y) = \frac{\sigma_{xy} + C}{\sigma_x\sigma_y + C} \quad (5)$$

These components are calculated based on the mean intensities (μ_x and μ_y), standard deviations (σ_x and σ_y), and covariance (σ_{xy}) between the image patches x and y. The constant C is used to avoid division by zero. The SSIM provides a comprehensive measure of image similarity.

The GCME is a statistical measure introduced by Haralick et al. (1973) to describe the texture of an image based on the Gray-Level Co-occurrence Matrix (GLCM) which is a measurement for analyzing how pixel intensities are distributed in an image. Asha et al. (2011) demonstrate its usefulness for detecting defects.

The GLCM is a matrix that records how often pairs of pixel intensities occur next to each other in a specific spatial relationship. For example, the GLCM can capture how often a pixel with intensity i is found to the right, above, or diagonally from a pixel with intensity j. The GCME uses the GLCM to calculate the randomness, or "entropy" in the texture of the image (see Eq. 6) (Haralick et al. (1973)).

$$\text{GCME} = -\sum_{i=1}^{N}\sum_{j=1}^{N} P(i, j) \log P(i, j) \quad (6)$$

$P(i, j)$ represents the normalized value from the GLCM, which indicates the probability of pixel pairs with intensities i and j occurring in the defined spatial relationship. The parameter N refers to the number of gray levels in the image, determining the size of the GLCM.

The GLCM captures the structural arrangement of textures, while the GCME summarizes this structure in a single value. High GCME values indicate complex, random textures, while low GCME values point to simple, uniform patterns.

In practice, anomalies are detected by comparing the GCME of test patches with that of reference patches. Large deviations in GCME values suggest irregularities, such as scratches or material inconsistencies, making this approach particularly effective for analyzing textured surfaces.

The algorithms for each approach are programmed with a focus on simplicity. While their current implementation can be optimized for efficiency, their purpose is to demonstrate the general utility of simple image features for VAD. The use of patches instead of calculating one value for the whole image improves the sensitivity of the methods, allowing them to capture subtle devia-

tions in texture, intensity, or structural properties that might otherwise be missed. This patch-based approach is not intended for precise spatial localization of anomalies but rather enhances the algorithms' overall ability to detect deviations from the normal reference.

2.3. *AI-based Method*

There has been significant progress in the development of AI-based VAD methods. However, challenges remain, including the lack of standardized benchmarks (Zhang et al. (2024)) and the trade-off between detection accuracy and inference time (Prunella et al. (2023)).

A common practice is to use feature extraction based on pretrained convolutional neural networks (CNN), such as ResNet, originally trained for image classification tasks. To effectively detect both small and large defects, multi-scale features extracted from different layers of these networks are often utilized (Yang et al. (2022)).

This paper focuses primarily on traditional methods; therefore, only one AI-based algorithm was selected for settings a baseline. PatchCore, introduced by Roth et al. (2021), is chosen because it achieves state-of-the-art detection accuracy on many datasets and stands out for its efficiency. Unlike many other deep learning methods, PatchCore does not require extensive training. Instead, it extracts features directly from a pretrained CNN and uses these to identify anomalies, which leads to very low trainings times compared to other AI-based VAD approaches.

3. Dataset

The dataset used in this study was intentionally kept simple, as this is the first comparison being conducted. Starting with a straightforward dataset allowed for a clear and controlled evaluation of the methods before moving on to more complex cases in future work. To achieve this, a custom dataset was created, building on earlier work by Hinz et al. (2019). In their study, knife surfaces were captured under consistent boundary conditions. The images were then uniformly cropped to ensure standardized input for further analysis.

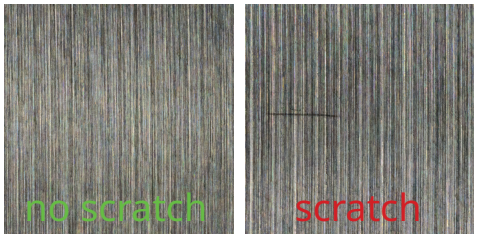

Fig. 2. Comparison of an image of the dataset with and without a scratch

Building upon these images, a representative dataset for VAD was created. The dataset consists of 250 images used for training and an additional 150 images reserved for testing. The test set includes 50 images of scratch-free metallic surfaces and 100 images displaying surfaces with scratches, all with a resolution of 2000x2000 pixels (compare Fig. 2). The size of the scratches varies, ranging from small ones that cover approximately a quarter of the image to larger ones that extend nearly across the entire image. Although the lighting conditions were consistent across all images, the overall coloration of the images is not entirely uniform, with slight variations depending on the specific curvature of the blade section being photographed. Hence, we are faced with the challenge that not every dark spot can simply be classified as a scratch. This requires a robust generalization of the models to distinguish actual anomalies from natural variations in surface appearance which also a much more realistic scenario. But it is important to note that in our data, the surfaces were only very slightly curved.

The size of this dataset aligns closely with the well-established MVTecAD dataset from Bergmann et al. (2021), which on average contains 242 images for training, 31 defect-free test images, and 84 anomalous test images.

As mentioned before, one of the principal challenges in VAD is the collection of enough representative data for training and evaluation. That is why the dataset analyzed in this study was designed to be comparable in structure to often-cited datasets.

4. Metrics

One commonly used metric for evaluating the performance of VAD models is the Area Under the Receiver Operating Characteristic Curve (AUROC). This metric assesses the trade-off between the true positive rate (TPR) and the false positive rate (FPR) across varying decision thresholds based on the anomaly-score of each image.

The true positive rate (TPR), also referred to as sensitivity, is defined in Eq. 7:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{7}$$

where *TP* denotes the number of true positives and *FN* denotes the number of false negatives.

The false positive rate (FPR) is calculated in Eq. 8:

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \tag{8}$$

where *FP* represents the number of false positives and *TN* represents the number of true negatives.

The Receiver Operating Characteristic (ROC) curve plots *TPR* against *FPR* at various threshold values. The AUROC is defined as the area under this curve (see. Eq. 9).

$$\mathrm{AUROC} = \int_0^1 \mathrm{TPR}(\mathrm{FPR})\, d(\mathrm{FPR}) \tag{9}$$

An AUROC value of 1 indicates perfect detection performance, while a value of 0.5 corresponds to random guessing. In contrast, a value of 0 means systematic misclassification, where anomalous samples are consistently classified as normal, and normal samples as anomalous. Values between 0 and 1 represent varying levels of detection performance, with higher values indicating better discrimination between anomalous and normal samples. Due to its ability to assess performance across all thresholds, AUROC is a reliable metric for benchmarking VAD models (Bergmann et al. (2021)).

Additionally, the training and inference time needs to be evaluated as well. While training time is less critical for practical applications, inference time plays an important role, especially in real-time or resource-constrained scenarios where quick and efficient decision-making is essential.

5. Methodology

The previously described dataset of metallic surfaces is analyzed using an unsupervised approach. Each model is trained exclusively on scratch-free images to learn the normal appearance of the surface. During testing, the model is presented with a test set which contains non-anomalous images and images with scratches.

Using the difference between the learned normal appearance and the image being analyzed, the model assigns an anomaly score ranging from 0 to 1. These anomaly scores are used to compute the AUROC value.

Two key comparisons are conducted: the AUROC for VAD and the inference time of each model. These evaluations aim to assess both the detection accuracy and the computational efficiency of the approaches.

6. Results

In figure 3 the ROC curves for all models are shown, illustrating their ability to distinguish images with or without scratches. The area under each curve indicates the overall classification performance, while the curves themselves show the trade-off between true positive and false positive rates across different decision thresholds. These thresholds are based on the anomaly scores of the images.

PatchCore achieves a nearly perfect result, with its curve approaching the top-left corner. SSIM shows limited performance, with its curve close to the diagonal which represents random guessing. GCME demonstrates moderate performance, with its curve lying above the diagonal but flattening in the second half. This flattening may indicate that the model performs well on images with clear and pronounced scratches but struggles to identify subtle or small defects. Images with minor scratches may have anomaly scores that are not distinct enough from non-scratched images, making them harder to classify correctly at higher thresholds. Histogram comparison and pixel average perform similarly well, with curves closer to the top-left corner.

In addition to detection accuracy, computational efficiency was analyzed by comparing both

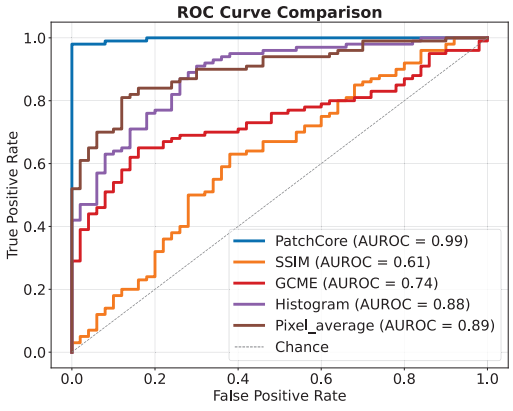

Fig. 3. The anomaly detection performance can be visualized using ROC curves. The AUROC values indicate the overall detection accuracy, with PatchCore achieving the highest AUROC. Traditional methods such as Pixel Average and Histogram Comparison also perform well. GCME shows potential with moderate accuracy but is still not reliable for this task. SSIM demonstrates poor overall accuracy, barely better than random guessing

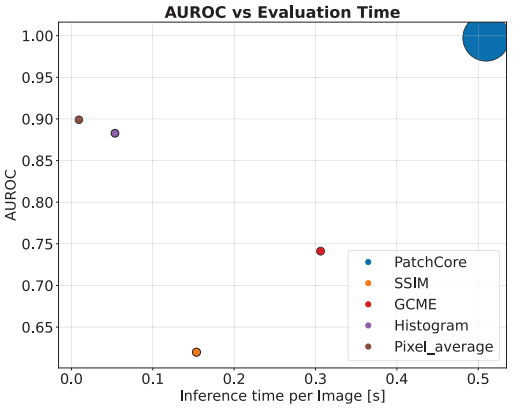

Fig. 4. Comparison of AUROC values and inference times per image for different models. The circle sizes represent the training time required for each model. PatchCore achieves the highest AUROC but with the longest training and inference times. Pixel Average and Histogram Comparison provide a balance between high AUROC and computational efficiency. SSIM and GCME show low accuracy with short computational times

training and inference times, as visualized in Figure 4. The AUROC value is plotted against inference time per image, with circle sizes representing the training time required for each model. For the PatchCore model, input images had to be resized to 1000x1000 pixels due to memory limitations, as higher resolutions could not be processed.

PatchCore achieves the highest AUROC but requires the longest inference and training times, despite analysing smaller images, as reflected by its position and large circle size.

The Pixel average method achieves high performance with very short inference times and minimal training effort, making it computationally very efficient. The Histogram approach performs similarly but requires slightly more inference time. GCME, while showing moderate classification performance, has relatively high inference time for a traditional method, reflecting its computational intensity.

SSIM is computationally efficient with short inference times, but its poor performance makes the speed less meaningful for practical applications.

Table 1 provides a detailed comparison of the AUROC, training time, and inference time for all models. PatchCore, the only AI-based approach, achieved the highest AUROC of 0.99, demonstrating excellent detection accuracy. However, it required significantly longer training (103.93 seconds) and inference times (0.51 seconds) compared to the traditional methods, highlighting its computational demands.

Among the traditional methods, Pixel Average and Histogram Comparison performed particularly well. Pixel Average achieved an AUROC of 0.89 and excelled in computational efficiency, requiring only 2.12 seconds for training and 0.01 seconds for inference. Histogram Comparison had a slightly lower AUROC of 0.88 but was still very efficient, with training and inference times of 2.15 seconds and 0.05 seconds.

In contrast, SSIM and GCME showed lower AUROC values of 0.61 and 0.74, respectively, suggesting they are less suited for detecting scratches on metallic surfaces. Despite this, they demonstrated relatively short training and infer-

Table 1. Comparison of model performance in terms of AUROC, training time, and inference time.

Model	AUROC	Training	Inference
Patchcore	0.99	103.93 s	0.51 s
SSIM	0.61	2.44 s	0.15 s
GCME	0.74	2.39 s	0.31 s
Histogram	0.88	2.15 s	0.05 s
Pixel Average	0.89	2.12 s	0.01 s

ence times. SSIM required 2.44 seconds for training and 0.15 seconds per image for inference, while GCME required 2.39 seconds for training and 0.31 seconds per image for inference.

These findings highlight the trade-offs between accuracy and computational efficiency. While AI-based methods like PatchCore achieve the highest accuracy with an AUROC of 0.99, they come with significantly longer training (103.93 seconds) and inference times (0.51 seconds). In contrast, traditional methods such as Pixel Average and Histogram Comparison strike a good balance of detection performance and speed, making them viable options for real-time or resource-constrained applications.

7. Summary and Outlook

This study underscores the value of traditional methods for visual anomaly detection, highlighting that statistical features should not be underestimated. Despite their simplicity, these methods can achieve strong predictive performance, with significantly lower computational demands. While AI-based approaches achieve higher detection accuracy, this comes at the cost of substantially increased training and inference times, as well as greater memory requirements. Although AI algorithms are expected to become increasingly optimized through continued advances in hardware, model architectures, and training techniques, such improvements may gradually narrow the current gap in computational efficiency.

One key limitation observed with PatchCore, the AI-based method, is its high memory demand. In contrast, traditional statistical methods, due to their simplicity, are far less restricted by such constraints, making them more flexible in scenarios with high-resolution image data or limited hardware resources.

Although traditional methods are mathematically straightforward, their evaluation offers significant flexibility. For example, the performance of GCME and SSIM could potentially be improved by exploring alternative parameter configurations or applying different comparison techniques. This highlights the importance of optimizing these methods, as their effectiveness can vary depending on the chosen parameters and evaluation strategies.

The analysis in this study is conducted on simple structural anomalies. It is unclear how well these findings translate to more complex datasets with greater variability in defect types or more challenging anomalies. Additionally, this work only examined the detection of anomalies and did not explore classification or segmentation, which are also important in real-world tasks.

Future research should further explore the potential of traditional methods by testing a wider range of statistical features and algorithms. Hybrid approaches, combining traditional methods with AI-based techniques, are a promising direction as they could merge the efficiency of traditional methods with the accuracy and flexibility of AI models.

References

Asha, V., N. U. Bhajantri, and P. Nagabhushan (2011). Glcm-based chi-square histogram distance for automatic detection of defects on patterned textures. *International Journal of Computational Vision and Robotics 2*(4), 302.

Bergmann, P., K. Batzner, M. Fauser, D. Sattlegger, and C. Steger (2021). The mvtec anomaly detection dataset: A comprehensive real-world dataset for unsupervised anomaly detection. *International Journal of Computer Vision 129*(4), 1038–1059.

Burger, W. and M. J. Burge (2013). *Principles of Digital Image Processing*. London: Springer London.

Cao, J., G. Yang, and X. Yang (2021). A pixel-level segmentation convolutional neural net-

work based on deep feature fusion for surface defect detection. *IEEE Transactions on Instrumentation and Measurement 70*, 1–12.

Gonzalez, R. C. (2007). *Digital image processing* (3rd ed. ed.). Reading, Massachusetts and Upper Saddle River, NJ: Addison-Wesley Publishing Company and Pearson Prentice Hall.

Haralick, R. M., K. Shanmugam, and I. Dinstein (1973). Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics SMC-3*(6), 610–621.

Hinz, M., M. Radetzky, L. Hannah Guenther, P. Fiur, and S. Bracke (2019). Machine learning driven image analysis of fine grinded knife blade surface topographies. *Procedia Manufacturing 39*, 1817–1826.

Ho, C.-C., M. A. Benalcazar Hernandez, Y.-F. Chen, C.-J. Lin, and C.-S. Chen (2022). Deep residual neural network-based defect detection on complex backgrounds. *IEEE Transactions on Instrumentation and Measurement 71*, 1–10.

Klarák, J., R. Andok, P. Malík, I. Kuric, M. Ritomský, I. Klačková, and H.-Y. Tsai (2024). From anomaly detection to defect classification. *Sensors (Basel, Switzerland) 24*(2).

Niu, S., B. Li, X. Wang, and H. Lin (2020). Defect image sample generation with gan for improving defect recognition. *IEEE Transactions on Automation Science and Engineering*, 1–12.

Pele, O. and M. Werman (2010). The quadratic-chi histogram distance family. In K. Daniilidis, P. Maragos, and N. Paragios (Eds.), *Computer Vision – ECCV 2010*, Berlin, Heidelberg, pp. 749–762. Springer Berlin Heidelberg.

Prunella, M., R. M. Scardigno, D. Buongiorno, A. Brunetti, N. Longo, R. Carli, M. Dotoli, and V. Bevilacqua (2023). Deep learning for automatic vision-based recognition of industrial surface defects: A survey. *IEEE Access 11*, 43370–43423.

Roth, K., L. Pemula, J. Zepeda, B. Schölkopf, T. Brox, and P. Gehler (2021). Towards total recall in industrial anomaly detection.

Wang, Z., A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society 13*(4), 600–612.

Yang, J., R. Xu, Z. Qi, and Y. Shi (2022). Visual anomaly detection for images: A systematic survey. *Procedia Computer Science 199*, 471–478.

Zhang, J., H. He, Z. Gan, Q. He, Y. Cai, Z. Xue, Y. Wang, C. Wang, L. Xie, and Y. Liu (2024). A comprehensive library for benchmarking multi-class visual anomaly detection.