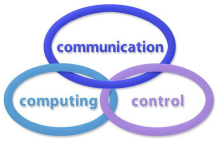
INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 21, Issue: 6, Month: December, Year: 2026
Article Number: 7591, https://doi.org/10.15837/ijccc.2026.6.7591



CCC Publications



Green AI-Driven Synthetic Data Generation for Improved Industrial Metal Surface Defect Detection

S. Mătase, R. Andonie

Stelian Mătase
Department of Electronics and Computers
Transilvania University of Braşov, Romania
*Corresponding author: stelian.matase@unitbv.ro

Răzvan Andonie
1. Computer Science Department
Central Washington University, Ellensburg, WA, USA
2. Department of Electronics and Computers
Transilvania University of Braşov, Romania
razvan.andonie@cwu.edu

Abstract

Defect detection of industrial metal surfaces is often limited by the scarcity of annotated data and the high cost of manual labeling, which limits the performance of data-driven inspection systems. This paper presents a Green AI-driven framework for synthetic data generation that efficiently augments industrial defect datasets while minimizing computational and annotation costs. The proposed approach employs a deterministic procedural pipeline to generate synthetic defect images together with aligned segmentation masks at the pixel-level, providing accurate supervision without the computational overhead of deep generative models. The framework is evaluated on the Kolektor Surface-Defect Dataset 2 (KSDD2), which is extended into a hybrid dataset through synthetic augmentation followed by human-in-the-loop validation, to ensure the realism and reliability of the generated samples and annotations. A lightweight state-of-the-art anomaly detection model, SuperSimpleNet, is trained under controlled ablation settings on both the original and hybrid datasets using identical test splits for fair comparison.

Performance is evaluated using both image-level and pixel-level metrics, including AUROC and average precision. Experimental results demonstrate that the proposed hybrid dataset consistently improves defect detection and localization performance while substantially reducing the need for costly data acquisition and manual annotation. Unlike computationally intensive generative approaches based on GANs or diffusion models, the proposed procedural framework synthesizes defects directly within predefined regions, enabling automatic generation of pixel-accurate ground-truth annotations without human intervention. The resulting hybrid dataset improves all evaluation metrics, achieving the largest gain in anomaly localization performance, with AP-loc increasing by 5.3%, while image-level AUROC increases from 0.9840 to 0.9907. By combining efficient synthetic data generation with precise automatic annotation, the proposed approach provides a resource-efficient solution that aligns with Green AI principles.

Keywords: metal surface defect detection, synthetic data generation, generative AI, green AI, human-in-the-loop

1 Introduction

Current AI systems are based on large and high-quality datasets [1]. However, in many real-world applications, data acquisition remains a major challenge due to limited data availability and the cost and complexity of manual annotation. This limitation is particularly evident in industrial inspection scenarios, where defect samples are inherently rare and labeling requires domain expertise. To address these challenges, synthetic data generation (SDG) has emerged as a viable approach to augment training datasets and improve model generalization. SDG encompasses multiple approaches, including *procedural methods* (using computer graphics and statistical models) and *generative AI* techniques (learning-based approaches such as VAEs, GANs, and diffusion models).

Generative AI techniques aim to learn the underlying distribution of input data and produce new samples that preserve the structural, statistical, and semantic properties of the original dataset. Recent advances in generative modeling have demonstrated remarkable capabilities in producing high-quality synthetic data across various domains [2, 3]. These approaches have been successfully applied in areas such as computer vision, robotics, and autonomous systems, where large-scale annotated datasets are essential. However, despite their effectiveness, many of these methods are computationally intensive, require complex training procedures, and often lack explicit control over the generated content, particularly in terms of precise annotation alignment.

The growing computational demands of these models have also raised concerns about their energy consumption and environmental impact. Consequently, the concept of Green AI has gained increasing attention, emphasizing the need to evaluate AI systems not only by their predictive performance but also by their computational efficiency and sustainability. From a Green AI perspective, computationally intensive generative models raise concerns regarding scalability, sustainability, and industrial deployment [4]. In manufacturing and inspection systems, practical solutions must be efficient, easy to deploy, and computationally lightweight.

These requirements motivate the development of SDG approaches that provide greater efficiency and controllability. Unlike generative AI, procedural SDG methods generate data using computer graphics and statistical models, allowing explicit control over scene composition, annotation generation, and precise pixel-level ground truth. Conventional dataset acquisition methods, including crowd-sourcing, web scraping, and manual annotation, are often impractical in industrial environments due to their high cost, limited scalability, and inconsistent annotation quality. Consequently, procedural SDG enables the systematic generation of accurately labeled datasets, providing a scalable and efficient alternative to develop data-driven models [5, 6].

A key challenge in procedural SDG is the generation of reliable ground-truth annotations. Many existing methods rely on auxiliary models for labeling, introducing additional sources of error, and reducing the reliability of the dataset. Achieving accurate pixel-level correspondence between synthetic images and their annotations remains particularly difficult in defect detection applications. Human-In-The-Loop (HITL) validation can help address this limitation by enabling the verification and refinement of generated samples and their corresponding annotations, thus improving the fidelity and trustworthiness of the resulting dataset [7, 8].

Motivated by these considerations, this work presents a resource-efficient procedural SDG framework aligned with the principles of Green AI for the detection of industrial metal surface defects. Our main contribution is a deterministic pipeline that generates synthetic defect images with pixel-accurate segmentation masks, without relying on computationally intensive deep generative models such as GANs or diffusion models. By emphasizing deterministic generation, precise annotation control, and low computational overhead, the proposed approach enables the creation of high-quality hybrid datasets. A controlled experimental evaluation demonstrates that synthetic data augmentation improves both defect classification and localization performance, highlighting the potential of energy-efficient, data-centric strategies for industrial AI applications.

The remainder of the paper is structured as follows. Section 2 introduces the concept of Green AI, with particular emphasis on its relationship to synthetic data generation as a resource-efficient strategy in modern AI development. Section 3 presents the data set considered in this study, reviews recent and relevant data generation methods, describes the state-of-the-art classifier for the detection of metal surface defects and describes the evaluation framework used for the ablation study. Section 4

details the proposed generative methodology, including the rationale behind its selection from a Green AI perspective, and describes the SuperSimpleNet model employed in this work. Section 5 presents the results of the ablation study. Finally, Section 6 concludes the paper and summarizes the main findings.

2 Towards Sustainable Generative AI: Existing Approaches

This section reviews existing efforts to improve the energy efficiency and environmental sustainability of generative AI and identifies key gaps that motivate future research.

AI is fundamentally transforming the energy industry, enabling more efficient, sustainable, and secure systems. At the same time, AI itself has become a major consumer of computational resources, raising important questions about its own sustainability. A major concern is the energy required to train and deploy large-scale deep learning models. According to René Haas, CEO of Arm, AI data centers could consume as much as one quarter of all U.S. electricity by the end of the decade.[1] AI servers consume substantially more power than conventional servers because they rely on high-performance GPUs and specialized AI accelerators. Cooling further contributes significantly to energy demand, as these high-density systems generate enormous amounts of heat. Although advances in liquid cooling have improved efficiency, cooling still accounts for a substantial fraction of total data-center energy consumption. In addition, training foundation models requires millions of GPU-hours and large amounts of electricity. After deployment, inference often becomes the dominant source of energy consumption, as serving billions of user requests over a model's lifetime can exceed the energy required for training. The widespread adoption of generative AI further amplifies these trends.

Sustainable development has become a global priority, yet the ability to sustainably expand digital infrastructure is struggling to keep pace with the rapid growth of AI data centers. This imbalance is driven by the prevailing scaling paradigm, in which advances in generative AI are achieved primarily through larger training datasets, more model parameters, and increased computational resources. Although this strategy has produced remarkable improvements in model capability, it is approaching practical, economic, and environmental limits. Despite ongoing efforts to improve AI efficiency, the environmental footprint of generative AI—including greenhouse gas emissions, electricity demand, and water consumption—remains substantial [9]. Consequently, future progress will increasingly depend on algorithmic and architectural innovations that enhance computational efficiency rather than relying solely on larger models and greater computing power [10].

The Green AI paradigm [11] advocates the development of computationally efficient algorithms, energy-aware hardware, and optimized data processing pipelines. It also emphasizes transparency by reporting resource-related metrics—including computational cost, energy consumption, and carbon footprint—alongside traditional performance measures, thus promoting AI systems that are both effective and environmentally sustainable.

The growing importance of Green AI has stimulated extensive research in this area [12]. Zhou *et al.* [13] provide a comprehensive survey that formalizes Green AI around four key components: measures of greenness, energy-efficient AI methodologies, energy-efficient computing infrastructures, and AI applications for sustainability. Complementing this perspective, Clemm *et al.* [14] examine environmental assessment and ecodesign practices, proposing a life-cycle framework that models AI systems through four core elements: data, model, server, and cloud infrastructure. Their analysis highlights the substantial carbon footprint of modern computing platforms, highlighting the need for energy-aware design throughout the life cycle of the AI system.

From an industrial perspective, the adoption of Green AI is further motivated by practical considerations such as deployment speed, operational costs, and system maintainability. In many real-world applications, simpler and more computationally efficient models are preferred because they enable faster deployment, lower energy consumption, and reduced maintenance costs. Moreover, their greater interpretability enhances explainability and reliability, both of which are essential for industrial adoption, particularly in safety-critical and highly regulated domains.

[1] The Wall Street Journal, April 9, 2024.

In the context of generative AI, the work of Asperti *et al.* [15] represents one of the first efforts to analyze the energy efficiency of generative models. Focusing on variational autoencoders, the authors propose optimization techniques to reduce the computational cost of synthetic data generation. From a Green AI perspective, energy consumption in synthetic data generation (SDG) affects the AI development pipeline in two fundamental ways: (i) energy is directly consumed during the generation process and (ii) the quality of the generated data influences downstream stages, particularly model training and evaluation, thereby affecting the overall computational and energy requirements. Consequently, the choice of generative models, architectures, and hyperparameters plays a critical role in optimizing resource utilization.

While recent studies have explored GAN- and diffusion-based synthetic data generation for industrial applications, these approaches often require substantial computational resources. In contrast, this work demonstrates how a lightweight procedural SDG method for defect synthesis can improve the sustainability of data-intensive AI applications. By inserting synthetic defects into predefined regions, the proposed method automatically generates pixel-accurate ground-truth annotations, eliminating manual labeling efforts and reducing dependence on large annotated datasets. Although SDG introduces an initial computational overhead, this investment can be offset by improved efficiency in subsequent stages of the machine learning pipeline. High-quality synthetic datasets can accelerate model convergence, reduce the number of training iterations, and lessen the need for costly data collection and annotation. Consequently, the overall computational and energy costs associated with model development and deployment can be reduced, despite the upfront cost of synthetic data generation. The proposed framework therefore offers a computationally efficient approach that reduces the environmental impact of dataset creation while supporting the development of accurate and resource-efficient machine learning models for industrial inspection, maintaining a high anomaly detection performance.

3 Methods

This section describes the benchmark dataset, the synthetic data generation framework, the state-of-the-art classifier, and the evaluation protocol used in the metal surface defect detection study.

3.1 Dataset

The original dataset used in this study is the Kolektor Surface-Defect Dataset 2 (KSDD2) [16]. It is used in the generative phase of the experiments as a basis for constructing a larger hybrid data set. The dataset comprises approximately 3,300 grayscale images of metal surfaces acquired under real industrial conditions, with a significant imbalance between defective and non-defective samples. Defective instances are annotated with pixel-level ground truth masks, enabling both supervised and semi-supervised learning approaches. The dataset is characterized by small and low-contrast defects, which increase the difficulty of detection and make it suitable for evaluating high-precision inspection methods.

Several established generative approaches can be considered for synthetic defect generation, including variational autoencoders (VAEs) [17, 18], adversarial frameworks such as DefectGAN [19, 20], diffusion-based methods such as DefectFill [21], and large-scale generative models such as Stable Diffusion XL [22] and Diffusion Transformers (DiT) [23].

VAEs offer stable training and efficient latent representations, but tend to produce overly smooth and less realistic anomalies. Adversarial models provide sharper outputs and flexible defect synthesis; however, they are often prone to training instability and may introduce spatial inconsistencies that affect annotation accuracy. Diffusion-based methods achieve high-fidelity, context-aware synthesis, yet they typically incur significant computational cost and require complex sampling procedures. Similarly, large-scale generative models such as SDXL and DiT deliver photorealistic results but require substantial computational resources and additional mechanisms for ground-truth estimation, limiting their suitability for controlled industrial settings. In this context, procedural generation emerges as a viable alternative due to its deterministic nature, precise pixel-level control, and negligible computational overhead, making it particularly attractive for resource-efficient data synthesis scenarios.

3.2 SOTA Defect Classifier

Defect detection methods for industrial surface inspection can be broadly categorized into classical feature-based approaches and deep learning-based techniques. Traditional methods rely on handcrafted features such as texture descriptors, edge information, or statistical measures, combined with conventional classifiers; however, their performance degrades significantly in the presence of subtle, low-contrast, or highly variable defects, and they lack robustness under real industrial conditions. Deep learning-based approaches address these limitations by learning hierarchical representations directly from data and have therefore become the dominant paradigm in modern inspection systems.

While high-capacity frameworks—including U-Net++ [24], DeepLabv3+ [25], Mask2Former [26], and Swin Transformer-based variants [27], alongside traditional autoencoder-based anomaly detectors—achieve strong performance, they incur substantial computational overhead and architectural complexity. This burden frequently compromises their viability for real-time deployment in resource-constrained industrial settings. Moreover, these models often exhibit pronounced sensitivity to dataset bias, necessitating exhaustive hyperparameter tuning to generalize effectively across diverse defect morphologies. Consequently, the paradigm has shifted toward lightweight, efficient architectures designed to minimize environmental and computational footprints. Within this domain, SuperSimpleNet [28] has recently emerged as a state-of-the-art competitive solution for metal surface defect detection, offering a favorable trade-off between accuracy and efficiency. Based on SimpleNet model proposed by Liu *et. al* [29], SuperSimplNet adopts a simplified architecture that enables fast inference, stable optimization, and strong sensitivity to small and subtle defects, which makes it particularly suitable for real-time industrial inspection applications.

3.3 Evaluation Framework

The evaluation framework is designed to quantitatively assess the impact of the proposed synthetic data generation strategy on defect detection performance through an ablation study. The state-of-the-art SuperSimpleNet architecture is evaluated under two experimental conditions: training on the original KSDD2 dataset and training on the augmented hybrid dataset. To ensure a fair and consistent comparison, both experimental setups share an identical test subset, while only the training data differ between the two cases. This controlled split guarantees that performance variations observed in the ablation study are exclusively attributable to the inclusion of synthetic samples. Evaluation is conducted using both image and pixel-level metrics, defined in [30, 31, 32].

Image-level defect detection performance is evaluated using I-AUROC and AP-det. Segmentation performance is assessed using seg-I-AUROC and seg-AP-det, while pixel-level anomaly localization is evaluated using P-AUROC, AUPRO, and AP-loc. Together, these metrics provide complementary measures of image-level detection, segmentation quality, and spatial localization performance

4 Our approach

This section presents the proposed GreenAI-driven data curation pipeline, the construction of the KSDD2sg hybrid dataset, and the controlled benchmarking setup used to evaluate its impact on defect detection performance.

An overview of the proposed evaluation framework is shown in Fig. 4. The architecture is organized into three core phases: a GreenAI-driven data curation stage that passes low-cost procedural generations through a HITL verification step to construct the hybrid KSDD2 dataset; a parallel training stage where separate instances of the state-of-the-art SuperSimpleNet model are deployed; and a unified benchmarking stage. By evaluating both configurations against an identical fixed KSDD2 test split across multiple detection, segmentation, and localization metrics, we systematically isolate the performance gains introduced by our data generation pipeline. A detailed description of the pipeline components is provided further in the current section.

As an initial step in the proposed framework, a stochastic procedural generation mechanism is implemented to simulate high-frequency textural disruptions on clean surface samples, synthesizing realistic industrial anomalies for the KSDD2 training split. The process begins with deterministic

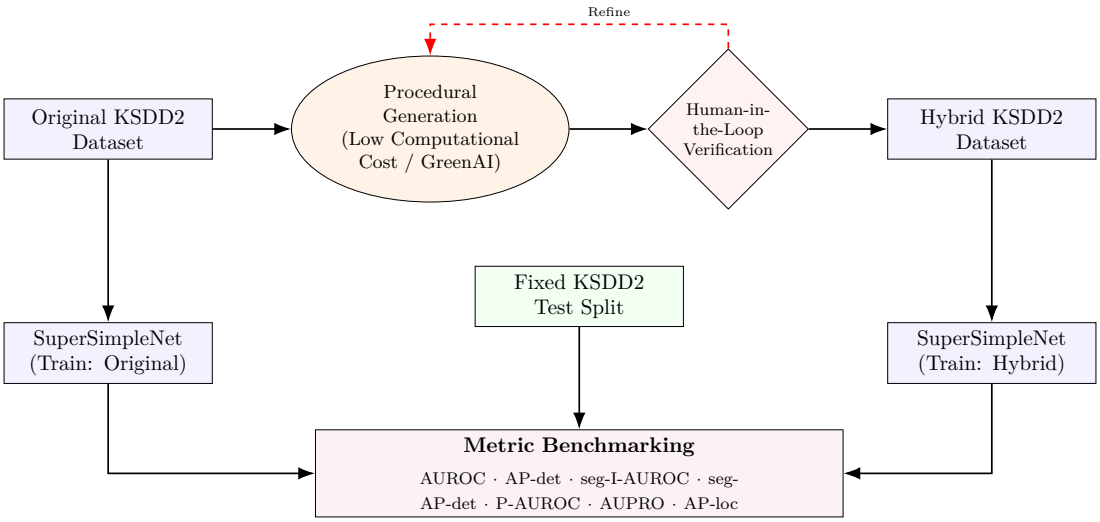


Figure 1: Overview of the proposed framework, illustrating the computationally efficient synthetic data generation, HITL verification, subsequent hybrid dataset construction, parallel network training, and comprehensive metric benchmarking stages.

environment calibration to ensure reproducibility, followed by the generation of defect geometries using randomized overlapping elliptical primitives combined with morphological closing operations. To achieve seamless integration, the generated shapes are subjected to Gaussian edge smoothing and subsequently fused with the original images via alpha blending, defined as $I_{out} = I_{orig} \times (1 - \alpha) + Texture \times \alpha$.

The textura-injection stage dynamically selects among three distinct degradation models—abrasion, textured stain, and structural flaw—to replicate the physical manifestations of real-world manufacturing anomalies, while employing smoothed Gaussian noise to approximate material grain and surface irregularities. Specifically, the textured stain model simulates localized chemical discoloration (Fig. 2a) and specular fluid depositions such as oil contamination (Fig. 2c), characterized by sharp, high-contrast outer boundaries. The structural flaw model synthetically generates macrostructural casting porosities and surface voids (Fig. 2b), manifesting as distinct cavities with well-defined geometries. Similarly, the abrasion model replicates mechanical surface scratches (Fig. 2d), producing sharp linear anomalies that simulate localized material displacement. In parallel with this injection process, deterministic ground-truth masks are generated, with optional boundary smoothing to provide soft labels and ensure precise pixel-level alignment for supervised learning. This automated pipeline enables efficient batch generation of high-fidelity synthetic samples that remain both visually and structurally consistent with authentic industrial defects, thereby providing robust training data for downstream surface defect detection algorithms.

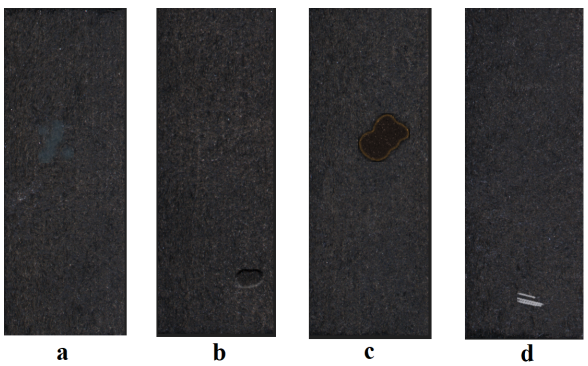


Figure 2: Examples of industrial surface defects used for synthetic model alignment: (a) surface discoloration, (b) casting porosity/void, (c) fluid contamination (oil stain), and (d) mechanical abrasion (scratch).

This approach is preferred over latent generative methods such as variational autoencoders (VAEs), which tend to produce overly smooth anomalies, and adversarial frameworks such as DefectGAN, which may introduce spatial inconsistencies and artifacts affecting label accuracy. Additionally, large-scale generative models, including Stable Diffusion XL (SDXL) and Diffusion Transformers (DiT), while capable of high photorealism, require auxiliary mechanisms for ground-truth estimation. In contrast, the proposed procedural framework ensures fully deterministic mask generation and precise spatial correspondence between defects and annotations. Although recent methods such as DefectFill [21] provide state-of-the-art context-aware inpainting, the proposed approach prioritizes exact pixel-level control and annotation reliability, which are critical for training high-sensitivity detection architectures.

The generative phase is followed by a HITL validation stage, wherein the synthesized samples are systematically reviewed to ensure fidelity and relevance. In this process, defective images and their corresponding masks are carefully examined and only those demonstrating realistic defect characteristics and precise spatial correspondence are retained. This curation step mitigates the inclusion of artifacts or implausible patterns, thereby enhancing the overall quality and reliability of the augmented dataset for subsequent training and evaluation. The resulting curated hybrid dataset is denoted as KSDD2sg, where s refers to the synthetic augmentation process and g highlights its alignment with green, resource-efficient data generation principles.

For our approach, the SuperSimpleNet architecture is selected as the baseline classifier to evaluate the impact of the proposed synthetic data generation pipeline. This choice is motivated by its computational efficiency, stable training behavior, and sensitivity to subtle defect patterns—characteristics that are representative of both the KSDD2 dataset and its hybrid extension KSDD2sg. Using a lightweight and well-controlled model allows the analysis to focus on the contribution of the data itself rather than on the model complexity. The performance of both configurations is evaluated using a comprehensive set of metrics including AUROC, AP-det, seg-I-AUROC, seg-AP-det, P-AUROC, AUPRO, AP-loc, which collectively assess detection, segmentation, and localization capabilities.

5 Experiments

This section presents the creation of the KSDD2SG hybrid dataset using procedural synthetic data generation methods, followed by the experimental evaluation of its impact on industrial anomaly detection performance. The experiments assess both the efficiency of the procedural synthetic data generation framework and its contribution to defect detection and localization accuracy. To isolate the effect of the generated data from architectural factors, the same SuperSimpleNet configuration is trained under identical conditions using both the original KSDD2 dataset and the proposed KSDD2SG dataset. In addition, we analyze the computational cost of the generation process to highlight the resource-efficient characteristics of our approach within the Green AI paradigm.

5.1 Synthetic dataset generation

The first phase of the experimental framework involves the generation of the extended hybrid dataset KSDD2sg, obtained by augmenting the original KSDD2 dataset with newly synthesized defective images and their corresponding defect masks.

Synthetic data are produced using a procedural generation pipeline, in which defects are synthesized within regions defined by the corresponding defects mask images, as described in Section 4. Importantly, the generated defect instances are not restricted to the original dataset annotations; instead, the procedural generation pipeline introduces additional intra-class variability by synthesizing multiple defect modalities—including abrasion, textured stains, and structural flaws as defined in Section 4—thereby extending the effective defect distribution in terms of both appearance diversity and morphological characteristics beyond the original annotations. The generated synthetic images exhibit clearly defined defect regions that correspond to the associated defect masks. The defect texture follows the characteristics of the original surface model, although it does not achieve full photorealism. This represents a deliberate trade-off consistent with the Green AI paradigm, prioritizing computational efficiency over maximal visual fidelity.

Specifically, the generation of 1000 images and their corresponding defect masks requires less than 150 seconds on a standard laptop (Lenovo ThinkPad, Intel i7-8850H CPU, 32 GB RAM, NVIDIA Quadro P2000 GPU). In contrast, a generative adversarial approach such as DefectGAN (trained for 200 epochs) requires approximately 17 hours of computation on a Google Colab system equipped with an NVIDIA A100 GPU to produce a comparable synthetic dataset [33]. Moreover, the localization of defects in the generated defect mask images is less precise for the DefectGAN approach. Expert human interaction is required to select the most realistic defect instances, consistent with a HITL paradigm, which is widely adopted in interactive machine learning/AI systems. For the purposes of this study, 500 synthetic defective images were selected, along with their corresponding defect masks. The authors performed this selection process based on the classical metal surface defect patterns reported in previous work [33], including peer-reviewed journal articles, textbooks, and established industrial experience in the areas of surface inspection and quality control.

5.2 Defect detection and localization

The ablation study is conducted on both the original KSDD2 dataset and the hybrid KSDD2sg dataset, which share an identical test subset composed exclusively of KSDD2 images. The experimental protocol follows the SuperSimpleNet training framework, which employs structured `.pyb` configuration files to define a weakly supervised data splitting strategy. These files specify a constrained learning setting in which only a small subset of training images is associated with pixel-level defect annotations, while the remaining samples are treated as unlabeled or weakly labeled data for semi-supervised learning. This formulation enables the model to learn robust anomaly representations from limited supervision while leveraging additional unlabeled data, thereby reducing annotation overhead and computational resource requirements. Such a design is consistent with the Green AI paradigm, as it minimizes reliance on extensive manual labeling and promotes efficient use of available data without increasing model complexity.

Within this framework, the state-of-the-art SuperSimpleNet model is evaluated on both datasets, with training performed for 30 epochs. All training settings and hyperparameters are kept unchanged; only the dataset composition is modified to isolate the effect of the proposed synthetic augmentation strategy on performance.

Split	Epochs	I-AUROC	AP-det	seg-I-AUROC	seg-AP-det	P-AUROC	AUPRO	AP-loc
53	10	0.9217	0.8278	0.9693	0.9071	0.9496	0.9581	0.6112
53	20	0.9549	0.8746	0.9558	0.9073	0.9455	0.9294	0.6535
53	30	0.9388	0.8643	0.9623	**0.9107**	0.9588	0.9415	0.6911
246	10	0.9793	0.8904	0.9676	0.9012	0.9827	0.9716	0.7158
246	20	0.9590	0.8895	0.9796	0.9036	0.9784	0.9642	0.7118
246	30	**0.9840**	**0.9271**	**0.9845**	0.9097	**0.9847**	**0.9764**	**0.7475**

Table 1: Performance comparison for weakly supervised splits (53 and 246) run on the original KSDD2 dataset, over 10, 20, and 30 epochs on GPU T4x2 Kaggle accelerator.

Two groups of experiments were conducted to evaluate the model under different data conditions. The first group used the original data set and used weakly supervised split files containing 53 and 246 samples. The second group used a synthetic dataset generated from the original data. Specifically, a weakly supervised split of 106 samples was derived from the original 53-sample split, while larger splits containing 492 and 592 samples were generated from the original 246-sample split. This experimental design enables a direct comparison between model performance on the original dataset and on augmented synthetic datasets with increased sample sizes, allowing an assessment of the impact of synthetic data generation on anomaly detection and localization performance. All experiments were trained for a total of 30 epochs, with performance metrics recorded at epochs 10, 20, and 30 to monitor learning progression and evaluate the effect of training duration on model performance.

From a training-time perspective, the duration of one epoch is mainly determined by the number of samples contained in the *.pyb weakly split file, as presented in Table 2. As the number of samples increases, the number of training iterations per epoch also increases, which in turn leads to longer

epoch durations. Minor variations across splits may also arise from differences in data loading efficiency and preprocessing overhead.

Samples in .pyb	53	106	246	492	592
Epoch Time (min:sec)	01:20	02:34	04:51	10:36	14:52

Table 2: Epoch duration as a function of the number of samples in the weakly split file.

5.2.1 Original KSDD2 dataset results

Table 1 presents a comparison of model performance on the original KSDD2 dataset using two weakly supervised splits (53 and 246 samples), evaluated over 10, 20, and 30 training epochs on a GPU T4x2 Kaggle accelerator. Overall, the results show that the larger 246-sample split consistently outperforms the smaller 53-sample split across most evaluation metrics, indicating the benefit of increased training data.

For the 53-sample split, performance improves from 10 to 20 epochs in terms of image-level AUROC (I-AUROC increasing from 0.9217 to 0.9549) and AP-det (from 0.8278 to 0.8746), while localization performance (AP-loc) steadily increases from 0.6112 to 0.6911 across training. However, some metrics, such as AUPRO and segmentation-related scores, fluctuate between epochs, suggesting less stable convergence due to limited data.

In contrast, the 246-sample split achieves stronger and more stable performance overall. At 10 epochs, it has already surpassed the detection and segmentation performance (I-AUROC 0.9793, AP-det 0.8904, AUPRO 0.9716). Although minor variations are observed at 20 epochs, the model achieves its best overall performance at 30 epochs, with the highest reported values across key metrics, including I-AUROC (0.9840), AP-det (0.9271), seg-I-AUROC (0.9845), P-AUROC (0.9847), and AP-loc (0.7475).

Overall, the results indicate that increasing the weakly supervised split size significantly improves model robustness and final performance, while longer training (up to 30 epochs) yields the best results, particularly for the larger 246-sample configuration.

5.2.2 Hybrid KSDD2sg dataset results

Table 3 presents the performance of the model on the synthetic KSDD2sg dataset using weakly supervised splits of 106, 492, and 592 samples, evaluated over 10, 20, and 30 training epochs on a GPU T4x2 Kaggle accelerator. Overall, the results demonstrate a clear positive impact of increasing both the size of the dataset and the duration of training on the performance of the model, particularly for detection and localization tasks.

For the smallest synthetic split (106), performance remains relatively modest and stable across epochs, with only gradual improvements observed in I-AUROC (from 0.9226 to 0.9366) and AP-det (from 0.8237 to 0.8597). However, other metrics such as AP-loc and AUPRO show fluctuations, indicating limited benefit from extended training on this smaller synthetic subset. This suggests that the 106-sample configuration does not provide sufficient diversity for consistent performance gains.

In contrast, the 492-sample split shows substantial improvement in all metrics. At 10 epochs, the model already achieves strong performance (I-AUROC 0.9611, AP-det 0.9046), and this steadily increases with training, reaching its best overall performance at 30 epochs, including I-AUROC of 0.9883, AP-det of 0.9434, and AP-loc of 0.7695. Similar trends are observed in segmentation and region-based metrics, indicating more stable learning behavior and better generalization with increased synthetic data.

The largest split (592) consistently achieves the strongest overall performance across most metrics. After 10 epochs, it already reaches high baseline results (I-AUROC 0.9774, AP-det 0.9230) and continues to improve, peaking at 30 epochs with the best overall scores in the table, including I-AUROC 0.9907, AP-det 0.9454, seg-I-AUROC 0.9909, P-AUROC 0.9938, AUPRO 0.9884, and AP-loc 0.7873. These results indicate that the 592-sample configuration provides the most stable and highest-performing setting, benefiting most from longer training.

In general, Table 3 demonstrates that synthetic data augmentation significantly enhances model performance, with larger synthetic splits consistently yielding better and more stable results, while 30 epochs emerge as the optimal training duration across all configurations.

Split	Epochs	I-AUROC	AP-det	seg-I-AUROC	seg-AP-det	P-AUROC	AUPRO	AP-loc
106	10	0.9226	0.8237	0.9503	0.8885	0.9171	0.8884	0.6164
106	20	0.9329	0.8460	0.9449	0.8772	0.9568	0.9443	0.6073
106	30	0.9366	0.8597	0.9662	0.8755	0.9416	0.9382	0.6286
492	10	0.9611	0.9046	0.9715	0.9036	0.9780	0.9634	0.7067
492	20	0.9787	0.9385	0.9743	0.9113	0.9758	0.9687	0.7249
492	30	0.9883	0.9434	0.9880	0.9232	0.9906	0.9846	0.7695
592	10	0.9774	0.9230	0.9777	0.9218	0.9848	0.9727	0.7290
592	20	0.9792	0.9203	0.9776	0.9061	0.9844	0.9746	0.7181
592	30	**0.9907**	**0.9454**	**0.9909**	**0.9296**	**0.9938**	**0.9884**	**0.7873**

Table 3: Performance comparison for weakly supervised splits (106, 492, 592) run on the synthetic KSDD2sg dataset, over 10, 20, and 30 epochs on GPU T4x2 Kaggle accelerator.

5.2.3 Original versus hybrid dataset results

To conclude the experimental evaluation, an ablation study is conducted based on the best-performing results obtained in the previous experiments. The analysis compares the original KSDD2 configuration defined in the `.pyb` split, consisting of 246 images, against the proposed synthetic augmentation setting, which extends the training distribution with an additional 346 procedurally generated defective samples, resulting in a weakly supervised split of 592 images. This augmentation increases data availability without requiring additional manual annotation, thereby maintaining a low labeling cost. From a Green AI perspective, this approach enhances data diversity and model robustness while preserving computational efficiency and avoiding the need for additional data collection or more complex model architectures.

To ensure a fair comparison, the evaluation considers the best-performing configuration for each setting, corresponding to the optimal epoch identified in the previous experiments. The results are evaluated using the metrics described in Section 3.3, and Table 4 reports the strongest performance achieved by the original 246-sample split and compares it with the synthetic configuration of 592-samples.

Overall, the results show that synthetic augmentation consistently improves all evaluation metrics under optimal training conditions. Image-level performance improves, with AUROC increasing from 0.9840 to 0.9907 and AP-det from 0.9271 to 0.9454. Similarly, segmentation performance benefits from increased data diversity, with Seg-I-AUROC rising from 0.9845 to 0.9909 and Seg-AP-det improving from 0.9097 to 0.9296. These results provide a direct quantitative indication of the effectiveness of synthetic data augmentation in improving both anomaly detection and localization performance.

At the pixel level, the hybrid dataset achieved a P-AUROC of 0.9938 and an AUPRO of 0.9884, outperforming the original dataset by 0.92% and 1.23%, respectively. The greatest gain was observed in AP-loc, which increased from 0.7475 to 0.7873, indicating an improved localization of anomalous regions.

These results suggest that the proposed hybrid dataset improves the diversity of features and improves the generalizability of SuperSimpleNet, leading to better anomaly detection and localization performance.

5.3 Visual interpretation

To provide a visual interpretation of the proposed anomaly detection process, a composite representation is generated for each test sample as in Fig. 3. The pixel-level anomaly localization and verification pipeline consists of five sequential components used to evaluate the correspondence between model predictions and ground-truth annotations. The first component, the input image (i), represents the raw material surface under inspection, containing potential localized textural or structural defects. The second component, the ground-truth mask (ii), is a binary, human-annotated

Metric	Original Dataset	Hybrid Dataset	Relative Improvement
I-AUROC	0.9840	**0.9907**	+0.68%
AP-det	0.9271	**0.9454**	+1.97%
Seg-I-AUROC	0.9845	**0.9909**	+0.65%
Seg-AP-det	0.9097	**0.9296**	+2.19%
P-AUROC	0.9847	**0.9938**	+0.92%
AUPRO	0.9764	**0.9884**	+1.23%
AP-loc	0.7475	**0.7873**	+5.32%

Table 4: Ablation study comparing the original dataset and the proposed hybrid dataset at 30 epochs.

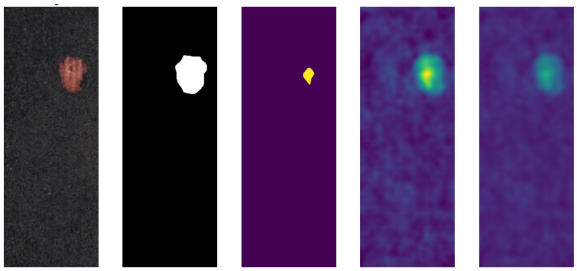


Figure 3: Composite visualization of the pixel-level anomaly localization and verification pipeline for test samples, illustrating the relationship between input data, ground-truth annotation, model prediction, and anomaly score representations.

reference map in which defect regions are explicitly defined at the pixel level. The third component, the predicted mask (iii), is obtained by applying a globally selected threshold (τ) to the network's continuous anomaly score map, producing a binary segmentation that reflects correct detections or missed defect regions depending on the ground truth. The fourth component, the normalized anomaly map (iv), is derived from the raw anomaly score map via min–max normalization and provides a visually interpretable heatmap of spatial anomaly confidence. The fifth component, the raw anomaly map (v), contains the unnormalized pixel-wise anomaly scores along with the image-level confidence score and peak segmentation score (SScore), serving as the quantitative basis for thresholding and performance evaluation.

5.4 Computational cost of the proposed approach: A Green AI perspective

During the experimental work in [33] based on the DefectGAN model, the dataset generation pipeline required resizing the original images from 230×630 pixels to 224×224 pixels to satisfy the strict square-input constraints of DefectGAN. The generative model was subsequently trained for 200 epochs to generate 63,000 synthetic instances along with their respective defect masks, collected at different training epochs: 120, 160, and 200. Following this, a HITL curation phase was deployed to filter out non-viable patterns. Ultimately, only 1,535 defect images were retained, as numerous samples suffered from localization alignment errors, irregular defect scale anomalies (e.g., excessively small or large), or severely unphotorealistic morphology. A final post-processing step was also required to remove salt-and-pepper noise from the generated mask surfaces. In terms of pure SDG execution time—excluding preprocessing, post-processing, and manual HITL filtering, and operating on scaled-down inputs—the training workflows deployed via Google Colab Pro[2] exhibited estimated time of arrival (ETA) thresholds from 17 to 590 hours, depending on the available hardware configuration.

In contrast, the proposed procedural framework does not require image resizing or post-processing, natively maintaining the original Kolektor Surface-Defect Dataset 2 (KSDD2) resolution of 230×630 pixels. The proposed pipeline synthesizes 1,000 complete image-mask pairs in just 150 s. Furthermore, approximately 75% of the generated samples successfully pass the HITL validation phase, demonstrating high visual fidelity to authentic manufacturing anomalies. Crucially, rather than restricting the

[2]https://colab.google/

outputs to the baseline statistical distribution of KSDD2, the proposed framework generalizes to simulate broader taxonomy variations of metal surface irregularities. A summary of raw processing time metrics is detailed in Table 5.

Method	Hardware Architecture	Generation Time
DefectGAN [33]	CPU	590.00 h
DefectGAN [33]	TPU+high memory	58.00 h
DefectGAN [33]	NVIDIA A100 GPU	17.00 h
Proposed Procedural	Intel Core i7-8850H CPU	**0.06 h**

Table 5: Computational Generation Time Comparison for 1,500 Synthetic Samples

5.5 Energy consumption for generating synthetic data

To assess the operational efficiency of the proposed synthetic data generation pipeline, we performed a comparative analysis of Energy-to-Solution (EtS) on two heterogeneous hardware platforms. EtS is adopted as the primary efficiency metric because it captures the total energy required to complete the full workload and is therefore suitable for comparing systems with different compute characteristics.

The Energy-to-Solution is defined as:

$$\text{EtS} = E = P \cdot t \tag{1}$$

where E is the total energy consumed in watt-hours (Wh), P denotes the average active processor power draw in watts (W), and t denotes the total execution time in hours (h). Accordingly, EtS is expressed in watt-hours (Wh) and may be reported in kilowatt-hours (kWh) for readability and scale normalization. This metric directly reflects the operational cost of completing the synthetic data generation task and supports a hardware-agnostic comparison of computational efficiency. In the spirit of Green AI, this evaluation emphasizes energy efficiency alongside task completion, rather than relying solely on raw throughput or nominal hardware capability.

The evaluated hardware configurations are: (i) an NVIDIA A100 GPU (PCIe variant, 300, W Thermal Design Power [TDP]) representing a centralized high-performance cloud infrastructure utilized in premium environments like Google Colab Pro, and (ii) an Intel Core i7-8850H CPU (45, W TDP) representing a localized edge workstation.

The benchmark results are summarized in Table 6.

Methodology	Platform	Nominal Power (W)	Runtime (h)	Energy-to-Solution (Wh)	Energy-to-Solution (kWh)
DefectGAN	NVIDIA A100 GPU	250	17.00	4250.00	4.25000
Procedural	Intel Core i7-8850H CPU	45	0.06	2.70	0.0027
Savings (Δ Absolute / %)	—	—	**16.94**	**4247.30**	**4.24730 (99.93%)**

Table 6: Energy-to-Solution (EtS) comparison across generation methodologies and heterogeneous compute platforms

The results indicate that EtS provides a clear and interpretable measure of computational efficiency across heterogeneous architectures for different SDG methods. Although the NVIDIA A100 GPU is designed for high-throughput parallel workloads, its EtS is substantially higher for the DefectGAN training method due to the long execution time required by this SDG workflow (17.00 h), resulting in an energy cost of 4.25 kWh. By contrast, the Intel Core i7-8850H CPU completes the procedural synthetic data generation method in 0.06 h, yielding an EtS of 2.70 Wh, or 0.0027kWh. Relative to the DefectGAN method executed on the GPU configuration, the CPU-based procedural method achieves an absolute reduction of 4247.30 Wh, corresponding to a 99.93% decrease in EtS.

These findings suggest that, for the SDG methods evaluated in this study, energy efficiency is determined more by the execution characteristics of the generation approach than by peak compute capability. This is consistent with Green AI literature, which advocates treating energy consumption and environmental cost as primary evaluation criteria in AI systems.

6 Conclusion

In the context of Green AI, addressing data scarcity requires synthetic data generation methods that jointly balance visual realism, computational efficiency, and downstream model performance while minimizing energy consumption. This paper presented a Green AI-oriented procedural SDG framework for industrial metal surface defect detection based on computationally efficient procedural synthesis. The proposed pipeline generates realistic defective samples together with pixel-accurate defect masks, followed by a HITL validation stage to ensure annotation quality and visual fidelity. The resulting hybrid data set was used to train the lightweight SuperSimpleNet architecture, demonstrating that increasing the amount of high-quality synthetic data consistently improves the detection, segmentation, and localization performance of anomalies.

Compared to training in the original KSDD2 dataset, the hybrid data set improved all evaluation metrics, with the largest gain of 5.32% in AP-loc, while also increasing image and pixel detection accuracy. These results indicate that carefully designed procedural synthetic data can effectively mitigate data scarcity without relying on computationally expensive generative models, extensive manual annotation, or more complex deep learning architectures. Consequently, the proposed framework provides a scalable and resource-efficient approach to develop high-performance industrial vision systems while advancing the sustainability objectives of Green AI.

Within this framework, Visual Knowledge Discovery (VKD) [34, 35, 36, 37] offers a complementary perspective by supporting the extraction and interpretation of meaningful visual patterns and structural regularities in both real and synthetic data. VKD facilitates the analysis of latent defect characteristics, the identification of discriminative texture regions, and the interpretation of feature representations learned by deep neural networks. Moreover, it provides a principled mechanism for assessing visual consistency between real and synthetically generated defects, thereby enabling systematic evaluation and refinement of synthetic data generation methods. By enhancing the interpretability of both generated data and downstream models, VKD supports more effective use of synthetic datasets while reducing the reliance on extensive real-world data collection. In this way, VKD aligns naturally with Green AI objectives by promoting resource-efficient, transparent, and reliable industrial AI systems.

References

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, Massachusetts, USA: MIT Press, 2016. [Online]. Available: https://www.deeplearningbook.org/

[2] K. Man and J. Chahl, "A review of synthetic image data and its use in computer vision," *Journal of Imaging*, vol. 8, no. 11, p. 310, 2022.

[3] G. Pei, J. Zhang, M. Hu, Z. Zhang, C. Wang, Y. Wu, G. Zhai, J. Yang, and D. Tao, "Deepfake generation and detection: A benchmark and survey," *ACM Comput. Surv.*, vol. 58, no. 11, Apr. 2026. [Online]. Available: https://doi.org/10.1145/3801962

[4] E. Barbierato and A. Gatti, "Toward green ai: A methodological survey of the scientific literature," *IEEE Access*, vol. PP, pp. 1–1, 01 2024.

[5] M. Abufadda and K. Mansour, "A survey of synthetic data generation for machine learning," in *2021 22nd International Arab Conference on Information Technology (ACIT)*, 2021, pp. 1–7.

[6] Y. Lu, L. Chen, Y. Zhang, M. Shen, H. Wang, X. Wang, C. van Rechem, T. Fu, and W. Wei, "Machine learning for synthetic data generation: A review," 2025. [Online]. Available: https://arxiv.org/abs/2302.04062

[7] X. Wu, L. Xiao, Y. Sun, J. Zhang, T. Ma, and L. He, "A survey of human-in-the-loop for machine learning," *Future Generation Computer Systems*, vol. 135, pp. 364–381, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X22001790

[8] D. Kundu and B. L. Evans, "Full-reference visual quality assessment for synthetic images: A subjective study," in *2015 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2015, pp. 2374–2378.

[9] T. L. Goff, "Recommendations for public action towards sustainable generative AI systems," *CoRR*, vol. abs/2402.01646, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.01646

[10] N. Bashir, P. Donti, J. Cuff, S. Sroka, M. Ilic, V. Sze, C. Delimitrou, and E. Olivetti, "The Climate and Sustainability Implications of Generative AI," *An MIT Exploration of Generative AI*, mar 27 2024, https://mit-genai.pubpub.org/pub/8ulgrckc.

[11] R. Schwartz, J. Dodge, N. A. Smith, and O. Etzioni, "Green ai," *Commun. ACM*, vol. 63, no. 12, p. 54–63, Nov. 2020. [Online]. Available: https://doi.org/10.1145/3381831

[12] T. Nazaré, J. Gadelha, E. Nepomuceno, and R. Lozi, "Green computing for energy transition: A survey," *IEEE Latin America Transactions*, vol. 21, no. 9, pp. 937–948, 2023.

[13] Y. Zhou, X. Lin, X. Zhang, M. Wang, G. Jiang, H. Lu, Y. Wu, K. Zhang, Z. Yang, K. Wang, Y. Sui, F. Jia, Z. Tang, Y. Zhao, H. Zhang, T. Yang, W. Chen, Y. Mao, Y. Li, D. Bao, Y. Li, H. Liao, T. Liu, J. Liu, J. Guo, X. Zhao, Y. WEI, H. Qian, Q. Liu, X. Wang, W. Kin, Chan, C. Li, Y. Li, S. Yang, J. Yan, C. Mou, S. Han, W. Jin, G. Zhang, and X. Zeng, "On the opportunities of green computing: A survey," *arXiv:2311.00447*, 2023. [Online]. Available: https://arxiv.org/abs/2311.00447

[14] C. Clemm, L. Stobbe, K. Wimalawarne, and J. Druschke, "Towards green ai: Current status and future research," in *2024 Electronics Goes Green 2024+(EGG)*. IEEE, 2024, pp. 1–11.

[15] A. Asperti, D. Evangelista, and E. L. Piccolomini, "A survey on variational autoencoders from a green ai perspective," *SN Computer Science*, vol. 2, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:232092485

[16] J. Božič, D. Tabernik, and D. Skočaj, "Mixed supervision for surface-defect detection: from weakly to fully supervised learning," *Computers in Industry*, 2021.

[17] D. Kingma and M. Welling, "An introduction to variational autoencoders," *Foundations and Trends® in Machine Learning*, vol. 12, pp. 307–392, 11 2019.

[18] Y. Chen, J. Liu, L. Peng, Y. Wu, Y. Xu, and Z. Zhang, "Auto-encoding variational bayes," *Cambridge Explorations in Arts and Sciences*, vol. 2, 02 2024.

[19] G. Zhang, K. Cui, T.-Y. Hung, and S. Lu, "Defect-gan: High-fidelity defect synthesis for automated defect inspection," in *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 2523–2533.

[20] S. Niu, H. Lin, T. Niu, B. Li, and X. Wang, "Defectgan: Weakly-supervised defect detection using generative adversarial network," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, 2019, pp. 127–132.

[21] J. Song, D. Park, K. Baek, S. Lee, J. Choi, E. Kim, and S. Yoon, "Defectfill: Realistic defect generation with inpainting diffusion model for visual inspection," in *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 18 718–18 727.

[22] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, "SDXL: Improving latent diffusion models for high-resolution image synthesis," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=di52zR8xgf

[23] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 4172–4182.

[24] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: a nested U-Net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings*. Berlin, Heidelberg: Springer-Verlag, 2018, p. 3–11. [Online]. Available: https://doi.org/10.1007/978-3-030-00889-5_1

[25] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham: Springer International Publishing, 2018, pp. 833–851.

[26] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 1280–1289.

[27] L. C. C. Filho, J. L. A. Samatelo, and R. F. Vassallo, "Hierarchical vision transformer using shifted windows for skeleton-based action recognition," in *2025 38th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, 2025, pp. 1–6.

[28] B. Rolih, M. Fučka, and D. Skočaj, "Supersimplenet: Unifying unsupervised and supervised learning for fast and reliable surface defect detection," in *Pattern Recognition*, A. Antonacopoulos, S. Chaudhuri, R. Chellappa, C.-L. Liu, S. Bhattacharya, and U. Pal, Eds. Cham: Springer Nature Switzerland, 2025, pp. 47–65.

[29] Z. Liu, Y. Zhou, Y. Xu, and Z. Wang, "Simplenet: A simple network for image anomaly detection and localization," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 20 402–20 411.

[30] T. Fawcett, "Introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, pp. 861–874, 06 2006.

[31] P. Bergmann, K. Batzner, M. Fauser, D. Sattlegger, and C. Steger, "The mvtec anomaly detection dataset: A comprehensive real-world dataset for unsupervised anomaly detection," *Int. J. Comput. Vision*, vol. 129, no. 4, p. 1038–1059, Apr. 2021. [Online]. Available: https://doi.org/10.1007/s11263-020-01400-4

[32] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, pp. 303–338, 06 2010.

[33] S. Mătase and R. Andonie, "Synthetic data generation for enhanced visual metal defect detection," in *2025 29th International Conference Information Visualisation (IV)*, vol. 29th International Conference Information Visualisation (IV), 2025, pp. 231–236.

[34] B. Kovalerchuk, *Visual Knowledge Discovery and Machine Learning*. Springer, 2018, vol. 144.

[35] B. Kovalerchuk, K. Nazemi, R. Andonie, N. Datia, and E. Banissi, *Integrating Artificial Intelligence and Visualization for Visual Knowledge Discovery*. Springer, 2022.

[36] B. Kovalerchuk, R. Andonie, N. Datia, K. Nazemi, and E. Banissi, "Visual knowledge discovery with artificial intelligence: Challenges and future directions," in *Integrating Artificial Intelligence and Visualization for Visual Knowledge Discovery*. Springer Charm, 2022, pp. 1–27.

[37] B. Kovalerchuk, K. Nazemi, R. Andonie, N. Datia, and E. Bannissi, *Artificial Intelligence and Visualization: Advancing Visual Knowledge Discovery*. Springer, 2024.



Copyright ©2026 by the authors. Licensee Agora University, Oradea, Romania.
This is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International License.
Journal's webpage: http://univagora.ro/jour/index.php/ijccc/



This journal is a member of, and subscribes to the principles of,
the Committee on Publication Ethics (COPE).
https://publicationethics.org/members/international-journal-computers-communications-and-control

Cite this paper as:

Mătase, S.; Andonie, R. (2026). Green AI-Driven Synthetic Data Generation for Improved Industrial Metal Surface Defect Detection, *International Journal of Computers Communications & Control*, 21(6), 7591, 2026.

https://doi.org/10.15837/ijccc.2026.6.7591